HIRSCHMANN
A BELDEN BRAND

User Manual

Configuration
TBF - The Belden Firewall

Technical support
https://hirschmann-support.belden.com

The naming of copyrighted trademarks in this manual, even when not specially indicated, should not be taken to mean that these names may be considered as free in the sense of the trademark and tradename protection law and hence that they may be freely used by anyone.

© 2026 Hirschmann Automation and Control GmbH

Manuals and software are protected by copyright. All rights reserved. The copying, reproduction, translation, conversion into any electronic medium or machine scannable form is not permitted, either in whole or in part. An exception is the preparation of a backup copy of the software for your own use.

The performance features described here are binding only if they have been expressly agreed when the contract was made. This document was produced by Hirschmann Automation and Control GmbH according to the best of the company's knowledge. Hirschmann Automation and Control GmbH reserves the right to change the contents of this document without prior notice. Hirschmann Automation and Control GmbH can give no guarantee in respect of the correctness or accuracy of the information in this document.

Hirschmann Automation and Control GmbH can accept no responsibility for damages, resulting from the use of the network components or the associated operating software. In addition, we refer to the conditions of use specified in the license contract.

You can get the latest version of this manual on the Internet at *doc.hirschmann.com*.

Hirschmann Automation and Control GmbH

Stuttgarter Str. 45-51, 72654 Neckartenzlingen

Germany

Contents

1. General explanations....16
Introduction....16
Architecture....16
Technology stack....16
Release schedule....16
Set up default login....17
Appliances....17
Supported browsers....17
Breadcrumb navigation....17
Search functionality....17
Default firewall users....18
Default firewall rules....18
Default services....18
2. XDP accelerator....20
XDP speedups....20
XDP DDoS protection....20
3. Appliances user manuals....21
IAF-240....21
Supported browsers....21
Hardware installation guide....22
Set up default login....22
Handle certificate warning....23
Use setup wizard....23
Licence activation....24
TBF-800V....28
TBF-800V requirements....28
Install TBF-800V on VirtualBox....29
Install TBF-800V on Proxmox....33
Install TBF-800V on Hyper-V....37
Install TBF-800V system....42
Set up default login....45
Access TBF-800V web UI....46
Licence activation....48
4. CLI....53
Access the CLI....53
Navigate the CLI....53
View the interface configuration....53
Enable the DNS server....54
5. Console access....55
SSH console access....55
Serial console access....57
6. Dashboard....59
States widget....60

Security widget....61
7. System settings....63
Global settings....63
Settings....63
Updates....67
Notifications....69
Advanced settings....70
Licence....76
Portal settings....76
Portal connection....77
Remote access through the portal....77
Addons....77
Copyright....79
8. Authentication....80
User....80
User permissions....81
API token management....81
SSH access configuration....82
Two-factor authentication....82
Group....82
Authentication server....83
LDAP server....84
LDAP scope....85
LDAP user....85
LDAP advanced....85
RADIUS server....86
Identity awareness....86
Identity awareness settings....87
Identity awareness user settings....88
9. Interfaces....90
Assign interfaces....91
Interface abstraction....92
Set up interfaces....92
General interface settings....92
IPv4 interface settings....93
IPv6 interface settings....94
Advanced interface settings....95
Wi-Fi client setup....96
WPA authentication protocols....96
Interface-dependent options....96
VLAN....96
QinQ....98
Bridge....99
Create a new bridge....99
Bridge VLAN....100

STP/RSTP....101
Bridge port settings....101
Bridge port STP/RSTP settings....102
Compatibility....102
Bridging scenarios....103
Bond (link aggregation)....104
Create and set up bonds....104
Bond settings....106
Tunnel....107
Create a new tunnel....107
Interface tunnel settings....108
GRE with failover....108
PPPoE....109
PPPoE settings....109
Set up a DSL connection via PPPoE....110
PPP....110
PPP settings....111
Set up an LTE/5G PPP connection....112
LTE450 overview....114
Set up an LTE450 PPP connection....114
MacVLAN....115
SHDSL....115
SHDSL tuning....117
Bitrate mode....117
SHDSL configuration....117
VDSL....119
VDSL configuration....120
VRF....121
VRF firewall integration....121
VRF management network....122
VRF layer 3 VPN capabilities....122
VXLAN....122
VXLAN settings....123
Interface groups....124
10. Firewall....125
Firewall general....125
Stateful versus stateless....126
Firewall flow....126
Firewall processing....127
Flowtable bypass....127
eXpress Data Path....128
Network objects....128
Hosts (single IP addresses)....129
Hosts (network range)....129
Ports....130

MAC addresses ... 130
DNS domains ... 131
Firewall rules (forward and input) ... 131
Create and update firewall rules ... 132
Move firewall rules ... 133
Delete firewall rules ... 133
Search firewall rules ... 134
Convert firewall rules ... 134
Firewall rule settings ... 134
Firewall rule usage and changes ... 137
Hostnames in firewall rules ... 138
Use TCPDump with firewall rules ... 138
Trace firewall rules ... 138
Global firewall rules (forward and input) ... 139
Firewall rule evaluation process ... 139
Firewall rule search criteria ... 139
Bridge firewall rules (forward) ... 140
DNAT (prerouting) ... 141
Move DNAT rules ... 142
Create and update DNAT rules ... 142
DNAT general settings ... 142
DNAT sources settings ... 143
DNAT destinations settings ... 143
DNAT NAT settings ... 144
DNAT advanced settings ... 144
SNAT (postrouting) ... 144
Move SNAT rules ... 146
Create and update SNAT rules ... 146
Search SNAT rules ... 149
SNAT changes ... 149
BiNAT (prerouting and postrouting) ... 149
Move BiNAT rules ... 149
Create and update BiNAT rules ... 150
Search BiNAT rules ... 152
Firewall rules advanced ... 152
Geo blocking ... 153
Office 365 ... 154
Amazon Web Services ... 154
Google ... 155
Miscellaneous ... 156
Custom rules ... 157
Firewall time control ... 157
Time control ... 158
App control ... 158
Apps ... 159

Flows....162
Categories....162
Rules....163
Settings....165
DDoS rules....165
General settings....166
Create and manage firewall rule dividers....167
Learning mode....168
11. Enforcer....169
Enforcer general....169
Enforcer rule requirements....170
AMP....170
AMP settings....171
Task codes settings....174
AMP global settings....174
Task code reference....175
AMP message types reference....176
DNP3....176
DNP3 settings....177
DNP3 objects....178
DNP3 function codes....179
DNP3 default object list....180
ENIP....181
ENIP settings....181
ENIP class IDs....182
GOOSE....182
GOOSE settings....183
IEC104....183
IEC104 settings....184
IEC104 type ID meanings....186
Modbus....188
Modbus settings....188
Modbus rules....190
Modbus function codes....191
OPC....192
OPC settings....193
12. Certificates....194
Certificate settings....194
CAs....195
CA settings....196
Manage CAs....196
CSRs....197
CSR settings....198
Import CSR....198
Sign with certificate authority....198

CRL ... 201
Let's Encrypt ... 201
Create a Let's Encrypt account ... 202
Sign a Let's Encrypt certificate ... 202
Renew a Let's Encrypt certificate ... 203
Revoke a Let's Encrypt certificate ... 203
DNS ACME handle ... 203
13. Virtual IPs ... 205
VRRP ... 205
IP alias ... 206
Custom VRRP scripts ... 206
14. DynDNS ... 207
Set DynDNS settings ... 207
DynDNS settings ... 207
15. Routing ... 209
Routes ... 209
Gateway route ... 210
Interface route ... 210
Multipath route ... 210
Route settings ... 210
Routing tables ... 211
Load balancing ... 213
Load balancing settings ... 213
Gateways ... 214
Create a gateway ... 215
Gateway monitoring ... 216
Monitoring settings ... 217
Advanced settings ... 218
MPLS ... 218
Encapsulation ... 219
Label switch ... 220
Decapsulation ... 220
LDP ... 221
Dynamic ... 222
General settings ... 222
Administrative distance ... 223
BFD ... 223
Set up a BFD peer ... 224
BGP ... 224
BGP neighbor ... 226
ISIS ... 227
ISIS interface ... 228
OSPF ... 229
OSPF interface ... 231
OSPF area ... 232

OSPFv6 ... 233
OSPFv6 interface ... 234
Access list ... 236
Prefix list ... 236
Route map ... 237
Use cases ... 239
Multi-gateway setup ... 239
Policy routing ... 241
16. Services ... 243
Apcupsd ... 243
General settings ... 244
Power failures settings ... 245
UPS sharing settings ... 246
802.1X authenticator ... 247
802.1X authenticator settings ... 247
Set up RADIUS authentication ... 248
Set up an EAP authentication server ... 248
Avahi ... 250
Avahi settings ... 250
Captive portal ... 251
Captive portal instance ... 251
Captive portal files ... 254
Redirect device IDs ... 254
Voucher ... 255
Redirect mode ... 256
Cron ... 256
Preconfigured crons ... 257
Create a new cron job ... 257
DHCP and RA ... 259
Set up network booting ... 259
MAC and DUID deny ... 260
Extra options ... 260
Relay server ... 260
IPv6 router advertisement ... 261
IPv6 prefix delegation ... 262
Host reservation ... 262
DNS ... 263
General settings ... 263
Advanced settings ... 265
Host overrides ... 267
Domain overrides ... 267
DNS firewall ... 267
Redirect DNS requests ... 269
Create wildcard domain DNS entries ... 270
DNS troubleshooting ... 270

HAProxy ... 271
General settings ... 272
Backend settings ... 273
Frontend settings ... 276
Advanced settings ... 278
Error page settings ... 279
Access control lists settings ... 280
Actions settings ... 280
Set up DNAT for HAProxy port redirection ... 281
IGMPProxy ... 282
Upstream settings ... 283
Downstream settings ... 283
Intrusion detection ... 284
General settings ... 284
Performance settings ... 286
Update rule packages ... 288
Manage intrusion detection rules ... 289
Example Suricata configuration ... 290
Netflow ... 290
Install Netflow ... 291
Netflow settings ... 291
Ntopng ... 292
Ntopng settings ... 292
Install ntopng ... 294
NTP ... 294
NTP access restrictions ... 295
Custom access restrictions ... 296
QoS ... 296
QoS classes ... 297
QoS assignment ... 298
Limiter ... 298
SNMP ... 298
SNMP MIB ... 300
Startup scripts ... 300
Web filter ... 301
General settings ... 302
HTTP settings ... 302
HTTPS settings ... 302
Cache settings ... 303
ClamAV anti-virus settings ... 304
WPAD autoconfigure settings ... 305
Advanced settings ... 305
Access control lists ... 306
HTTP access rules ... 307
Authentication ... 308

LDAP authentication....309
Active Directory authentication....310
Blocklist....310
Set up the web filter as proxy....311
Set up a transparent proxy....311
Set up a proxy with anti-virus....312
Set up blocklist filtering....312
UPnP and NAT-PMP....313
UPnP and NAT-PMP settings....314
Web application firewall....314
OWASP CRS....315
WAF settings....315
Set up the WAF allowlist....316
WAF plugins....316
Advanced WAF settings....317
Manage WAF rules....319
ZeroTier....320
ZeroTier settings....320
ZeroTier multipath....321
17. VPN....322
IPsec....322
IPsec general settings....323
IPsec phase 1....324
IPsec phase 2....330
IPsec client settings....333
IPsec route based (VTI/XFRM)....334
Set up GRE over IPsec....335
IPsec troubleshooting....337
OpenVPN....338
OpenVPN general....339
OpenVPN settings....339
Export OpenVPN client configurations....350
OpenVPN override....351
WireGuard....352
WireGuard settings....352
WireGuard export settings....353
WireGuard peer settings....353
WireGuard peer export....354
WireGuard QR code....354
WireGuard import configuration....354
Example WireGuard configuration....354
WebVPN....358
WebVPN settings....358
WebVPN server settings....359
WebVPN user settings....361

WebVPN group settings ... 361
Use WebVPN ... 362
18. Apps ... 365
Settings ... 365
Set app settings ... 366
Apps overview ... 366
Networking ... 367
Images ... 367
Volumes ... 367
Containers ... 367
Container settings ... 368
Run a new application ... 369
Templates ... 370
Create a container from a template ... 371
Examples ... 371
UniFi Controller ... 372
Debian ... 372
MySQL ... 373
19. High availability ... 374
General high availability ... 374
Interface setup ... 374
Synchronization interface ... 375
VRRP mode ... 377
High availability nodes ... 378
Secondary firewall ... 378
Onboarding wizard ... 378
Configuration sync ... 380
Set up configuration sync ... 380
Configuration sync settings ... 381
Trigger manual synchronization ... 381
States sync ... 382
VRRP shared virtual IP address ... 382
HA outbound NAT ... 384
Gateway IP address ... 384
DHCP high availability ... 384
DHCP high availability settings ... 385
Setup examples ... 386
Configuration sync only ... 386
Full failover setup ... 386
20. Diagnostics ... 388
ARP & ND ... 388
Audit log ... 389
802.1X authenticator ... 390
BFD ... 390
BGP ... 391

Bond ... 391
Bridge ... 392
Captive portal ... 393
DDoS ... 393
Debug report ... 394
DHCP ... 395
DNS ... 397
DNS domain ... 398
Dynamic routing ... 398
Firewall ... 398
Gateways ... 401
GUI logins ... 402
Hard drives ... 402
HASync ... 404
Host connections ... 404
Identity awareness ... 405
Interfaces ... 406
Intrusion detection ... 407
IPsec ... 410
ISIS ... 412
Logging ... 413
Cellular ... 414
MDRaid ... 416
MPLS ... 417
NTP ... 418
OpenVPN ... 418
OSPF ... 420
OSPFv6 ... 420
QoS ... 420
Routes ... 421
Services ... 422
SFP ... 423
SHDSL ... 423
States ... 424
STP ... 426
Sysstat ... 427
System actions ... 428
UPnP and NAT-PMP ... 428
VDSL ... 429
Virtual IPs ... 430
VRF ... 432
VXLAN ... 432
Web filter ... 432
Web application firewall ... 434
WireGuard ... 436

Worker log....436
XDP....437
ZeroTier....438
21. Tools....440
ARPing....440
Backup & restore....441
Backup....441
Restore....441
Web console....441
Factory defaults....442
File manager....442
Use the file manager....443
Iftop....443
LLDP....443
Patch....444
Patch creation settings....444
Ping....445
Use ping....445
Ping examples....445
Restart....446
Script....446
Setup wizard....447
Shutdown....447
TCPDump....447
TCPDump settings....448
Troubleshooting via TCPDump....449
Templates....450
Top....451
Traceroute....451
Use traceroute....451
Wake-on-LAN....452
Wake-on-LAN devices....452
22. Configuration examples....453
Windows updates....453
IPv6 multi-WAN....453
Network Prefix Translation (NPTv6)....453
Custom scripts....454
System notification....455
Gateway change notification....455
23. API....456
General....456
Access the API via a console....457
24. Commands....459
Speedtest....459
Command-line tools....459

logcleanup....460
check-routes....460
XDP command-line tools....460
xdp_vtair_user....460
xdp_stats....461
25. Roadmap....462
26. Changelog....463
TBF 26.10....463
TBF 26.07....464
TBF 26.04....465
TBF 26.01....466

1. General explanations

This page contains general information about the system.

This section contains general information about TBF, including system architecture, login procedures, user interface features, and default configurations.

Introduction

TBF is a Linux-based firewall system that helps provide network security and network infrastructure management features.

TBF includes a modern web-based management interface, a REST API, and a command-line interface for flexible system administration.

Architecture

TBF operates on Linux with a customized kernel.

TBF operates on the Linux operating system and uses a customized Linux kernel to help ensure compatibility and network performance.

Technology stack

The technology stack consists of open-source software components.

TBF is built on a foundation of open-source software projects. Each component serves a specific networking or security function:

Table 1:

Component	Function
SNMPD	Simple Network Management Protocol services
FRR	Dynamic routing protocol support
Kea	Dynamic Host Configuration Protocol (DHCP) server
Unbound	Domain Name System (DNS) resolver
ntp.org	Network Time Protocol (NTP) daemon
keepalived	Virtual Router Redundancy Protocol (VRRP) and high availability failover
Docker	Application containerization platform
Suricata	Network intrusion detection and application control
Squid	Web content filtering and caching proxy
Coraza	Web application firewall functionality
HAProxy	Reverse proxy server and load balancing
Apache Guacamole	Clientless remote desktop gateway (WebVPN)
OpenVPN	Virtual private network (VPN) solution
strongSwan	IPsec-based VPN key management
WireGuard	Modern VPN protocol implementation

Release schedule

This page describes the quarterly release schedule and version numbering.

TBF releases new versions quarterly. The version number reflects the month and year of the release. For example, we released `2026.01.1` in January 2026.

Releases occur in January, April, July, and October. We number them by replacing `YYYY` with the year of the release:

- `YYYY.01`
- `YYYY.04`
- `YYYY.07`
- `YYYY.10`

We update the kernel with the April (04) and October (10) releases. There are exceptions, for example, when major vulnerabilities or other serious detected issues require us to release a kernel update outside the regular release schedule.

Set up default login

To set up the default login credentials for the GUI, create a password for the administrator user when you first access TBF.

When you first access TBF, set up the default login credentials for the GUI. TBF requires you to create a password for the default user `admin`.

To access TBF through SSH or console, use the `root` user account. Both the `root` and `admin` users share the same password. This means you log in only through SSH after you establish the default password.

TBF comes with an initial password: `private`. To access the GUI, navigate to *https://192.168.1.1*.

Appliances

This page lists the available hardware appliances.

IAF-240 on page 21

TBF-800V on page 28

Supported browsers

This page lists the supported web browsers.

TBF supports the following browsers:

- Google Chrome
- Microsoft Edge
- Mozilla Firefox
- Safari

Note: Internet Explorer is an unsupported browser and displays some GUI features incorrectly.

Breadcrumb navigation

This page describes the breadcrumb navigation and its additional menu options.

Each page includes a navigational breadcrumb. Some pages offer additional options in the breadcrumb menu bar.

These options include links to:

- Service overview page
- Service-specific diagnostics
- Log file

Search functionality

This page describes the search box in the upper navigation bar.

The search box in the upper navigation bar searches through the menu and returns page results.

Note: The search box does not search through stored data.

Default firewall users

TBF includes three preconfigured user accounts that are active by default.

User name	Primary function	Description
`admin`	Web interface administrator	Helps provide administrative access to the web-based management interface of the firewall
`hasync`	High availability synchronization	Manages configuration synchronization between firewall units in high availability setups. Uses a randomly generated password
`root`	Command-line administrator	Helps provide SSH access for advanced configuration and troubleshooting. The password automatically matches the `admin` user password.

Default firewall rules

This page describes the default firewall rules applied to interfaces.

TBF applies default firewall rules only to the **LAN** interface. These rules control which network traffic can pass through the interface.

Protocol	Source IP	Source port	Destination IP	Destination port	Description
TCP	Any	Any	LAN address	22, 80, 443	Anti-lockout rule
TCP/UDP	LAN	Any	LAN address	53, 853	DNS server
ICMP	LAN	-	LAN address	-	ICMP to TBF
Any	Any	Any	Private networks	Any	Access to private IP addresses v4 and v6
Any	Any	Any	NOT private networks	Any	Access to public IP addresses v4 and v6

The **WAN** interface operates differently and blocks all incoming traffic by default. This interface also includes an additional firewall rule that specifically blocks traffic from private IP addresses.

For information about open ICMP and ICMPv6 ports on each interface, see the tables below.

Default services

This page lists the default services and their open ports.

The following tables show the services and their default open ports in TBF:

Service	Port	Protocol	Default firewall rule	Active interface	Description
DNS	53	TCP and UDP	Yes on **LAN** interface	**LAN**	DNS server
DNS	853	TCP and UDP	Yes on **LAN** interface	**LAN**	DNS server TLS
HTTP	80	TCP	Yes on **LAN** interface	Any	Web server
HTTPS	443	TCP	Yes on **LAN** interface	Any	Web server
DHCP	67	UDP	Yes on **LAN** interface (except IAF-240)	**LAN**	DHCP server
SSH	22	TCP	Yes on **LAN** interface	Any	SSH server
NTP	123	UDP	No Blocked	Any	NTP server
ICMP		ICMP	Yes on **LAN** interface, also see table below	N/A	ICMP messages
ICMPv6		ICMPv6	See table below	N/A	ICMPv6 messages

Open ICMP types to TBF:

ICMP type	Input interface	Description
All	**LAN**	LAN ICMP to TBF
Destination unreachable (3)	ALL	Destination unreachable message

ICMP type	Input interface	Description
Time exceeded (11)	ALL	Time exceeded message
Parameter problem (12)	ALL	Parameter problem

Open ICMPv6 types to TBF:

ICMPv6 type	Input interface	Description
Destination unreachable (1)	ALL	Destination unreachable message
Packet Too Big (2)	ALL	Packet too big
Time exceeded (3)	ALL	Time exceeded message
Parameter problem (4)	ALL	Parameter problem
Neighbor Solicitation (135)	ALL	Neighbor solicitation
Neighbor Advertisement (136)	ALL	Neighbor advertisement
Multicast Listener Query (130)	ALL	Multicast listener query
Multicast Listener Report (131)	ALL	Multicast listener report
Multicast Listener Done (132)	ALL	Multicast listener done
Multicast Listener Report v2 (143)	ALL	Multicast listener report
Multicast Router Advertisement (151)	ALL	Multicast listener report
Multicast Router Solicitation (152)	ALL	Multicast listener report
Multicast Router Termination (153)	ALL	Multicast listener report
Echo Reply (129)	ALL (fe80::/10, ff02::/16 <-> fe80::/10, ff02::/16)	Link local only
Router Solicitation (133)	ALL (fe80::/10, ff02::/16 <-> fe80::/10, ff02::/16)	Link local only
Router Advertisement (124)	ALL (fe80::/10, ff02::/16 <-> fe80::/10, ff02::/16)	Link local only

2. XDP accelerator

The XDP accelerator uses eXpress Data Path (XDP) and extended Berkeley Packet Filter (eBPF) technologies to accelerate network traffic processing by up to 5 times.

The XDP accelerator enhances network performance by processing packets directly at the network interface card (NIC) driver level. This approach bypasses traditional kernel processing paths. This technology works as an add-on to nftables and combines accelerated packet processing with firewall features.

XDP speedups

Performance tests demonstrate that XDP helps provide significant speed improvements over standard `nftables` firewalls across different processor architectures.

We conducted performance tests that compare TBF XDP against a standard `nftables` firewall. The testing involved 3 different devices. Each device uses a different processor architecture to help provide performance data.

Device	CPUs	NFTables pps	TBF XDP pps	Speedup
TBF 100 (armhf)	2x Cortex v7	146 Kpps	775 Kpps	5.3
TBF 600 (arm64)	4x A72	594 Kpps	2840 Kpps	4.8
TBF 500 (x86)	4x Intel Atom C3558	659 Kpps	3192 Kpps	4.8

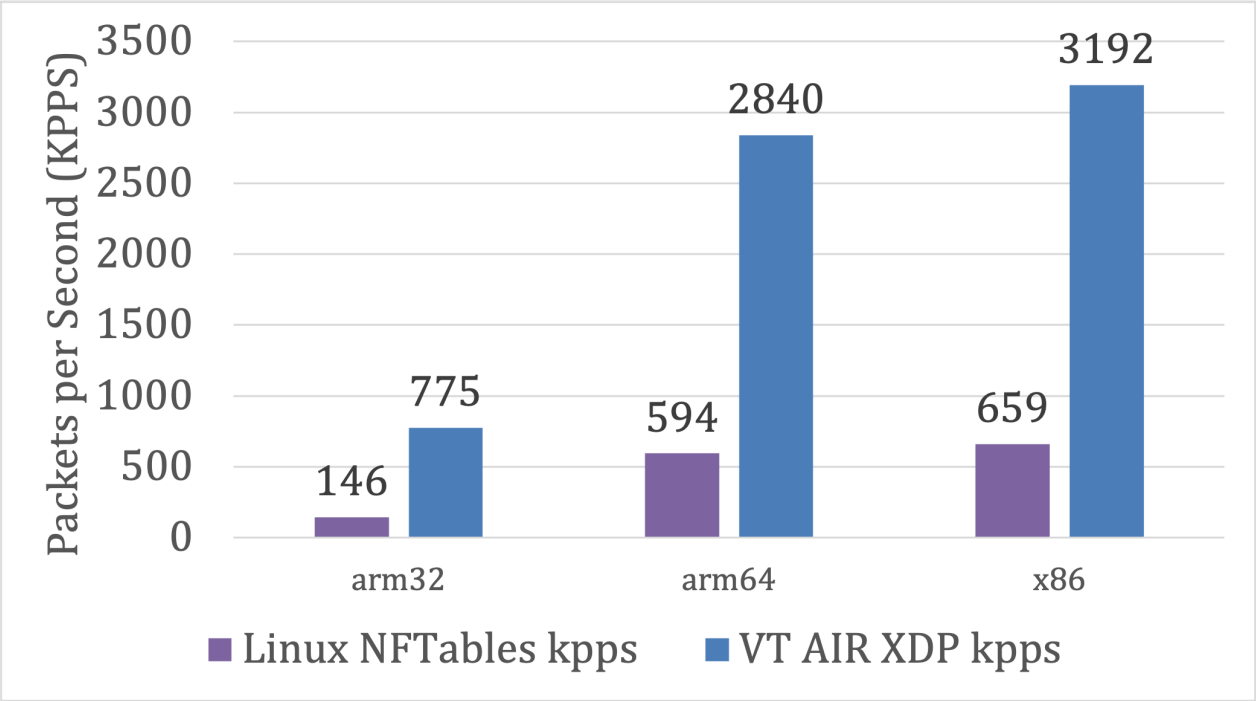


Figure 1: XDP speedup

XDP DDoS protection

XDP helps provide DDoS traffic blocking capabilities for networks that face increased DDoS attack risks.

XDP DDoS protection is especially valuable for networks that face increased DDoS attack risks. These networks include hosting environments, critical infrastructure, popular websites, and other high-value targets. Network operators can use the offloader to block DDoS attacks. This helps maintain network stability and helps prevent expensive downtime.

3. Appliances user manuals

Appliances help provide network security, VPN, and firewall capabilities across desktop, rack-mount, industrial, and cloud deployment options.

The appliance family delivers network security solutions with built-in firewall, VPN, and routing capabilities. Each appliance runs Linux and has a web-based management interface for configuration and monitoring.

Industrial appliances

These ruggedized appliances are designed for harsh industrial environments with extended temperature ranges and DIN rail mounting options.

- *IAF-240* on page 21: Industrial appliance with DIN rail mounting

Virtual appliances

These software-based appliances run on virtualization platforms and help provide flexible deployment options for various environments.

- *TBF-800V* on page 28: Virtual appliance for deployment on VirtualBox, Proxmox, and Hyper-V platforms

Features

Appliances include the following capabilities:

- Stateful packet inspection firewall
- VPN support (IPsec, OpenVPN, WireGuard)
- Network address translation (NAT)
- Quality of service (QoS) management
- Web-based management interface
- Command-line interface (CLI)
- SNMP monitoring support

IAF-240

Overview of IAF-240 device setup and configuration procedures.

IAF-240 delivers 360-degree cybersecurity, scalable defense in depth and deep visibility for OT networks. The device capabilities are geared towards applications in critical infrastructure and enterprise environments. It combines deep security features - such as IDS/IPS, application-level filtering, DDoS protection, and multi-factor authentication - with high-performance networking, including advanced routing, QoS, and VPN support. Its modular architecture supports both physical and virtual deployments, and centralized management is made simple with an intuitive Web GUI and REST API. The device helps ensure high availability, strong performance, and flexibility across a wide range of deployment scenarios.

Related information
Supported browsers on page 21
Hardware installation guide on page 22
Set up default login on page 22
Handle certificate warning on page 23
Use setup wizard on page 23
Licence activation on page 24

Supported browsers

Web browsers compatible with the IAF-240 device interface.

You can access the GUI using these web browsers:

- Google Chrome
- Microsoft Edge
- Mozilla Firefox
- Safari

Note: Internet Explorer is an unsupported browser and displays some GUI features incorrectly.

Hardware installation guide

Physical installation and setup documentation.

For detailed instructions on physical mounting, wiring, and power connections, refer to the comprehensive documentation available at *IAF-240*.

Set up default login

Set up the default login credentials for the GUI when you first access the system.

When you first access the system, you must set up the default login credentials for the GUI. The system requires you to create a password for the default user `admin`.

To access the system through SSH or console, use the `root` user account. Both the `root` and `admin` users share the same password. This means you can only log in through SSH after you create the default password.

The system comes with an initial password: `private`. You can access the GUI by navigating to the default IP address *https://192.168.1.1*.

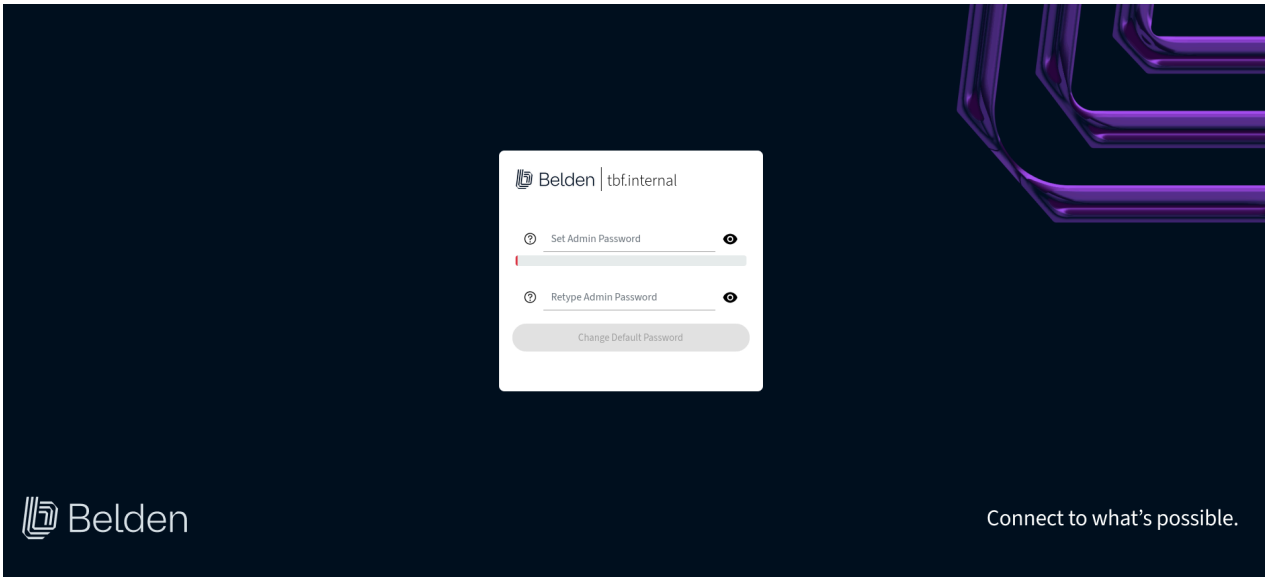


Figure 2: Initial password change

Perform the following steps:

1. Enter a new admin password.
2. Confirm the new admin password by entering it again.
3. Select **Change Default Password**.

Note: If you lose your password, reset the device to factory settings using the CLI or the **Reset** button.

Handle certificate warning

Handle the certificate warning that appears when accessing the device via a supported browser.

When you access the device through a supported browser, a certificate warning appears. This warning occurs because the device uses a self-signed certificate for security.

Note: The following example shows Firefox. The steps may vary slightly depending on the browser.

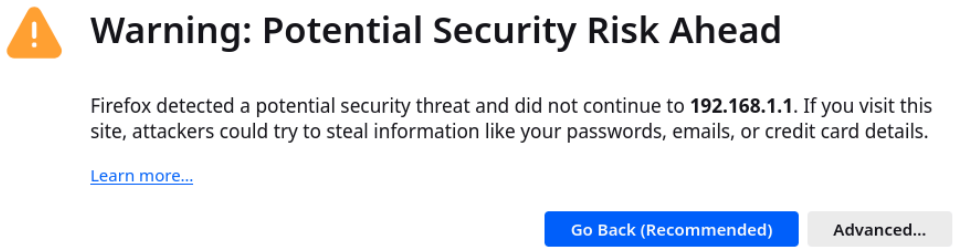


Figure 3: Firefox certificate warning

Perform the following steps:

1. Select **Advanced** to view additional details.
2. Select **Proceed to 192.168.1.1** to continue to the device interface.

Use setup wizard

Use the setup wizard that appears when you log in to the GUI for the first time.

When you log in to the GUI for the first time, the setup wizard automatically appears to help you set up initial settings.

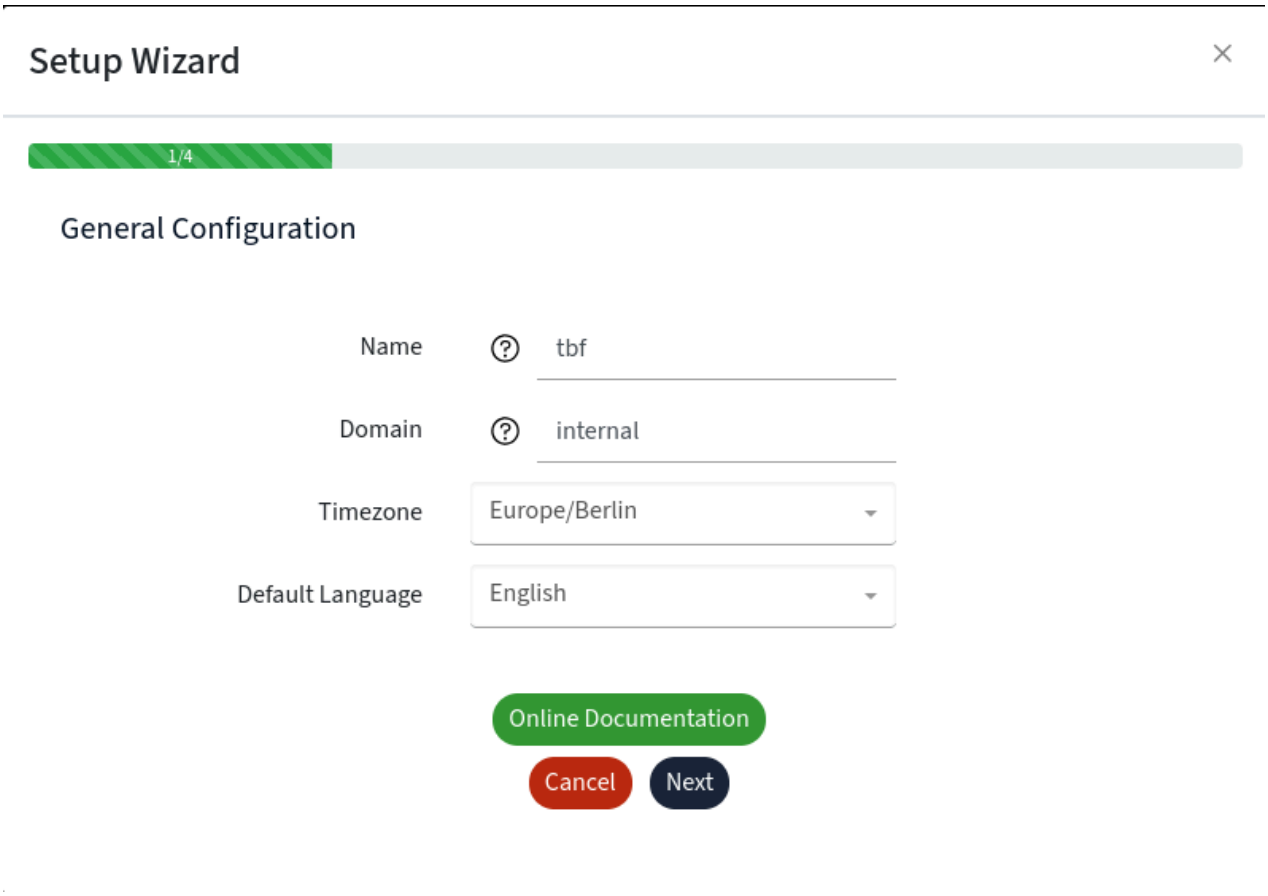


Figure 4: Setup wizard

Perform the following steps:

1. Set up the settings as needed for each step of the wizard.
2. Select **Next** to proceed to the next step.
3. When you reach the final step of the setup wizard, select **Finish** to complete the setup.

For detailed information about individual settings, see *Set up interfaces* on page 92.

Licence activation

Activate your TBF licence to unlock all features and ensure compliance with your entitlement.

TBF requires a valid licence to operate. You can validate your licence online or offline depending on your network connectivity.

The licence activation process involves:

- Validating your licence through online or offline methods
- Resolving any licence validation issues
- Accessing the Belden Portal when required
- Entering your entitlement ID (EID) correctly

Choose the appropriate validation method based on your network setup and requirements.

Licence validation overview

Overview of TBF licence validation methods and requirements.

TBF supports two licence validation methods:

Online validation

Validates your licence directly through the internet. This method requires TBF to have internet access and is the fastest validation option.

Offline validation

Validates your licence without internet access. This method requires manual steps including downloading a licence file from the Sentinel EMS portal.

Both methods require your entitlement ID (EID) and access to the Belden Portal for authentication.

Important: Ensure you have your EID available before starting the validation process. The EID is provided with your TBF purchase documentation.

Validate licence online

Validate a TBF licence online when the device has internet access.

Prerequisites:

TBF must have access to the internet for online activation. If it does not have access, see *Validate licence offline* on page 26 instead.

Take the following steps to validate a TBF licence online.

Perform the following steps:

1. Complete the Belden Portal login procedure.

   See *Belden Portal login* on page 27 for detailed steps.

2. Go to **System > Licence**.

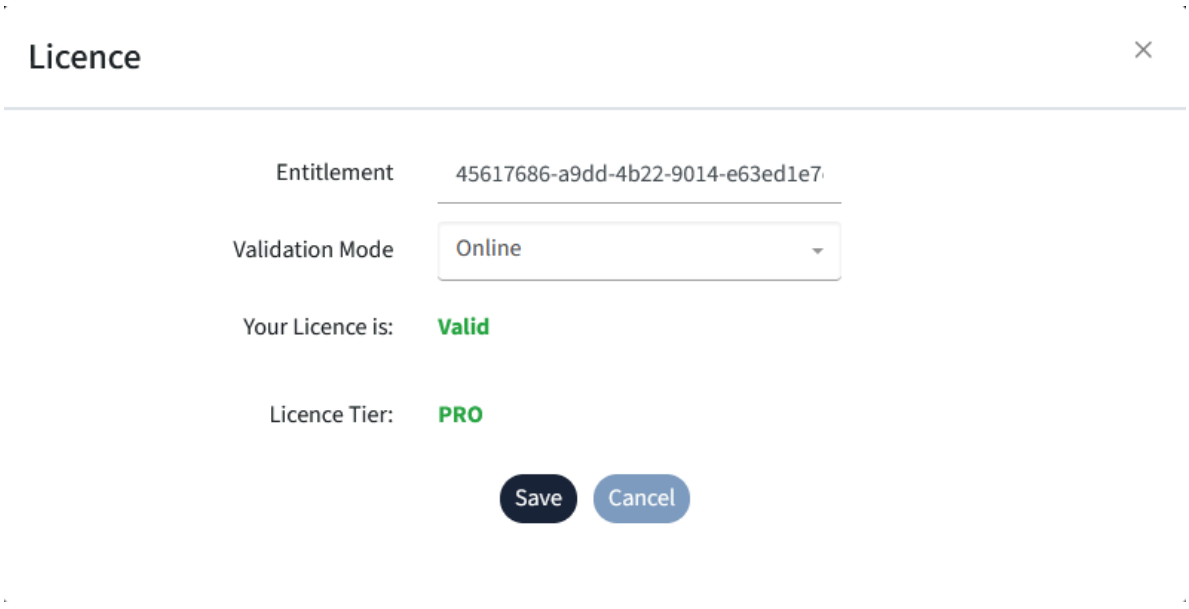


Figure 5: Online activation

3. For **Entitlement**, enter the EID.

   If the EID is not automatically displayed, you must enter it manually.

4. For **Validation Mode**, select **Online**.

5. Select **Save** to continue.

The online activation process starts. If it was successful, you see *Valid* and the licence tier.

Validate licence offline

Validate a TBF licence offline when the device does not have internet access.

Prerequisites:

You need access to a computer with internet connectivity to complete the offline validation process.

Use offline validation when TBF cannot access the internet directly. This process requires downloading a licence file from the Sentinel EMS portal.

Perform the following steps:

1. Retrieve the device ID from TBF.
    a. Go to **Dashboard**.
    b. In the **System Information** widget, locate the **Device ID**.
    c. Copy the Device ID for use in the Sentinel EMS portal.
2. Go to **System > Licence**.

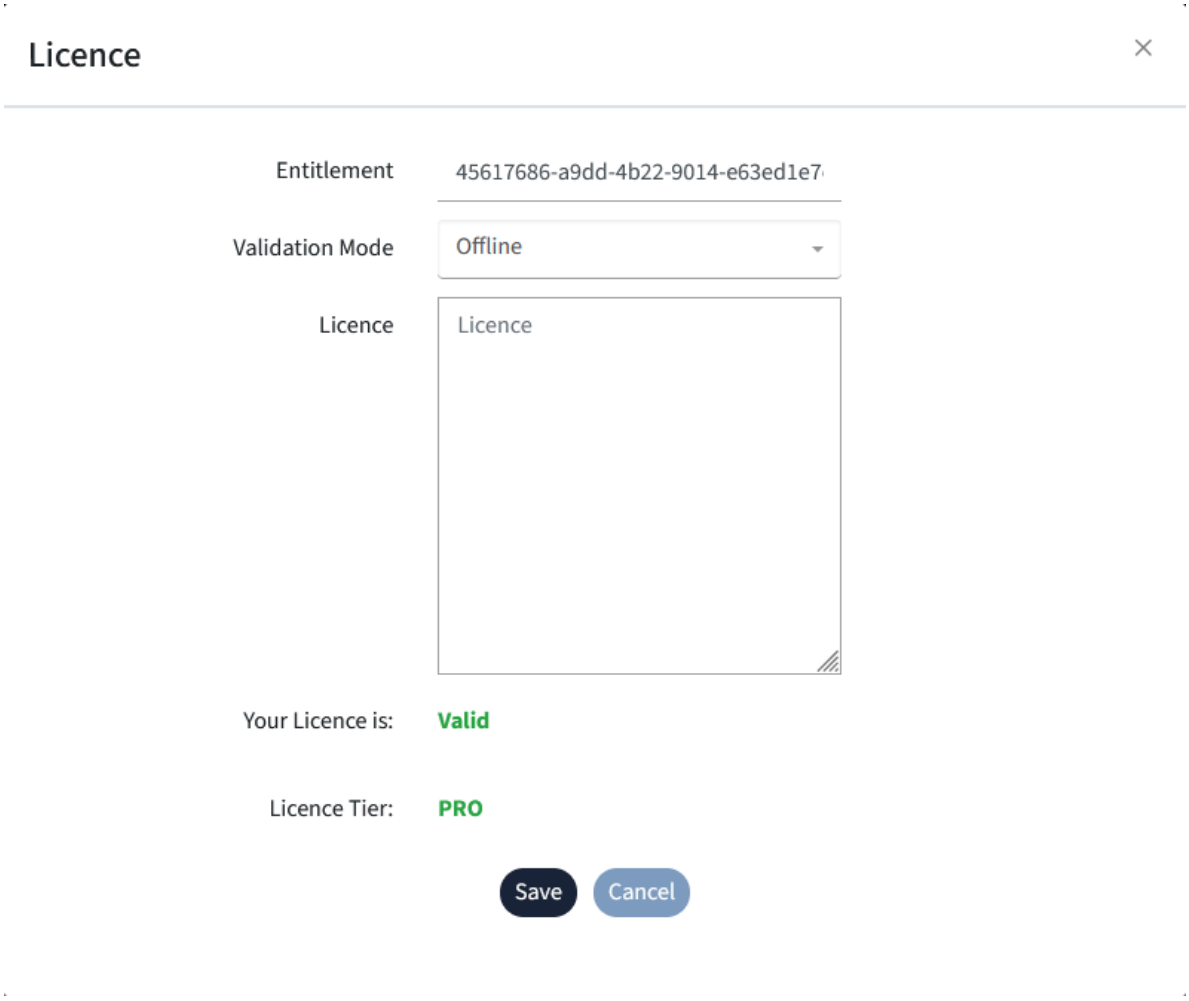


Figure 6: Offline activation

3. For **Entitlement**, enter the EID.

    If the EID is not automatically displayed, you must enter it manually.
4. For **Validation Mode**, select **Offline**.
5. Complete the Sentinel EMS activation process.

a. Go to the *Sentinel EMS portal*.
b. Log in with your credentials.
c. Navigate to the licence activation section.
d. Enter your EID and the Device ID you copied earlier.
e. Download the generated licence file.

6. Install the licence file in TBF.
   a. Return to the TBF **System > Licence** page.
   b. Select **Choose File** and upload the licence file you downloaded.
   c. Select **Save** to install the licence.

The offline activation process completes. If successful, you see *Valid* and the licence tier.

Resolve invalid licence

Troubleshoot and resolve issues when TBF licence validation fails.

Your TBF licence shows as invalid or the validation process fails.

Licence validation can fail for several reasons:

- Incorrect EID entry
- Network connectivity issues during online validation
- Expired or inactive licence
- Incorrect device ID for offline validation
- Corrupted licence file

Perform the following steps:

1. Verify the EID is entered correctly.

   Check your purchase documentation for the correct EID format and characters.

2. For online validation, check network connectivity.

   Ensure TBF can access the internet and that firewall rules allow outbound connections to the licence server.

3. For offline validation, verify the device ID.

   Ensure you copied the correct Device ID from the Dashboard and entered it accurately in the Sentinel EMS portal.

4. Check licence status in the Belden Portal.

   Log in to verify your licence is active and not expired.

5. If using offline validation, download a fresh licence file.

   The licence file may have become corrupted during download or transfer.

6. Contact Belden support if the issue persists.

   Provide your EID, device ID, and any error messages for faster resolution.

Belden Portal login

Access the Belden Portal to manage licence entitlements and activations.

Portal access

The Belden Portal provides access to licence management and support resources. You need valid credentials to access the portal.

Login procedure

1. Go to the Belden Portal website.
2. Enter your username and password.
3. Select **Login** to access your account.
4. Navigate to the licence management section for TBF-related tasks.

Account requirements

You must have a valid Belden Portal account associated with your TBF purchase. If you do not have an account or cannot access it, contact Belden support for assistance.

EID entry instructions

Instructions for entering the Equipment ID (EID) during licence validation.

EID entry

For **Entitlement**, enter the EID.

If the EID is not automatically displayed, you must enter it manually.

EID format

The EID is a unique identifier provided with your TBF purchase documentation. Ensure you enter it exactly as provided, including any hyphens or special characters.

TBF-800V

Installation guide for TBF-800V virtual appliance on various virtualization platforms.

TBF-800V is a virtual appliance that you can deploy on multiple virtualization platforms. This guide walks you through the installation process for a specific virtualization environment.

Installation process

Complete these steps in order to successfully install and set up a TBF-800V virtual appliance:

1. *Review system requirements* on page 28: Verify that the system meets the software and hardware requirements before starting the installation
2. Select the virtualization platform and follow the corresponding setup guide:
    - *Install TBF-800V on VirtualBox* on page 29
    - *Install TBF-800V on Proxmox* on page 33
    - *Install TBF-800V on Hyper-V* on page 37
3. *Install TBF-800V system* on page 42: Complete the operating system installation and initial setup
4. *Access TBF-800V web UI* on page 46: Connect to the web interface to begin setting up the appliance
5. *Licence activation* on page 24: Activate the TBF licence

TBF-800V requirements

These are the software and hardware requirements for the TBF-800V virtual appliance installation.

Software requirements

To run the TBF-800V virtual appliance, verify that the virtualization platform meets these minimum software requirements:

- **VirtualBox version 7.0 or higher**: *Download VirtualBox*
- **Proxmox Virtual Environment 8.4.14**: *Download Proxmox*
- **Hyper-V**: *Hyper-V documentation*
- **Latest TBF-800V installer**: `tbf-image-tbf800-installer-tbf-amd64-1.3.0-FINAL.3.iso`

Hardware requirements

To support the virtual appliance, verify that the host system meets these minimum hardware specifications:

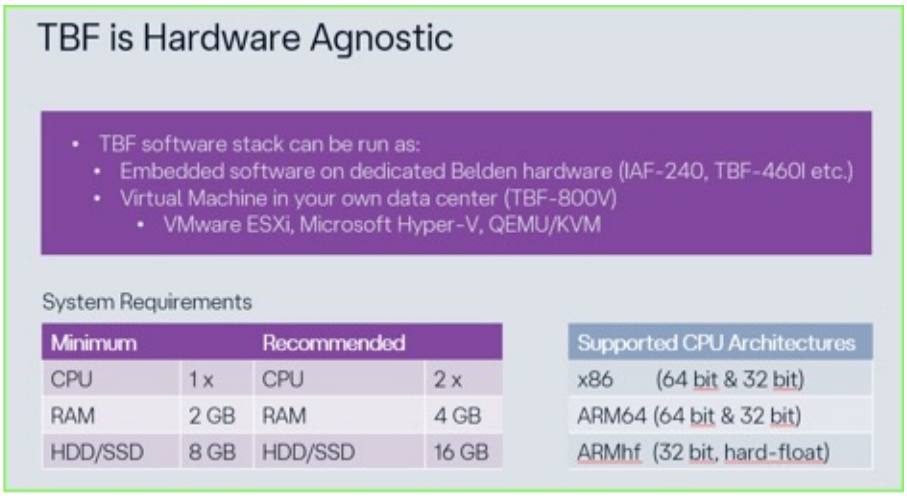


Figure 7: System requirements

Install TBF-800V on VirtualBox

This task describes how to create and set up a VirtualBox virtual machine (VM) for the TBF-800V installation.

Prerequisites:

Before you begin, verify that VirtualBox version 7.0 or higher is installed on the system and that the TBF-800V installer ISO file is available on the computer.

Perform the following steps:

1. Create a new VM with these basic settings:
    a. Enter a descriptive **Name** for the VM.
    b. Set the **Type** to **Linux**.
    c. Set the **Version** to **Other Linux (64-bit)**.

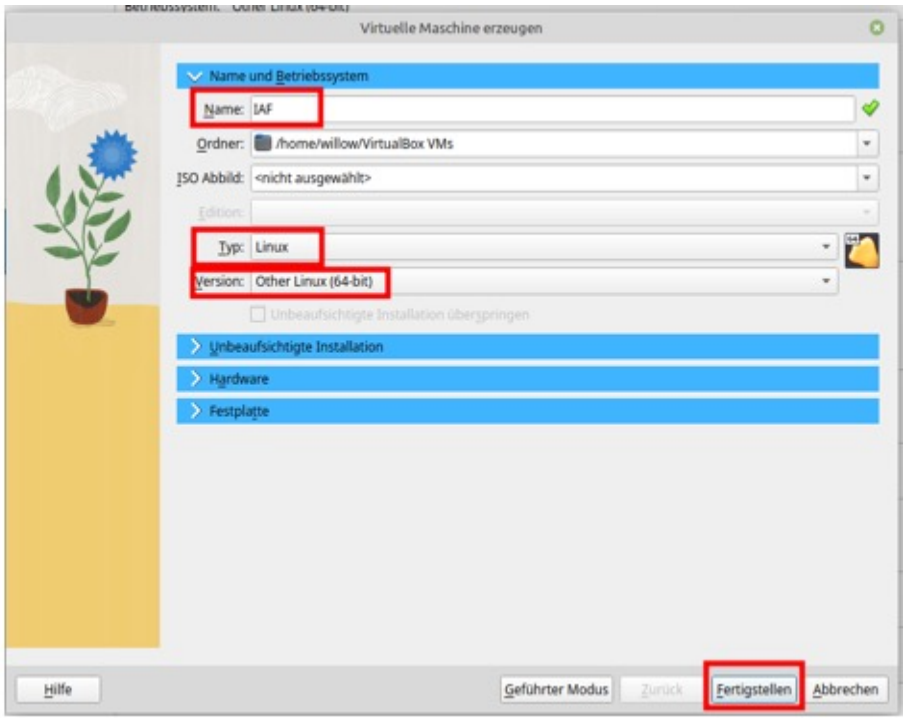


Figure 8: VirtualBox create VM

2. Set up the system hardware on the **Motherboard** tab:
    a. Allocate 2048 MB (2 GB) of RAM for **Base Memory**.
    b. Arrange the **Boot Order** as follows: **Hard disk** first, **Optical** second, and **Floppy** third.
    c. To enable EFI, select the **EFI** option.

3. Assign processor resources:
    a. Set the number of **Processors** to **2** CPUs.

4. Set up the storage configuration:

a. Add an optical drive to the storage devices.

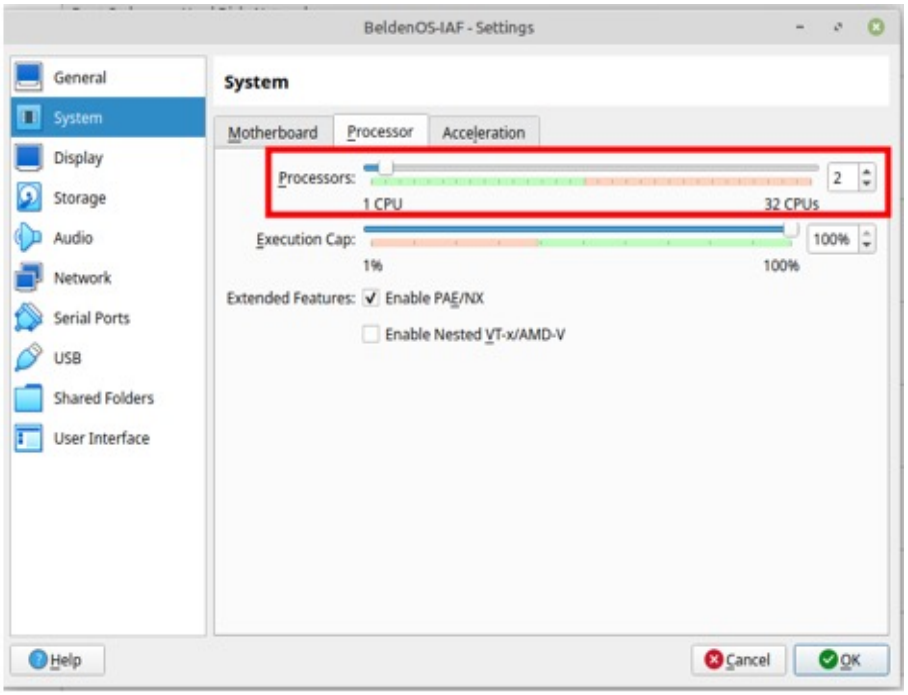


Figure 9: VirtualBox system

b. Mount the latest IAF Belden OS amd64 ISO file. Verify that the ISO file is added as the first boot device.

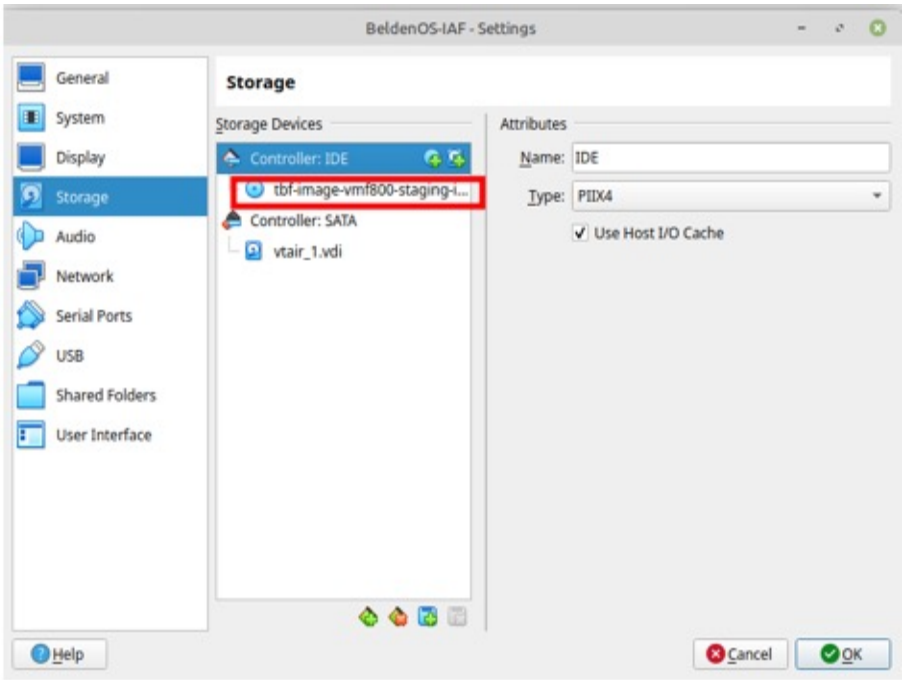


Figure 10: VirtualBox storage

5. Set up the network connections:

a. Create two network interfaces for the VM.

b. Set up adapter 1 as a **Bridged Adapter**. This adapter serves as the WAN connection. This adapter uses DHCP client configuration by default.

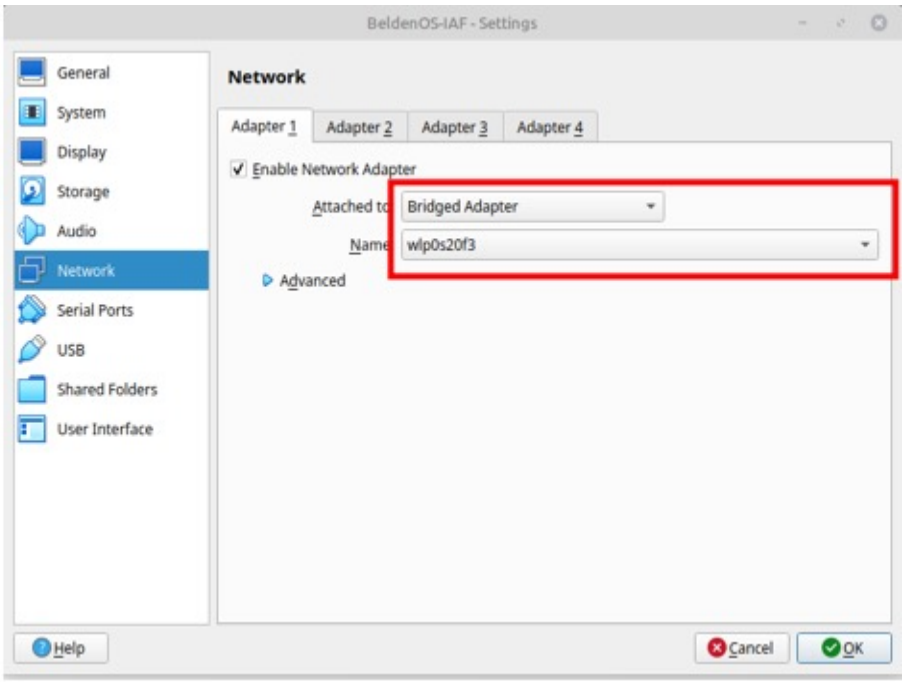


Figure 11: VirtualBox network

c. Set up adapter 2 as **NAT**. This adapter serves as the LAN connection.

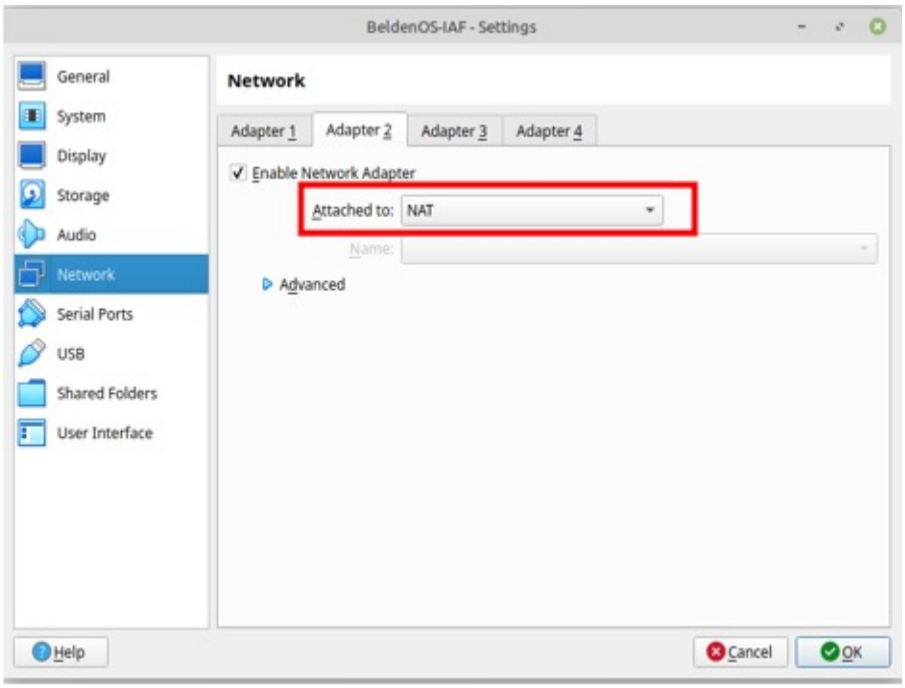


Figure 12: VirtualBox network 2

Leave all other settings at their default values. The VirtualBox VM is now properly set up and ready for the TBF-800V installation.

Install TBF-800V on Proxmox

This task describes how to create and set up a Proxmox virtual machine (VM) for the TBF-800V installation.

Prerequisites:

Before you begin, verify that Proxmox Virtual Environment 8.4.14 or higher is installed and that the TBF-800V installer ISO file is uploaded to the Proxmox storage.

Perform the following steps:

1. Create a new VM. On the **General** tab:
    a. To access additional configuration options, select the **Advanced** checkbox.
    b. Set **Node** to **proxmox**.
    c. Set **VM ID** to **106**.
    d. Set **Name** to **TBF-800-1.1.2**.
    e. Leave the **Start at boot** option unchecked. This setting helps prevent the VM from starting automatically when the Proxmox host boots.

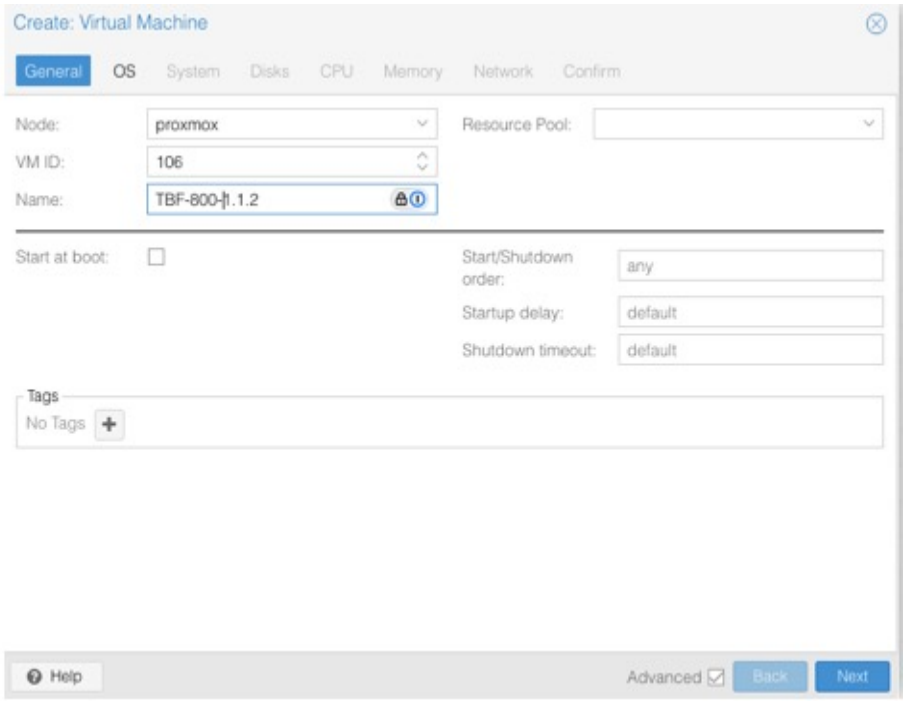


Figure 13: Proxmox create VM

2. Set up the operating system on the **OS** tab:
    a. To boot from the installer ISO, select **Use CD/DVD disc image file (iso)**.
    b. For **Storage**, select **Backup**.
    c. For **ISO image**, select **tbf-image-tbf800-installer-tbf-amd64-1.1.2-FINAL.1.iso**.
    d. Under **Guest OS**, set **Type** to **Linux** and **Version** to **6.x-2.6 Kernel**.

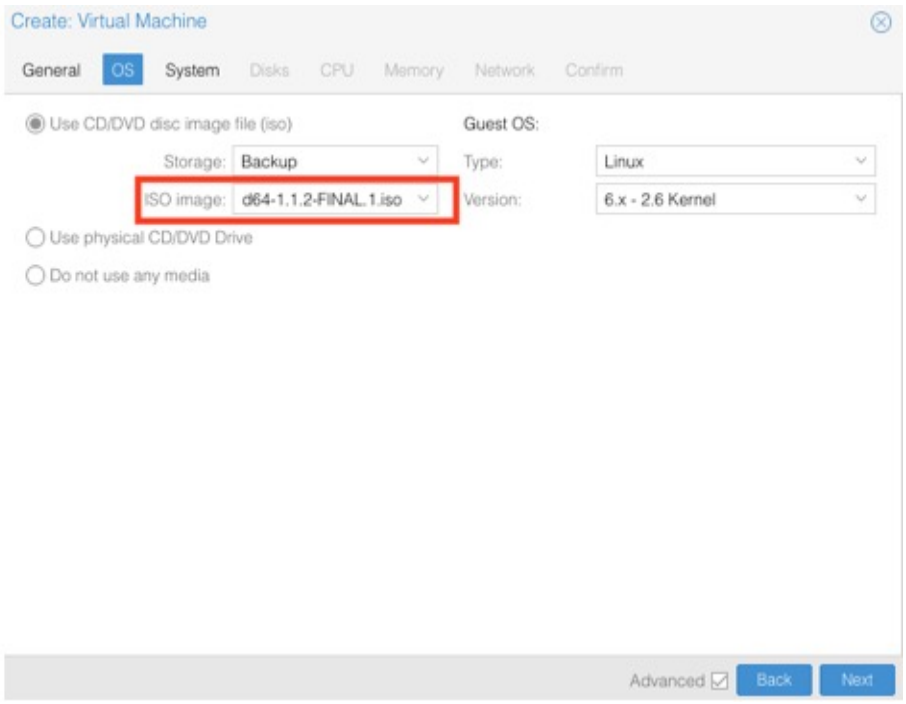


Figure 14: Proxmox OS

3. Set up system hardware on the **System** tab:
    a. For **Machine**, select **Default (i440fx)**.
    b. For **SCSI Controller**, select **VirtIO SCSI single**.
    c. To enable modern UEFI boot, select **OVMF (UEFI)** for **BIOS**.
    d. To create the required EFI system partition, select **Add EFI Disk**.
    e. To help avoid potential boot issues, leave **Pre-Enroll keys** unchecked.

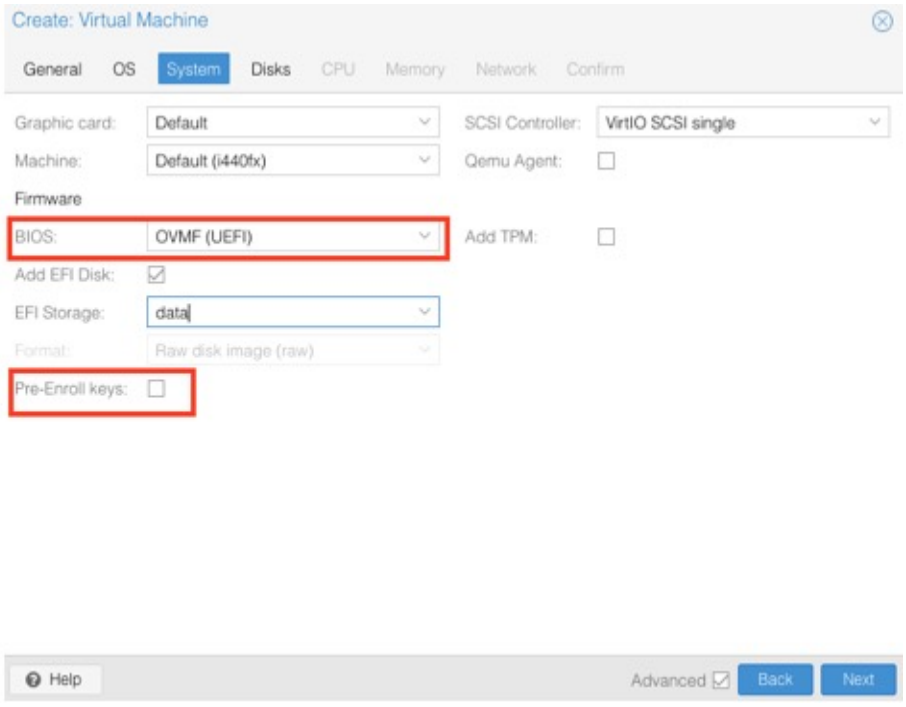


Figure 15: Proxmox system

4. Set up storage on the **Disks** tab:

a. For **BUS/Device**, select **SCSI0**.
b. For **Disk size (GiB)**, select **32**. This size helps provide sufficient space for the TBF-800V installation.

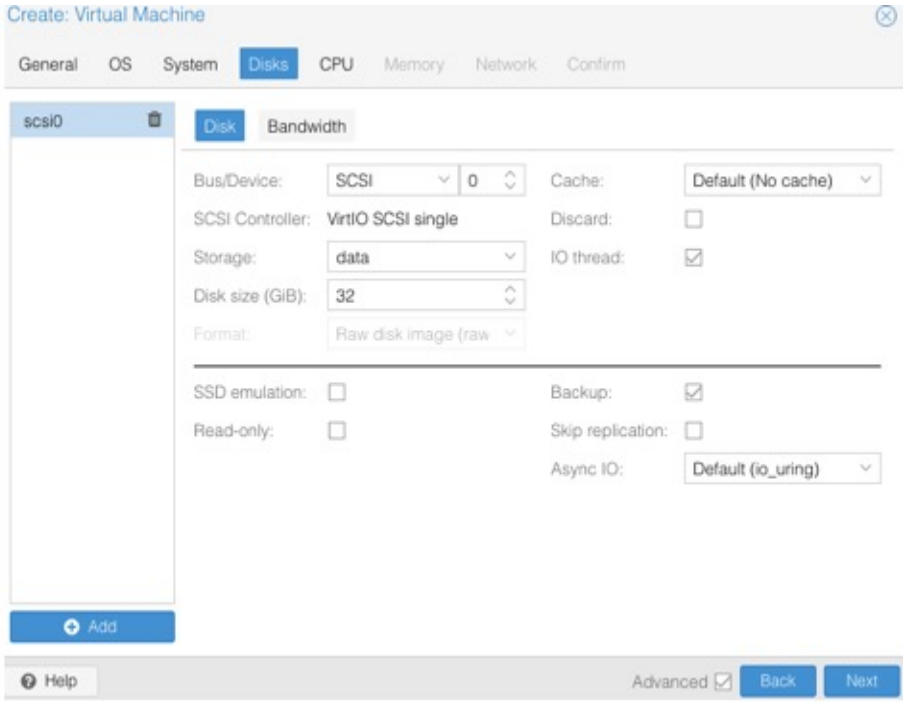


Figure 16: Proxmox disks

5. Allocate CPU resources on the **CPU** tab:
   a. For **Sockets**, select **1**.
   b. For **Cores**, select **2**. This configuration helps provide adequate processing power for TBF-800V operations.
   c. To pass through the host CPU features for optimal performance, select **host** for **Type**.

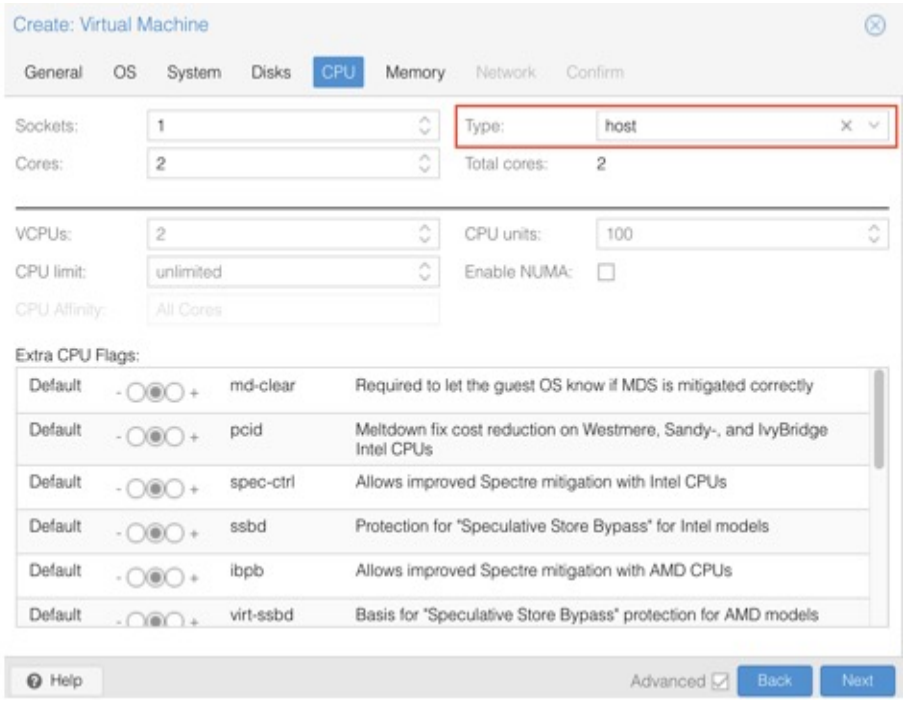


Figure 17: Proxmox CPU

6. Allocate memory on the **Memory** tab:

a. Set **Memory (MiB)** to **4096** (4 GB). This amount helps ensure smooth operation of the TBF-800V system.

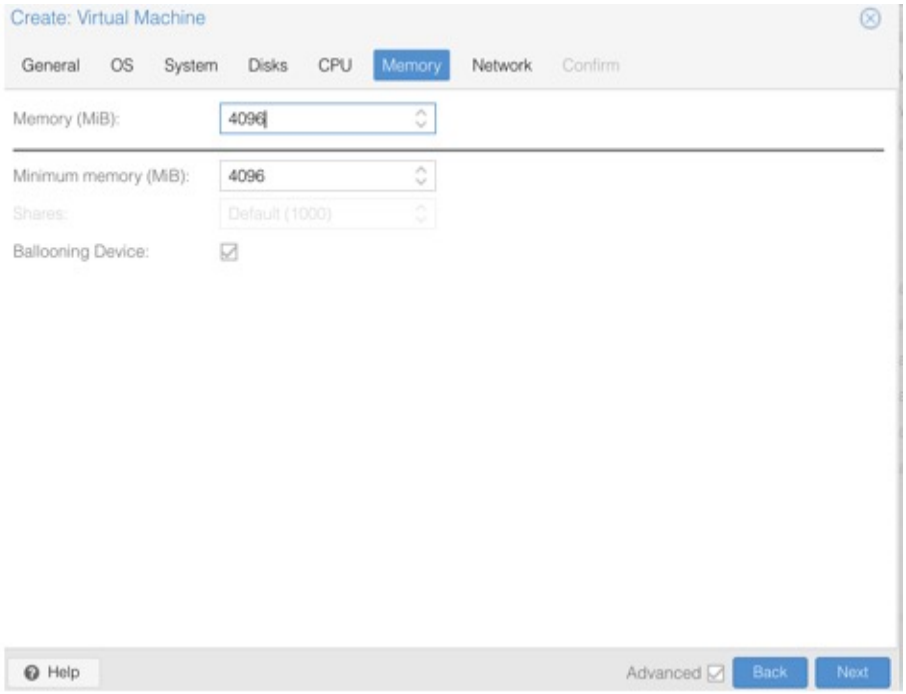


Figure 18: Proxmox memory

7. Set up the first network interface on the **Network** tab:
   a. For **Bridge**, select **vmbr5**.
   b. For better network performance, select **VirtIO (paravirtualized)** for **Model**.
   c. For **VLAN Tag**, leave as **No VLAN**.
   d. To enable the Proxmox firewall, select the checkbox for **Firewall**.
   e. To use the standard Ethernet frame size, select **1500 (1 = bridge MTU)** for **MTU**.

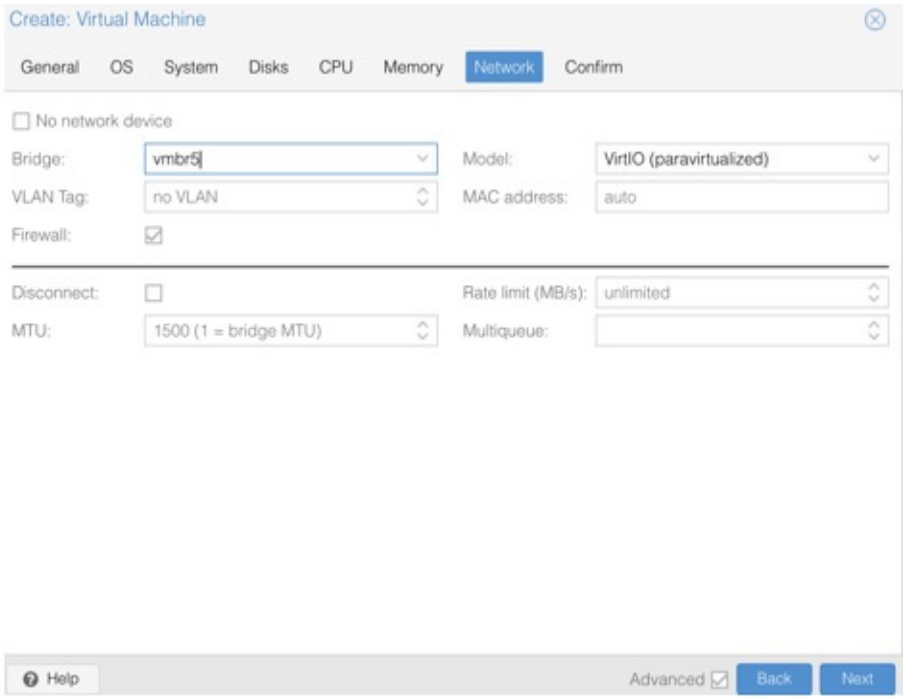


Figure 19: Proxmox network

8. Before creating the VM, review the configuration summary to verify that all settings are correct.

9. For additional connectivity, add a second network interface:
    a. Navigate to the **VM Hardware > Add** menu.
    b. To add a network device, select the option.
    c. For **Bridge**, select **vmbr6**.
    d. For **Model**, select **VirtIO (paravirtualized)**.
    e. For **VLAN Tag**, leave as **No VLAN**.
    f. For **Firewall**, select the checkbox.
    g. For **MTU**, select **1500 (1 = bridge MTU)**.

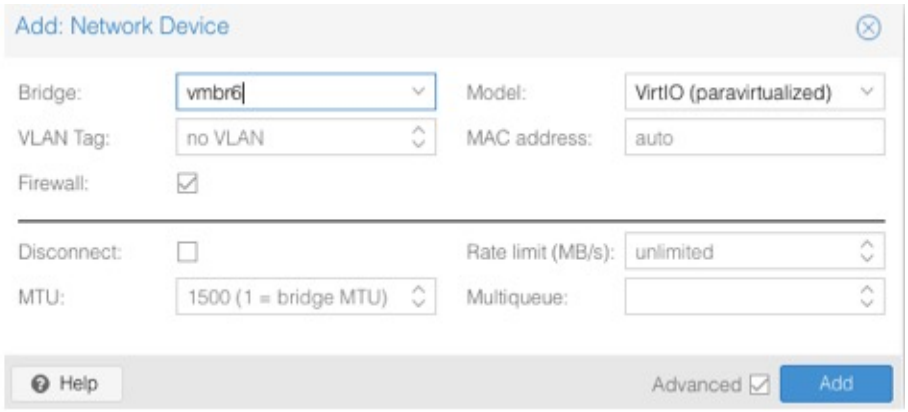


Figure 20: Proxmox add network device

Proxmox displays the complete VM configuration showing all hardware components and settings.



Figure 21: Proxmox configuration

To help avoid boot compatibility issues, if Proxmox enables secure boot by default during VM creation, disable the secure boot before proceeding with the TBF-800V installation.

Install TBF-800V on Hyper-V

This task describes how to create and set up a Hyper-V virtual machine (VM) for the TBF-800V installation.

Prerequisites:

Before you begin, verify that Hyper-V is enabled on the Windows system and that you have the TBF-800V installer ISO file ready.

Perform the following steps:

1. To open Hyper-V Manager, press the Windows key and type `Hyper-V Manager`.

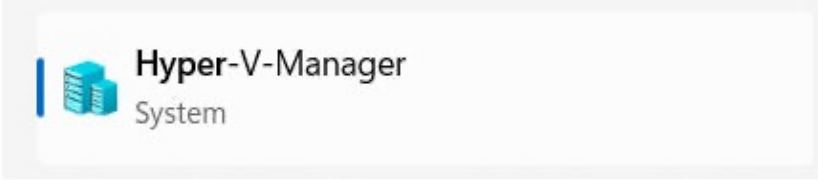


Figure 22: Hyper-V manager

2. To start the VM creation wizard, select **Quick create**.

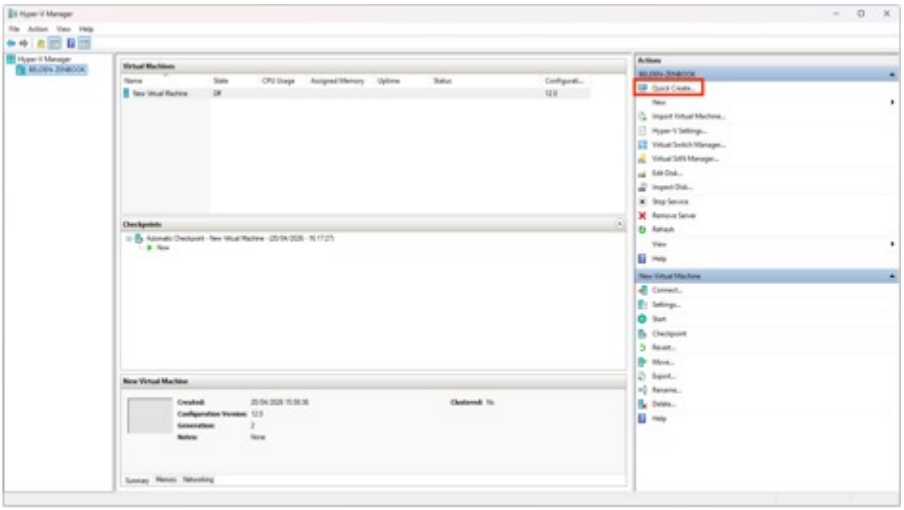


Figure 23: Hyper-V create VM

3. Set up the VM creation options:
    a. Select **Local Installation Source**.
    b. Enter a **Name** for the VM. This step is optional but recommended for easy identification.
    c. To browse for the ISO file, select **Change installation source**.

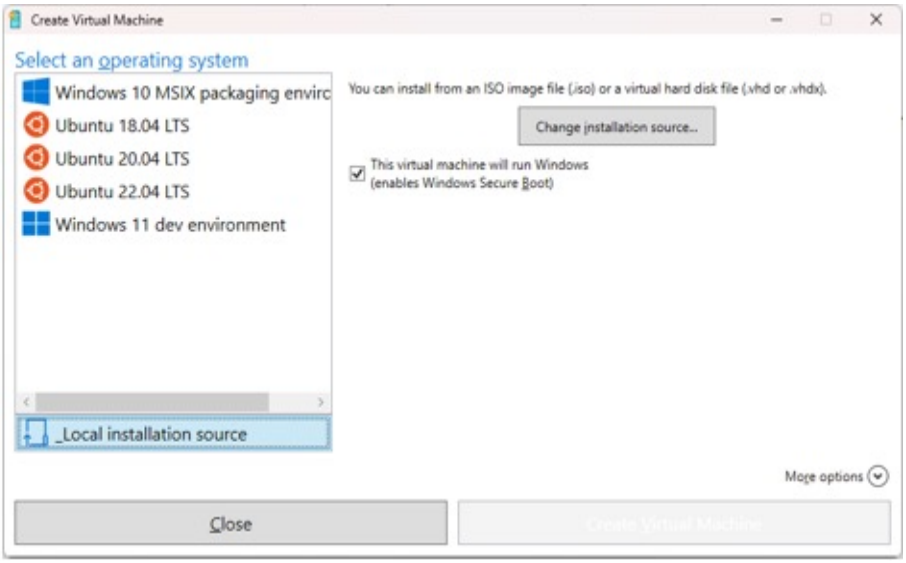


Figure 24: Hyper-V select OS

4. Browse and select the installer ISO file:
   a. Navigate to and select the **tbf-image-tbf800-installer-tbf-amd64-1.1.2-FINAL.1.iso** file.
   b. To proceed, select **Create Virtual Machine**.

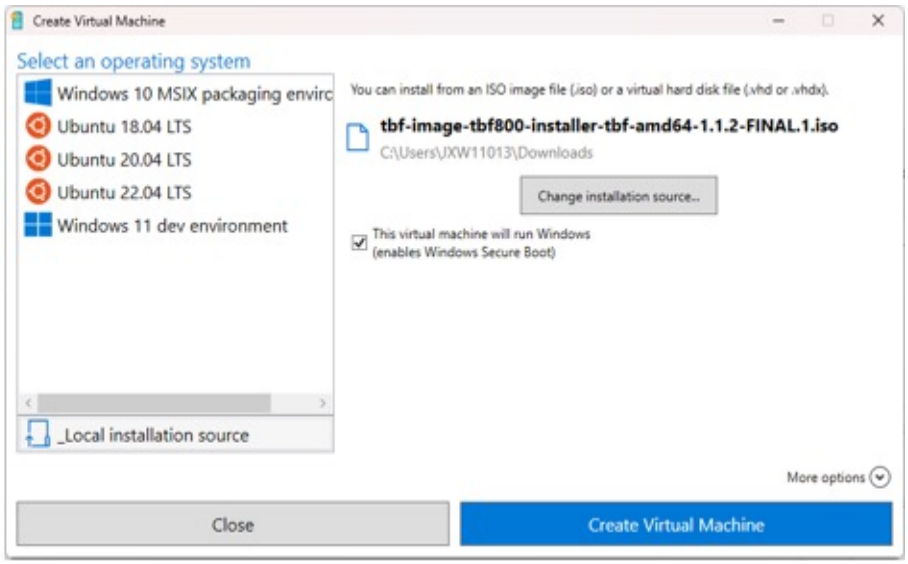


Figure 25: Hyper-V ISO

5. For better performance, adjust the VM settings:
   For an embedded device experience, set up the CPU to 2 cores and RAM to 2048 MB. You can adjust these values based on the system resources. To operate the device correctly, disable the secure boot.

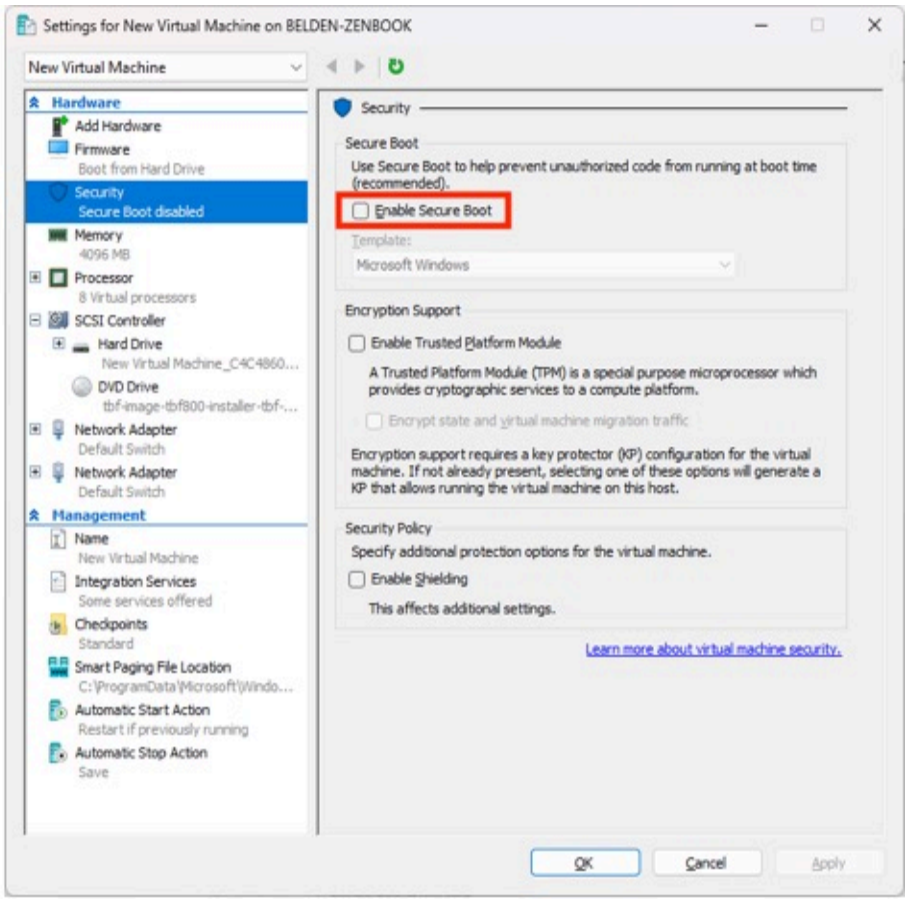


Figure 26: Hyper-V security

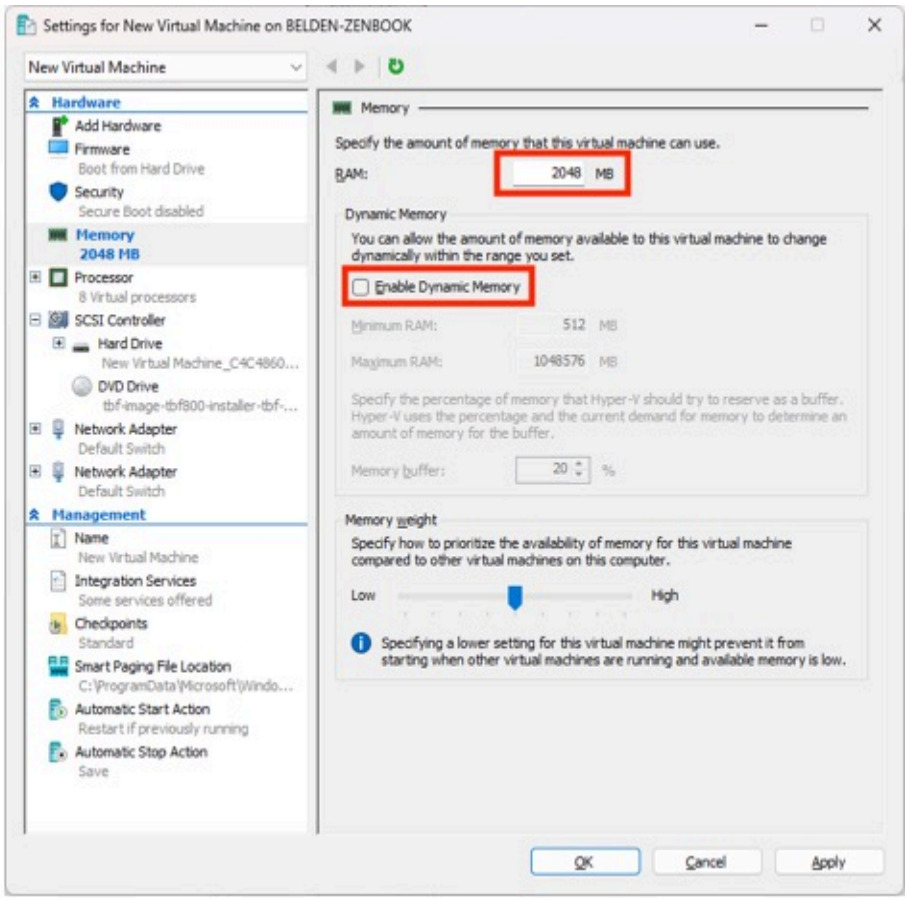


Figure 27: Hyper-V memory

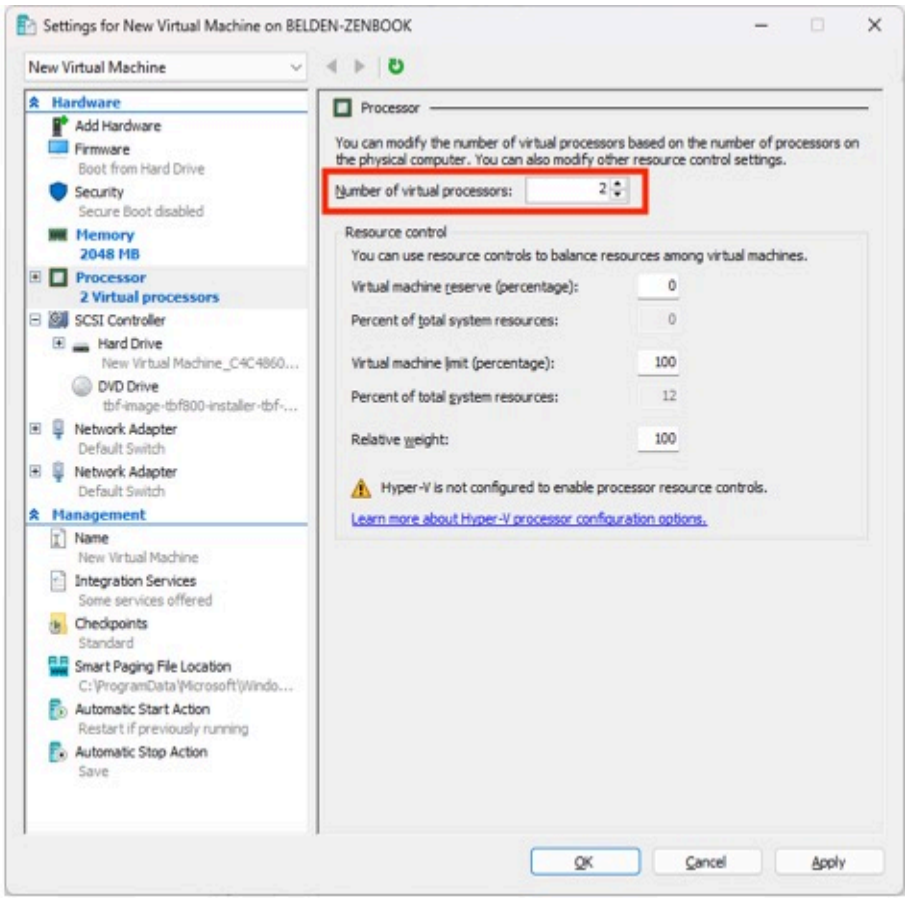


Figure 28: Hyper-V processor

6. Save the configuration and connect to the VM:

a. To save the settings, select **Apply**, then select **OK**.

b. To open the VM console, select **Connect**.

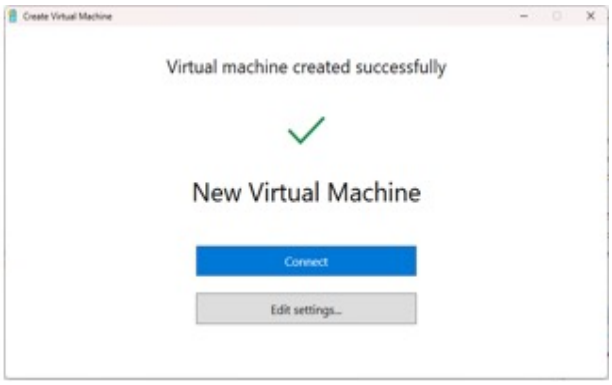


Figure 29: Hyper-V creation successful

7. To begin the installation process, power on the VM.

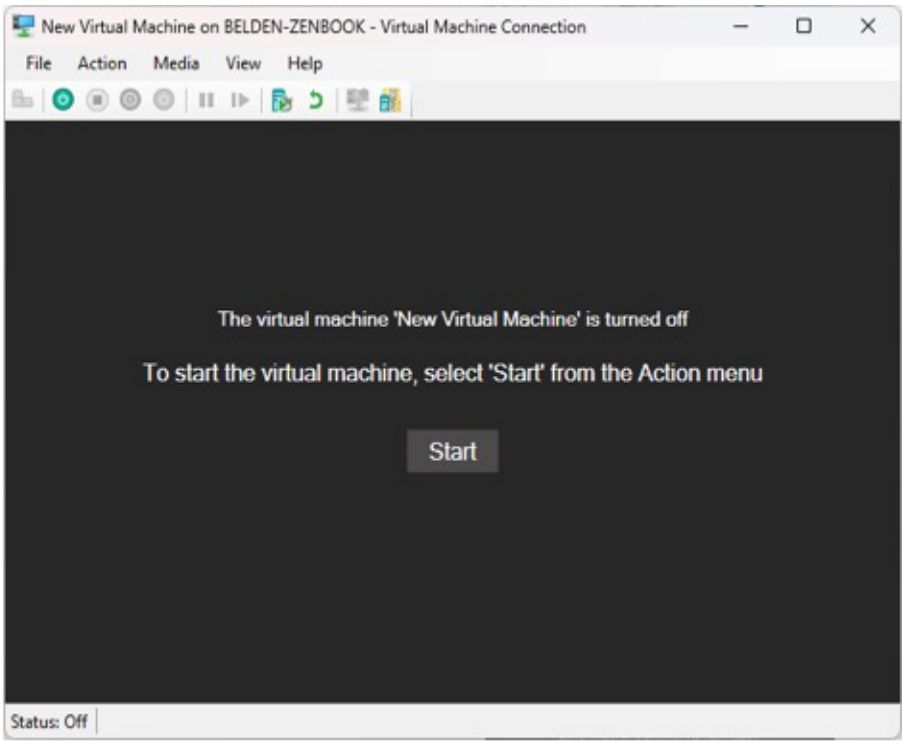


Figure 30: Hyper-V start

The Hyper-V VM is now set up and ready for the TBF-800V installation.

Install TBF-800V system

This task describes how to complete the TBF-800V operating system installation process on the virtual machine (VM).

Prerequisites:

Before you begin, verify that you have set up a VM (VirtualBox, Proxmox, or Hyper-V) and disabled secure boot if required.

Perform the following steps:

1. Start the VM from the installer ISO.

   After you disable secure boot (if needed), UEFI loads the installer ISO from the **QEMU DVD-ROM**. A boot progress bar shows that the system is loading the installation environment.



   Figure 31: Proxmox start

2. Allow the installation to complete.

   The **Installation Status** dialog shows *Installation In Progress...*, which means the operating system is installing on the VM.

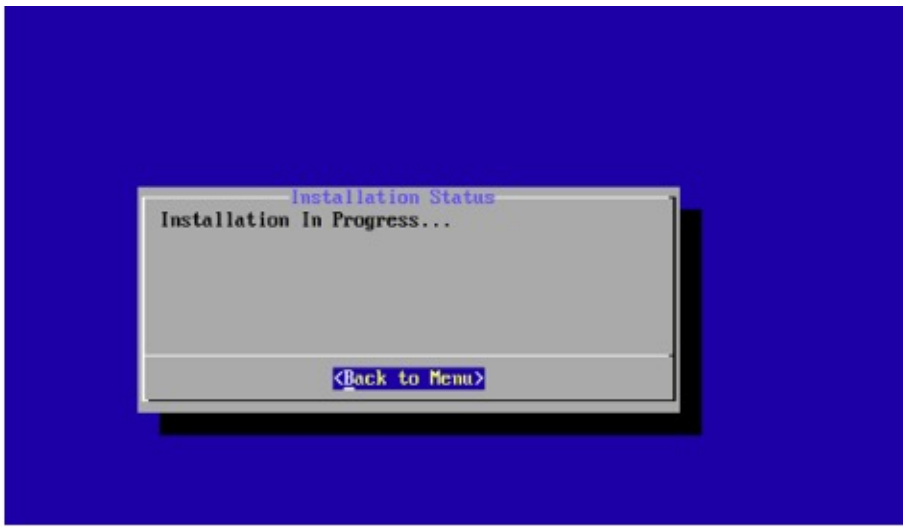


   Figure 32: Installation status

3. Verify that the installation is complete.

   When the installation finishes, the **Installation Status** dialog shows *Installation Done*.

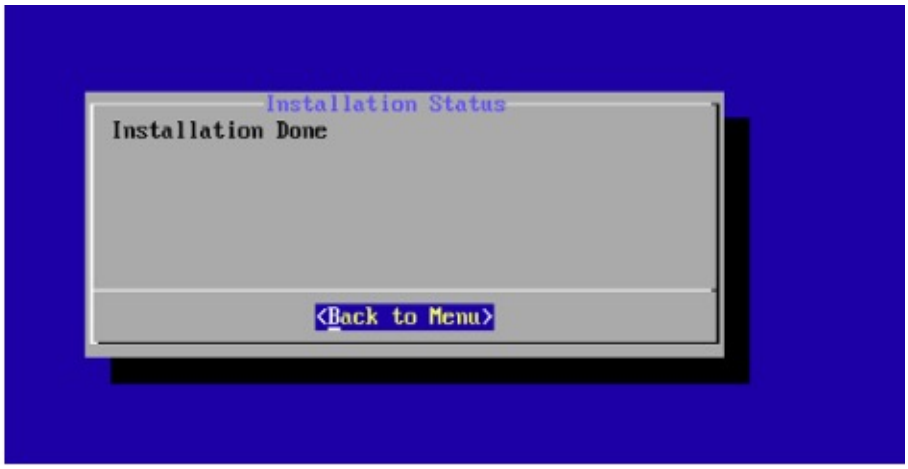


Figure 33: Installation done

4. Restart the system.
   The **Installation Menu** appears with option **3 Reboot System** highlighted. To restart the VM and boot into the newly installed system, select this option.

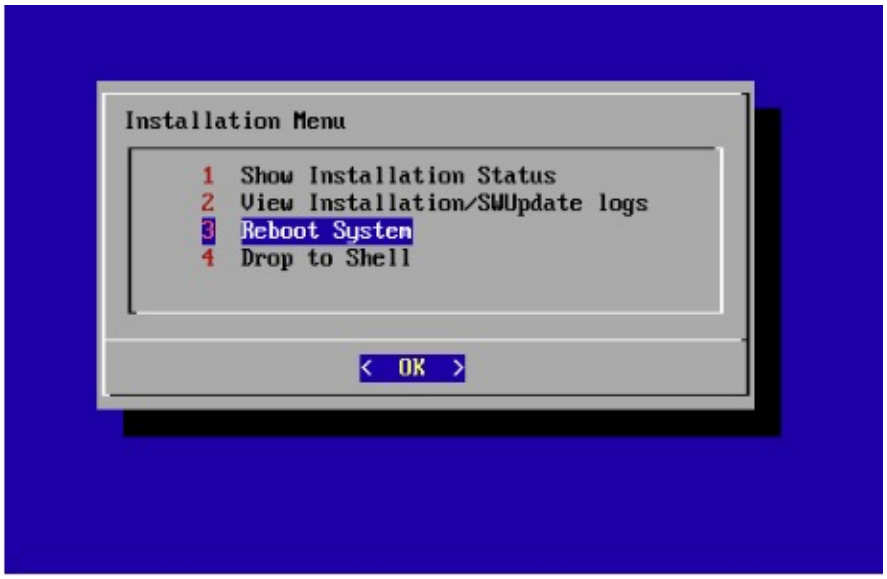


Figure 34: Reboot system

5. Access the system.

    a. After the system restarts, verify that it boots into the operating system and shows a **Password** prompt.



Figure 35: Password prompt

    b. Type the default password `private`.

After you log in, the TBF CLI shell becomes active and shows version *26.01.1-r0* for model TBF-800V.

The system shows all available management commands and displays the command prompt `(tbf)#`. This means the system is ready for setup and management tasks.



Figure 36: TBF CLI

Allow at least 2 to 3 minutes after startup for the system to become fully available.

Set up default login

Set up the default login credentials for the GUI when you first access the system.

When you first access the system, you must set up the default login credentials for the GUI. The system requires you to create a password for the default user `admin`.

To access the system through SSH or console, use the `root` user account. Both the `root` and `admin` users share the same password. This means you can only log in through SSH after you create the default password.

The system comes with an initial password: `private`. You can access the GUI by navigating to the default IP address *https://192.168.1.1*.

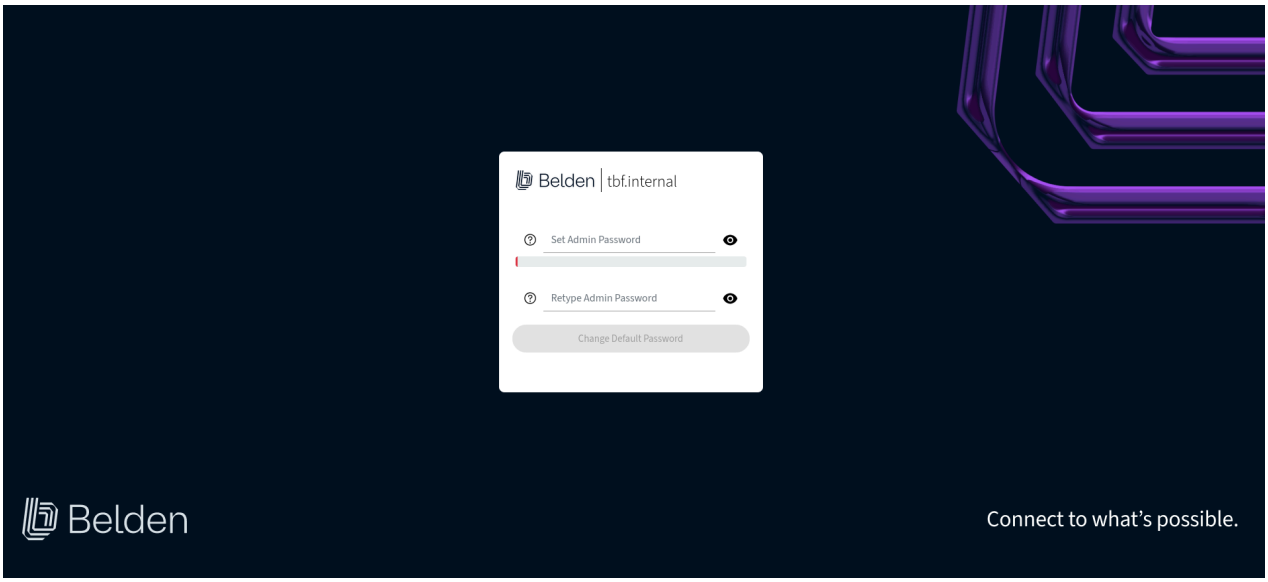


Figure 37: Initial password change

Perform the following steps:

1. Enter a new admin password.
2. Confirm the new admin password by entering it again.
3. Select **Change Default Password**.

Note: If you lose your password, reset the device to factory settings using the CLI or the **Reset** button.

Access TBF-800V web UI

This task describes how to access the TBF-800V web interface after system installation and initial setup.

Prerequisites:

Before you begin, verify that the TBF-800V system is installed and running. To help ensure the system is fully available, wait at least 2 to 3 minutes after boot up.

Perform the following steps:

1. To allow incoming traffic on the machine for web UI access via WAN, set up the network.

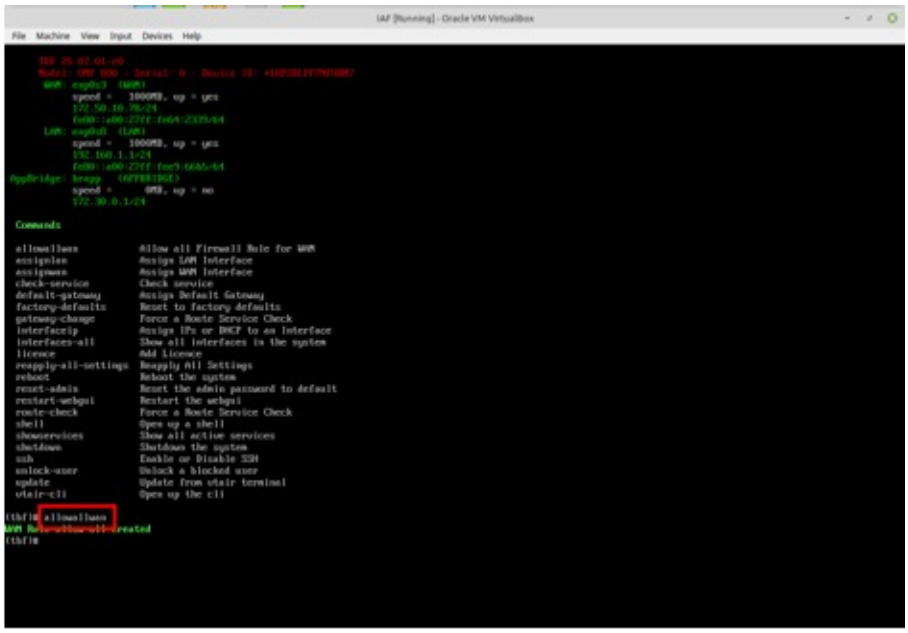


Figure 38: Allow WAN

2. Open a web browser. Navigate to the TBF-800V web interface.
   Enter the URL `https://<WANIP>` in the address bar of a browser. Replace `<WANIP>` with the actual WAN IP address of the system.

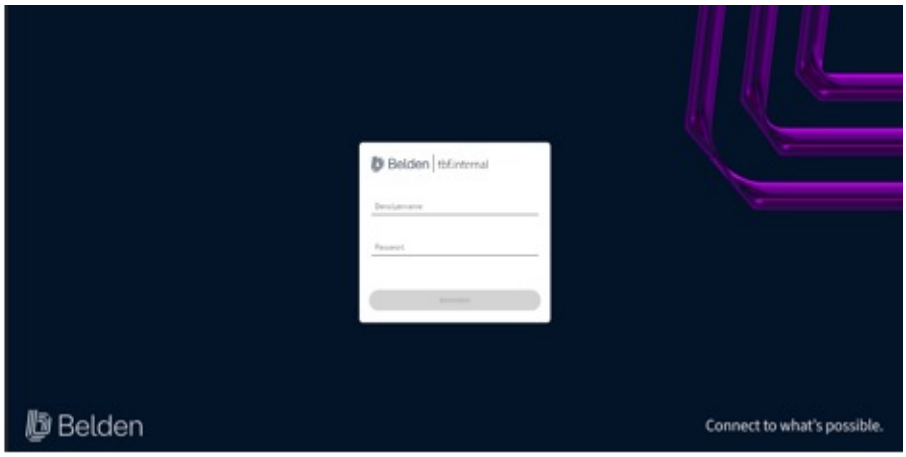


Figure 39: TBF login

3. When prompted, enter the default login credentials:
    a. Username: `admin`
    b. Password: `private`

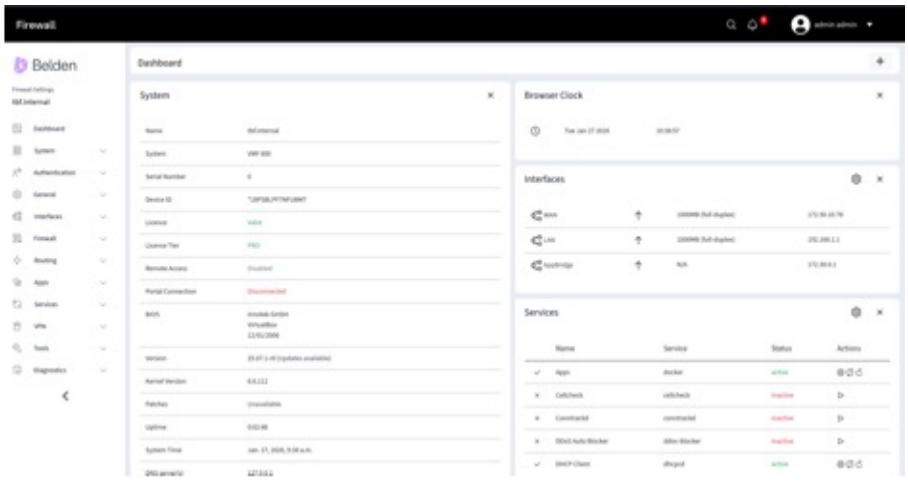


Figure 40: TBF dashboard

You now have access to the TBF-800V web interface, where you can perform system configuration and management tasks.

Licence activation

Activate your TBF licence to unlock all features and ensure compliance with your entitlement.

TBF requires a valid licence to operate. You can validate your licence online or offline depending on your network connectivity.

The licence activation process involves:

- Validating your licence through online or offline methods
- Resolving any licence validation issues
- Accessing the Belden Portal when required
- Entering your entitlement ID (EID) correctly

Choose the appropriate validation method based on your network setup and requirements.

Licence validation overview

Overview of TBF licence validation methods and requirements.

TBF supports two licence validation methods:

Online validation

Validates your licence directly through the internet. This method requires TBF to have internet access and is the fastest validation option.

Offline validation

Validates your licence without internet access. This method requires manual steps including downloading a licence file from the Sentinel EMS portal.

Both methods require your entitlement ID (EID) and access to the Belden Portal for authentication.

Important: Ensure you have your EID available before starting the validation process. The EID is provided with your TBF purchase documentation.

Validate licence online

Validate a TBF licence online when the device has internet access.

Prerequisites:

TBF must have access to the internet for online activation. If it does not have access, see *Validate licence offline* on page 26 instead.

Take the following steps to validate a TBF licence online.

Perform the following steps:

1. Complete the Belden Portal login procedure.

   See *Belden Portal login* on page 27 for detailed steps.

2. Go to **System > Licence**.

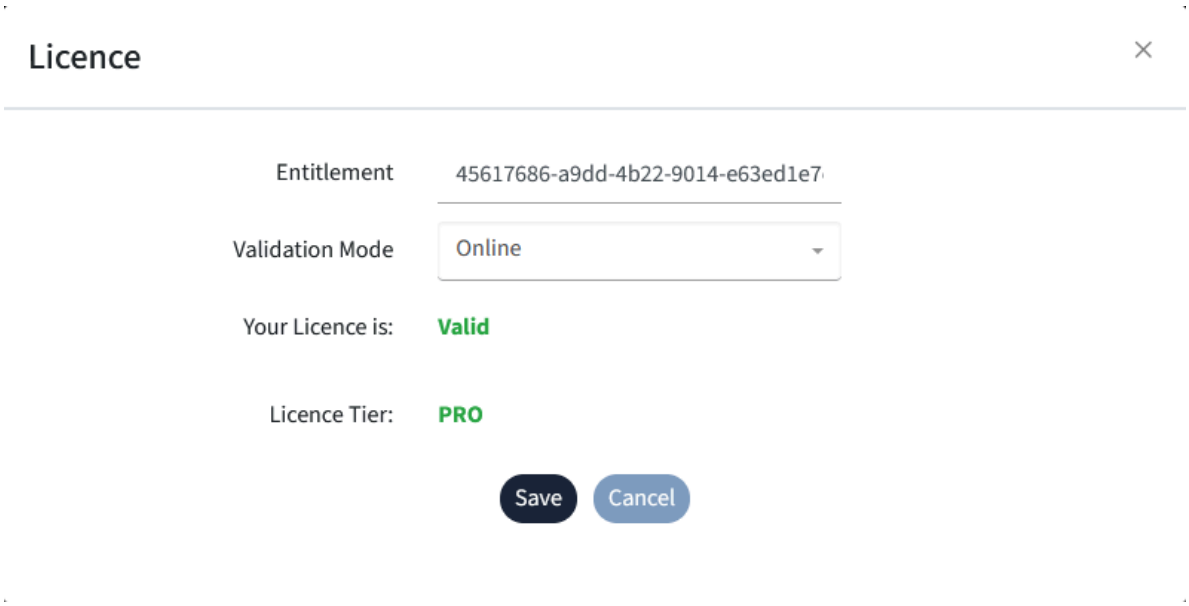


Figure 41: Online activation

3. For **Entitlement**, enter the EID.

   If the EID is not automatically displayed, you must enter it manually.

4. For **Validation Mode**, select **Online**.

5. Select **Save** to continue.

The online activation process starts. If it was successful, you see *Valid* and the licence tier.

Validate licence offline

Validate a TBF licence offline when the device does not have internet access.

Prerequisites:

You need access to a computer with internet connectivity to complete the offline validation process.

Use offline validation when TBF cannot access the internet directly. This process requires downloading a licence file from the Sentinel EMS portal.

Perform the following steps:

1. Retrieve the device ID from TBF.
    a. Go to **Dashboard**.
    b. In the **System Information** widget, locate the **Device ID**.
    c. Copy the Device ID for use in the Sentinel EMS portal.

2. Go to **System > Licence**.

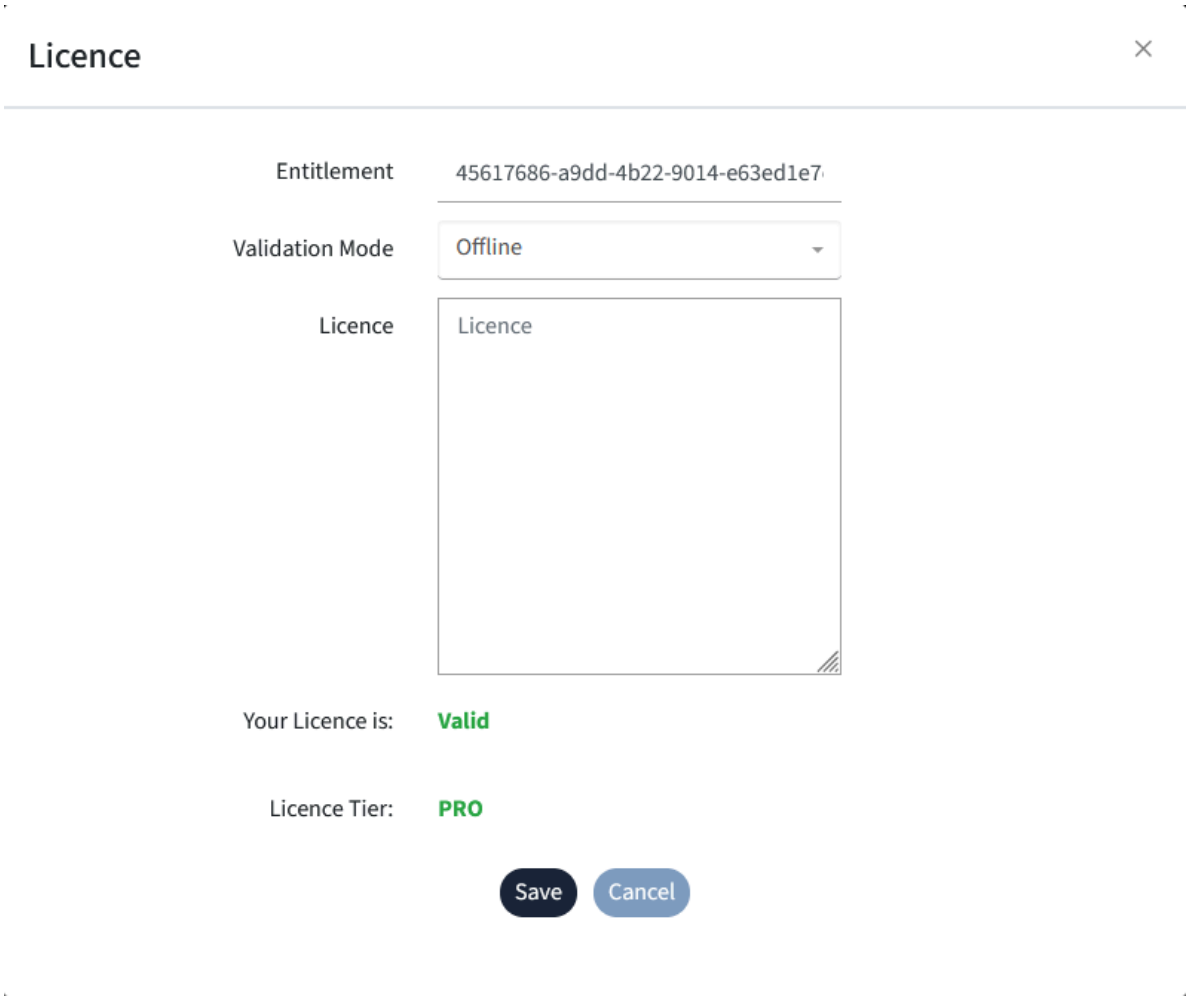


Figure 42: Offline activation

3. For **Entitlement**, enter the EID.

   If the EID is not automatically displayed, you must enter it manually.

4. For **Validation Mode**, select **Offline**.

5. Complete the Sentinel EMS activation process.
    a. Go to the *Sentinel EMS portal*.
    b. Log in with your credentials.
    c. Navigate to the licence activation section.
    d. Enter your EID and the Device ID you copied earlier.
    e. Download the generated licence file.

6. Install the licence file in TBF.
    a. Return to the TBF **System > Licence** page.
    b. Select **Choose File** and upload the licence file you downloaded.
    c. Select **Save** to install the licence.

The offline activation process completes. If successful, you see *Valid* and the licence tier.

Resolve invalid licence

Troubleshoot and resolve issues when TBF licence validation fails.

Your TBF licence shows as invalid or the validation process fails.

Licence validation can fail for several reasons:

- Incorrect EID entry
- Network connectivity issues during online validation
- Expired or inactive licence
- Incorrect device ID for offline validation
- Corrupted licence file

Perform the following steps:

1. Verify the EID is entered correctly.

   Check your purchase documentation for the correct EID format and characters.

2. For online validation, check network connectivity.

   Ensure TBF can access the internet and that firewall rules allow outbound connections to the licence server.

3. For offline validation, verify the device ID.

   Ensure you copied the correct Device ID from the Dashboard and entered it accurately in the Sentinel EMS portal.

4. Check licence status in the Belden Portal.

   Log in to verify your licence is active and not expired.

5. If using offline validation, download a fresh licence file.

   The licence file may have become corrupted during download or transfer.

6. Contact Belden support if the issue persists.

   Provide your EID, device ID, and any error messages for faster resolution.

Belden Portal login

Access the Belden Portal to manage licence entitlements and activations.

Portal access

The Belden Portal provides access to licence management and support resources. You need valid credentials to access the portal.

Login procedure

1. Go to the Belden Portal website.
2. Enter your username and password.
3. Select **Login** to access your account.
4. Navigate to the licence management section for TBF-related tasks.

Account requirements

You must have a valid Belden Portal account associated with your TBF purchase. If you do not have an account or cannot access it, contact Belden support for assistance.

EID entry instructions

Instructions for entering the Equipment ID (EID) during licence validation.

EID entry

For **Entitlement**, enter the EID.

If the EID is not automatically displayed, you must enter it manually.

EID format

The EID is a unique identifier provided with your TBF purchase documentation. Ensure you enter it exactly as provided, including any hyphens or special characters.

4. CLI

The CLI tool lets you set up a device through the console instead of the GUI.

The TBF CLI tool has a command-line interface for device configuration and management.

The CLI organizes commands into logical groups through a menu-driven interface. The CLI structure mirrors the functionality available in the GUI. You can use the CLI as an alternative to the GUI when you need console-based access or want to automate configuration tasks.

Key CLI features include:

- Hierarchical menu navigation
- Context-sensitive help
- Command completion
- Configuration validation
- Batch command execution

The CLI maintains the same security model as the GUI. Configuration changes require appropriate user permissions.

Access the CLI

Connect to the CLI through a console connection or SSH session.

Access the CLI through multiple connection methods depending on the network setup and security requirements.

Perform the following steps:

1. Choose a connection method:
    - Console connection: Connect directly to the device console port
    - SSH connection: Connect over the network using SSH
2. For console connection, connect the terminal to the device console port using appropriate cable and terminal settings.
3. For SSH connection, use an SSH client to connect to the device IP address.
4. Enter a username and password when prompted.
5. The CLI main menu appears after successful authentication.

You are now connected to the CLI. Navigate through the menu to perform configuration tasks.

Navigate the CLI

To move through the CLI, use menu navigation and help commands.

The CLI uses a hierarchical menu where you navigate through different levels to access specific configuration options.

Perform the following steps:

1. To select menu items, use the arrow keys or number keys.
2. To select a menu item or to execute a command, press **ENTER**.
3. To display available commands and options, type `help` or `?`.
4. To return to the previous menu level, use `back` or `exit`.
5. To exit the CLI completely, type `quit`.

You efficiently navigate through the CLI menu to access configuration options.

View the interface configuration

Display network interface settings and status information.

Use the CLI to view detailed information about network interface configurations, including IP addresses, status, and operational parameters.

Perform the following steps:

1. From the main menu, navigate to the `Network` or `Interface` configuration section.
2. Select the interface you want to view.
3. Select the option to display or to view interface configuration.
4. Review the displayed information, which includes:
    - Interface name and type
    - IP address and subnet mask
    - Interface status (up/down)
    - MAC address
    - Speed and duplex settings

The CLI displays the interface configuration details. Verify the current settings and troubleshoot connectivity issues.

Enable the DNS server

Enable DNS server functionality through the CLI.

The TBF device can function as a DNS server to resolve domain names for network clients. Use the CLI to enable and to set up DNS server functionality.

Perform the following steps:

1. From the main menu, navigate to the network services or DNS configuration section.
2. Select the **DNS Server** configuration option.
3. Select the option to enable the DNS server.
4. Set DNS server settings as needed:
    - Set upstream DNS servers
    - Set DNS zones if required
    - Set cache parameters
5. Save the configuration changes.
6. If the device prompts you, then restart the DNS service.

The DNS server resolves domain names for network clients.

5. Console access

Access the console menu through various connection methods to perform system administration tasks.

Access the console menu in two ways:

- Through the serial connection
- Through the VGA screen of the appliance

When you log in as the `root` user via SSH, the console displays as the main entry menu.

Use these default login credentials for both SSH and console access:

- Username: `root`
- Password: private

The console gives you access to system functions, including:

- System reboot and shutdown operations
- Network interface configuration viewing
- Shell access for advanced system administration
- Basic system diagnostics and troubleshooting

SSH console access

Connect to the console via SSH to perform system administration tasks.

Prerequisites:

Enable SSH first, as described in the *settings* on page 352.

SSH gives you encrypted remote access to the console interface. When you log in with the `root` user, the console menu appears automatically.

Perform the following steps:

1. Connect to the appliance via SSH using the `root` credentials.
2. Log in with the `root` user account.
   The console menu appears automatically after successful login.

You now have access to the SSH console interface and can perform system administration tasks.

```
     TBF 25.10.1-r2
     Model: TBF 800 - Serial: 0b80aa0e-1b6e-4b2c-9420-d59622e32295 - Device ID:
*1NQHF39EEV85KAT
      WAN: enp0s18 (WAN)
           speed =      0MB, up = yes
           192.168.10.131/24
           fe80::be24:11ff:febc:c8f9/64
      LAN: enp0s19 (LAN)
           speed =      0MB, up = yes
           192.168.1.2/24
           fe80::be24:11ff:fe98:7257/64
AppBridge: brapp   (APPBRIDGE)
           speed =      0MB, up = no
           172.30.0.1/24

 Commands

 allowallwan           Allow all Firewall Rule for WAN
 assignlan             Assign LAN Interface
 assignwan             Assign WAN Interface
 check-service         Check service
 default-gateway       Assign Default Gateway
 factory-defaults      Reset to factory defaults
 gateway-change        Force a Route Service Check
 interfaceip           Assign IPs or DHCP to an Interface
 interfaces-all        Show all interfaces in the system
 licence               Add Licence
 reapply-all-settings  Reapply All Settings
 reboot                Reboot the system
 reset-admin           Reset the admin password to default
 restart-webgui        Restart the webgui
 route-check           Force a Route Service Check
 shell                 Open up a shell
 showservices          Show all active services
 shutdown              Shutdown the system
 ssh                   Enable or Disable SSH
 unlock-user           Unlock a blocked user
 update                Update from vtair terminal
 vtair-cli             Open up the cli
```

Figure 43: Console (SSH)

Console usage

The console lets you perform basic system tasks such as rebooting or shutting down the system. You can also view network interface configurations within the system.

To access the same shell available from the GUI, use the `shell` command. To exit the shell, type `exit`.

Press **ENTER** or **CTRL+d** to clear the console window.

```
(vtair)# showservices

Name                | Service              | Enabled  | Status

Apps                | docker               | True     | active
Cellcheck           | cellcheck            | False    | inactive
Conntrackd          | conntrackd           | False    | inactive
DDoS Auto Blocker   | ddos-blocker         | False    | inactive
DHCP Client         | dhcpcd               | True     | active
DHCPRA              | radvd                | False    | inactive
DNS                 | unbound              | False    | inactive
DynamicRouting      | frr                  | True     | active
Gateway Pinger RRD Collector | dpinger-rrd  | True     | active
HAProxy             | haproxy              | False    | inactive
IPSec               | strongswan           | True     | active
Intrusion Detection | suricata             | False    | inactive
KEA DHCP Control Agent | kea-ctrl-agent    | False    | inactive
KEA DHCP v4         | kea-dhcp4-server     | False    | inactive
KEA DHCP v6         | kea-dhcp6-server     | False    | inactive
LLDP                | lldpd                | False    | inactive
Loganalyzer         | loganalyzer          | False    | inactive
MSTP                | mstpd                | True     | active
NTP                 | ntp                  | True     | active
Nginx               | nginx                | True     | active
PortalConnector     | vtair-portal-connector | False  | inactive
Postfix             | postfix              | False    | inactive
Redis               | redis                | True     | active
Routing             | vtair-routing        | True     | active
SNMP                | snmpd                | False    | inactive
SSH                 | ssh                  | True     | active
Syslog              | rsyslog              | True     | active
Tofino AMP          | tofino-amp           | False    | inactive
Tofino DNP3         | tofino-dnp3          | False    | inactive
Tofino ENIP         | tofino-enip          | False    | inactive
Tofino GOOSE        | tofino-goose         | False    | inactive
Tofino IEC104       | tofino-iec104        | False    | inactive
Tofino Modbus       | tofino-modbus        | False    | inactive
Tofino OPC          | tofino-opc           | False    | inactive
UPnPNAT             | miniupnpd            | False    | inactive
VirtualIP           | keepalived           | True     | active
Watchdog            | watchdog             | False    | inactive
Web Filter          | squid                | True     | active
Web Filter CICAP    | squid-logger         | False    | inactive
Web Filter Clamav   | clamav-daemon        | False    | inactive
Web Filter Freshclam | clamav-freshclam    | False    | inactive
Web Filter ICAP Logger | c-icap             | True     | active
vnStat              | vnstat               | True     | active

(vtair)#
```

Figure 44: Console (SSH)

Serial console access

Connect to the console through a serial connection using terminal programs.

Some TBF models support USB cable connections for establishing a serial connection. Use a terminal program to connect to the serial console interface and access system administration functions.

Perform the following steps:

1. Select a terminal program that matches the operating system:

Operating system	Recommended program
Windows	PuTTY
Mac OS X	Screen, serial
Linux	Screen

2. Identify the serial port that the connection uses.
3. Enter the serial port information in the terminal program.
4. Set the baud rate to `115200`.
5. For screen-based programs, use this command:

   ```
   screen /dev/ttyNAME 115200
   ```

   Replace `NAME` with the USB connection name, for example `USB0`. Press **ENTER** twice.

The console menu appears and displays the same interface that you see when you connect through SSH. This menu gives you access to system administration functions.

6. Dashboard

The dashboard has a customizable interface to monitor the system status through widgets.

The dashboard is the main interface for monitoring the system status and managing information displays. This section covers dashboard functionality and available widgets.

Dashboard menu

The **Dashboard** menu appears as the first screen when you log in to TBF. You can return to the dashboard at any time by selecting the logo in the upper-left corner.

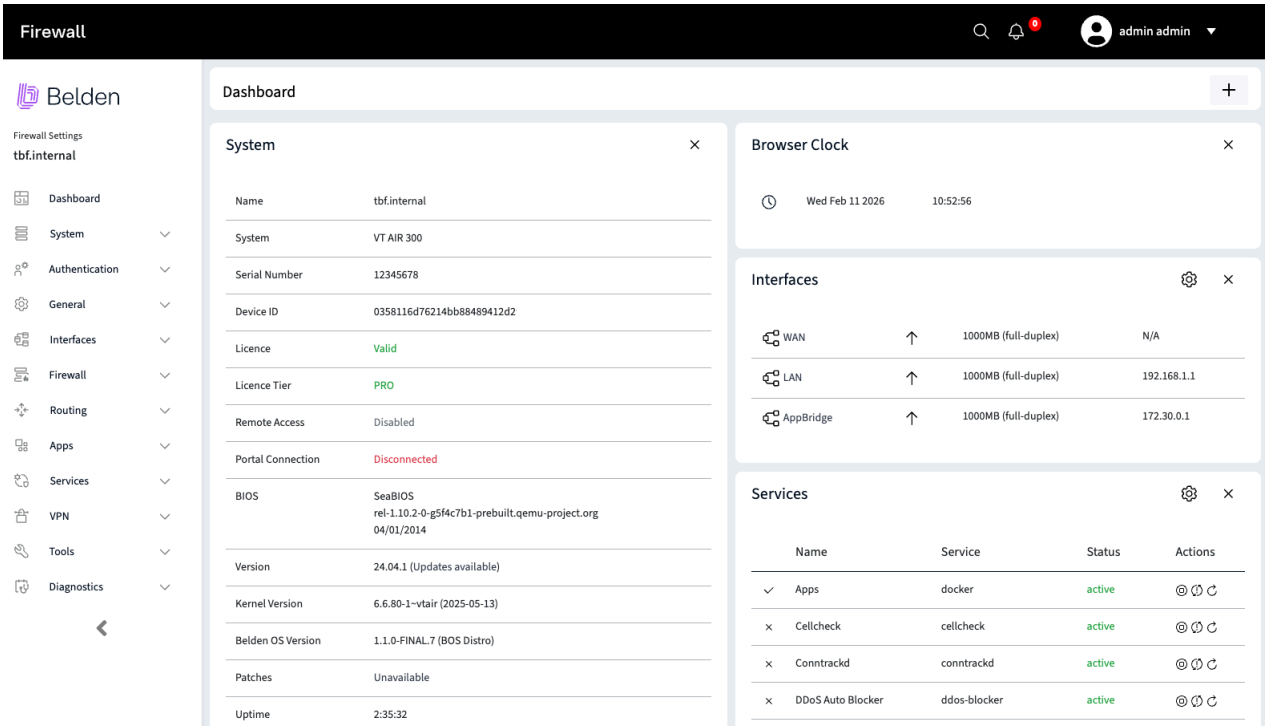


Figure 45: Dashboard

The dashboard shows 2 columns by default. You can add widgets to each column and move widgets between columns as needed.

You can also change the column layout in the *settings* on page 63.

Widget management

To remove a widget, select the **X** button in the upper-right corner of the widget.

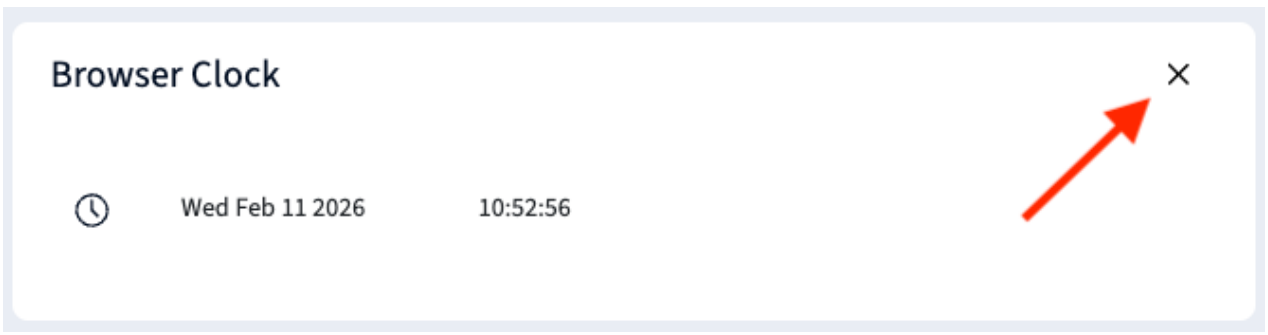


Figure 46: Remove widget

To add widgets to the dashboard, select the **+** button in the dashboard banner. This action opens a list of available widgets. Select the **+** button next to any widget to add it to the dashboard.

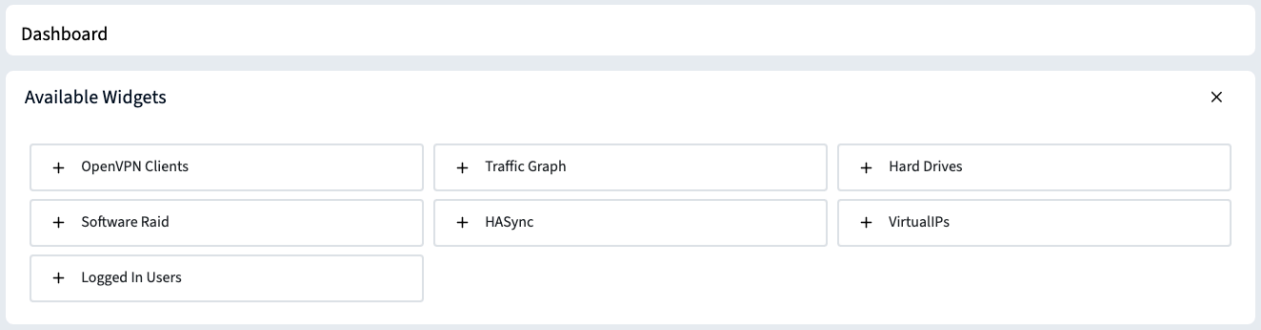


Figure 47: Available widgets

Some widgets allow you to customize their settings. When settings are available, a gear wheel symbol appears next to the **X** button in the upper-right corner of the widget.

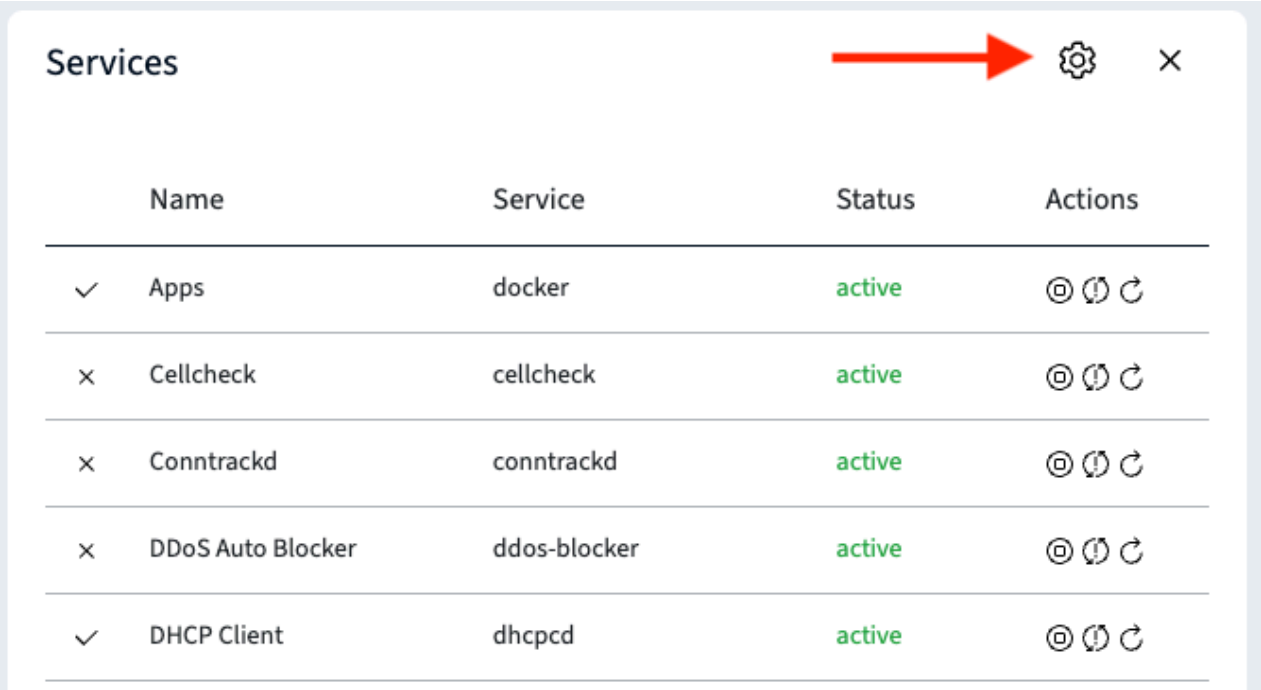


Figure 48: Edit widgets

Note: Widgets update their content automatically.

States widget

The **States** widget shows state information, including active states, maximum configured states, and offloaded states.

Functionality

The **States** widget shows the number of states compared to the maximum configured number of states. The widget also displays the number of offloaded states.

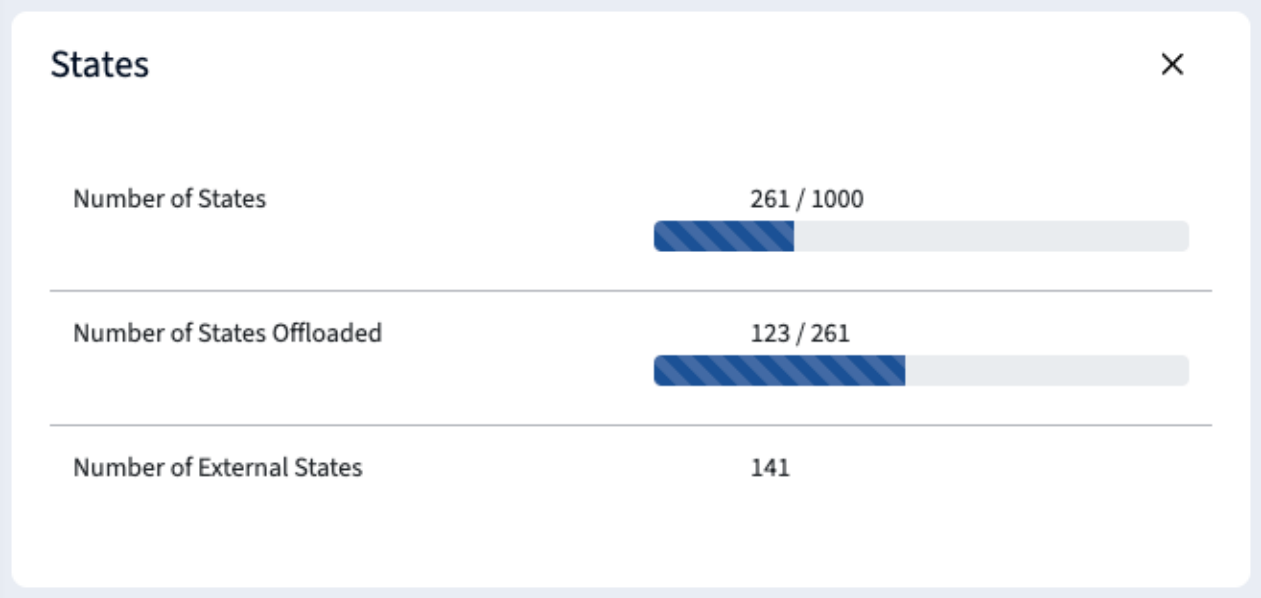


Figure 49: States widget

Metrics

The **States** widget shows these state metrics:

- Current active states
- Maximum configured states
- Number of offloaded states

Security widget

The **Security** widget shows the status and statistics for security services, including firewall and intrusion detection.

Functionality

The **Security** widget shows the status of security services.

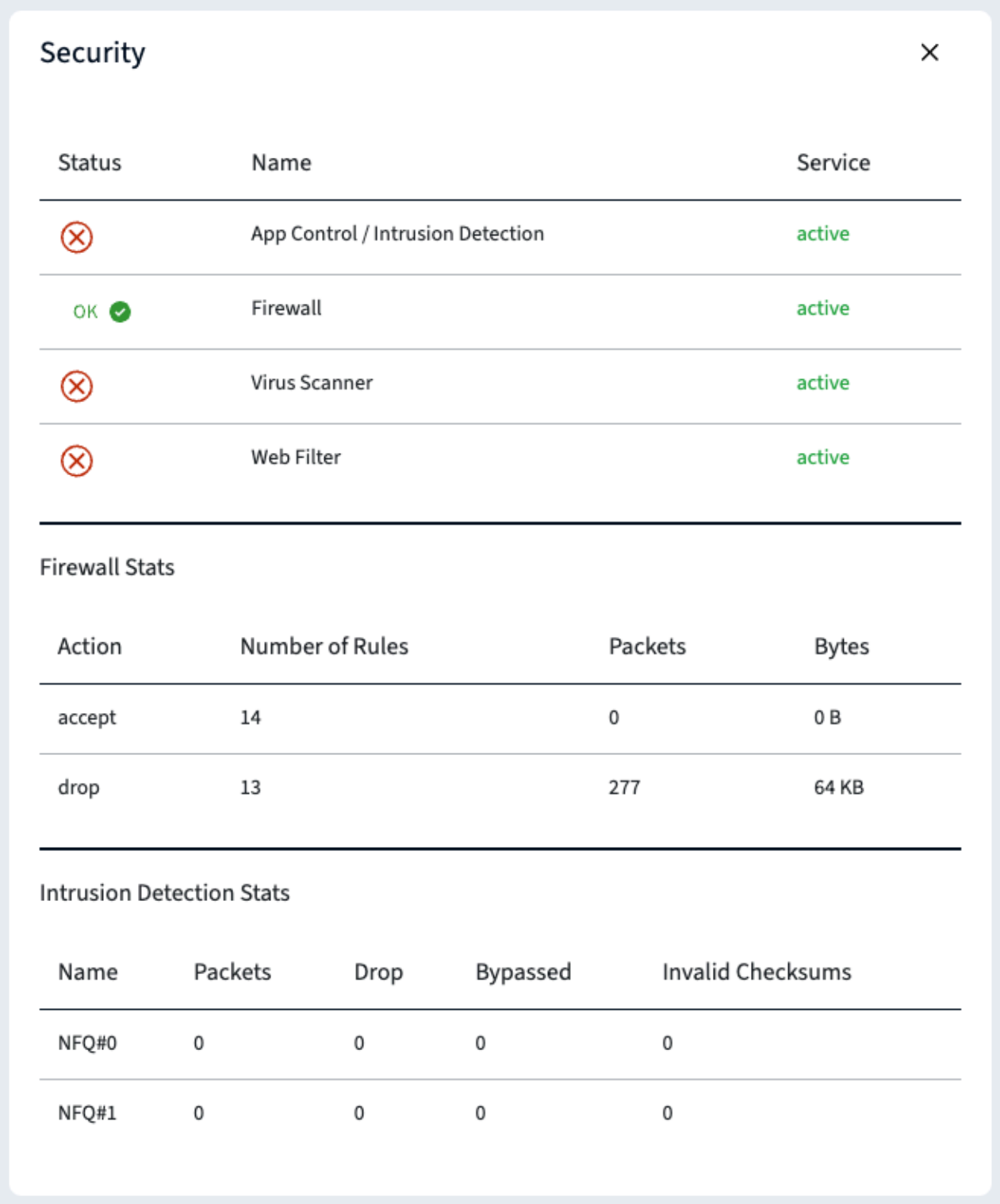


Figure 50: Security widget

Firewall statistics

Firewall Stats shows statistics for rules organized by categories such as `accept` and `drop`. **Firewall Stats** also shows the total number of rules, packets, and bytes processed.

Intrusion detection statistics

Intrusion Detection Stats shows statistics for each network interface, including the number of packets processed, drops, bypasses, and invalid checksums detected.

7. System settings

System settings help provide centralized control over device configuration, licencing, portal connectivity, and system extensions.

System settings include the essential configuration options for the TBF device. These settings control how the device operates, connects to external services, and manages features that affect the entire device.

The system settings include these categories:

- *Global settings* on page 63: Basic system configuration, including hostname, web server settings, language preferences, and system updates
- *Licence* on page 76: Licence management and feature activation
- *Portal settings* on page 76: Portal connectivity and remote access configuration
- *Addons* on page 77: System extensions and additional functionality modules
- *Copyright* on page 79: Legal information and third-party component details

Set these settings during initial system installation. The settings help ensure that the device operates well in the network. Many settings apply immediately, while some require a system restart to take effect.

Global settings

Global settings control fundamental system parameters, including hostname, web server configuration, language preferences, and system update options.

Global settings help provide centralized control over the core operational parameters of the TBF device. These settings affect the entire system, and set them up during initial setup to help provide expected performance.

The global settings include the following areas:

- *Settings* on page 63: Basic system configuration, including hostname, web server, and SSH settings
- *Updates* on page 67: System update management and scheduling
- *Notifications* on page 69: System notification preferences and delivery settings
- *Advanced settings* on page 70: Advanced system features and performance tuning options

Review these settings during initial setup to help ensure they meet operational requirements.

Note: The default settings function immediately but may require adjustment for the specific environment.

Settings

This section contains general configuration options for the system, including host settings, web server configuration, language preferences, and SSH settings.

Find the general settings under **System > Settings**. Set various options, including:

- Host name and domain
- Web server settings
- Global language and timezone
- SSH settings

This is a good starting point when you set up TBF. Review these options and verify they meet the requirements.

Note: The default settings are sufficient and work immediately without additional setup.

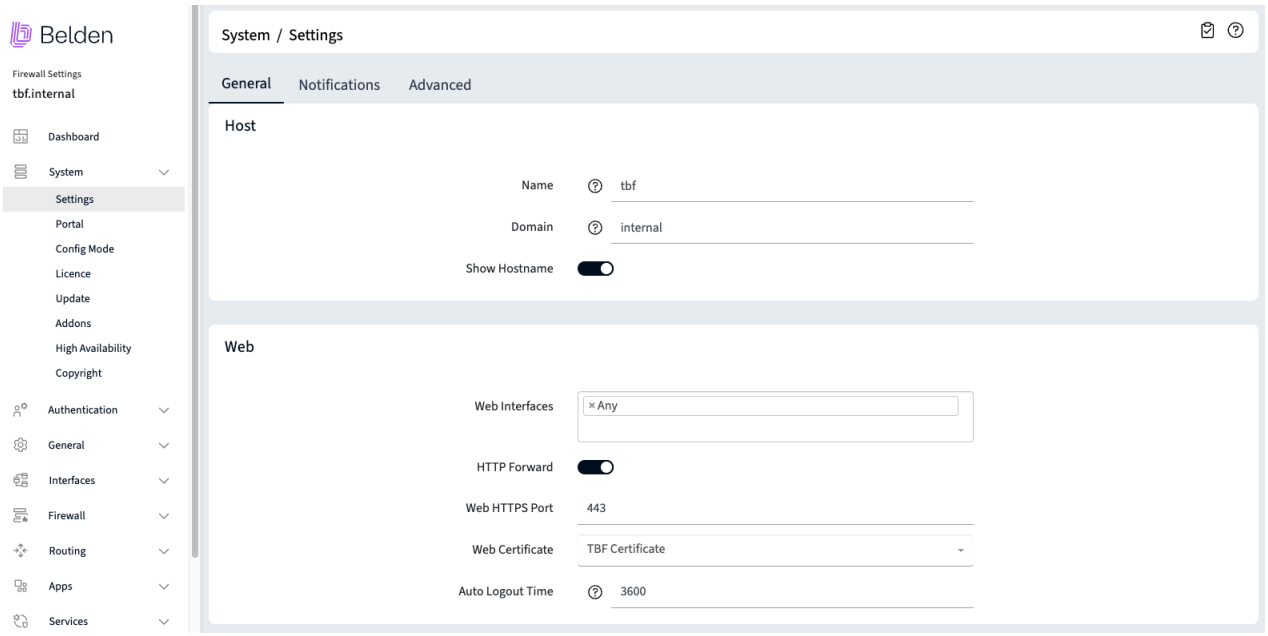


Figure 51: General

To enable **Config Mode**, navigate to **System > Config Mode**. When enabled, TBF does not apply any changes you make immediately. This lets you set up everything first and then apply any changes at once.

Note: The **Config Mode** feature is particularly useful to make changes during normal operation, as it helps reduce TBF downtime.

Host settings

Set hostname and domain settings for the device.

Name

This field specifies the hostname of TBF without the domain part.

Domain

This field specifies the domain part of the host.

Show Hostname

This setting displays the entire hostname in the top bar and login screen of the GUI. This display helps you identify a specific device.

Web settings

Set web interface settings, including HTTPS, ports, and certificates.

Web Interfaces

This setting sets up one or more interfaces and IP addresses for the web interfaces.

HTTP Forward

This setting automatically redirects HTTP requests to encrypted HTTPS connections.

Web HTTPS Port

This field specifies the HTTPS port that the web interfaces use. The default is `443`.

Web Certificate

This setting specifies the web certificate. TBF uses this certificate to help secure the HTTPS connection to the web interfaces.

Auto Logout Time

This field determines how long in seconds a user remains logged in while inactive. The default is `3600` seconds (1 hour).

Localization settings

Set timezone and language settings for the device interface.

Timezone

This setting specifies the timezone for the TBF device.

Default Language

This setting specifies the default language for the TBF device interface.

SSH settings

Set SSH access settings, including authentication and rate limiting.

Enable SSH

This setting enables or disables SSH access. This setting restricts authentication to SSH keys only. When enabled, add SSH keys to each user account for them to log in.

SSH Port

This field specifies the port number for SSH connections.

Key Authentication Only

This setting restricts authentication to SSH keys only. Password-based login is disabled.

SSH Rate Limit

This field determines the maximum number of connections per IP address that can access TBF.

SSH Rate Limit Time

This field specifies the time, in seconds or minutes, over which the rate limit applies.

For example, a rate limit of 5 per minute restricts a single IP address to a maximum of 5 connections within any 1-minute period.

Firewall settings

Apply firewall anti-lockout rules to help prevent accidental loss of access.

Anti Lockout Rules

This setting automatically creates firewall rules that help prevent you from accidentally losing access to the web interface because of misconfigured firewall settings.

DNS settings

Set DNS server settings, including localhost and forward servers.

DNS Server Override

This setting allows the DHCP server to override the DNS server settings.

DNS Localhost

This setting uses the internal DNS server for the network. Enable this feature under **Services**.

DNS Forward Servers

This setting displays the IP addresses of external DNS servers that TBF uses for DNS resolution.

NAT & firewall settings

Set up NAT reflection and automatic VRRP VIP firewall rules.

NAT Reflection

This feature allows internal clients to access DNAT-forwarded IP addresses using their external address. Without NAT reflection, a DNAT-forwarded WAN IP address cannot be accessed from within the local network. *Split DNS* can resolve this limitation. Set up *Split DNS* to direct the internal DNS entry to the local IP address of the server.

Auto VRRP VIP rules automatically generate firewall rules for the following services:

- VRRP VIP
- DHCP
- OSPF

TBF updates these rules automatically when you modify settings in any of these services. To manage these firewall rules manually, disable the automatic generation.

Miscellaneous settings

Set up various system options, which include dashboard, logging, and maintenance features.

Dashboard Columns

This setting determines the number of columns to display on the main dashboard. Select from:

- **2**: This option is the default.
- **3**
- **4**
- **5**

Prefer to use IPv4 even if IPv6 is available

When both IPv6 and IPv4 are available for a connection, TBF defaults to IPv6. To prioritize IPv4 connections instead, enable this setting.

Enable Watchdog

This setting enables the hardware watchdog, which automatically detects severe system errors that cause the software to become inoperable and automatically restarts the TBF device.

OpenSSL Engine

This setting is **Dynamic (Default)** or **AF ALG (Kernel Crypto API)**.

Console Password

This setting enables password-based access control for the console. To unlock console access, TBF requires the administrator password.

Disable API

This setting disables the REST API functionality.

Log to RAM

This setting temporarily stores log files in RAM and copies them to the SSD every hour. This reduces write operations on the SSD, which potentially extends its lifespan.

Maximum log to RAM size (MB)

This field determines the maximum size of the RAM disk that is used for logging. TBF reserves this amount from available RAM. Set this value to a size that leaves sufficient RAM for system operation.

Logfile Analysis

This feature stores selected log files such as firewall and intrusion protection logs for long-term analysis in the GUI.

Note: This feature impacts performance and uses disk storage. When you require higher performance, disable this feature.

Maximum Logfile Analysis Days

This field determines how many days TBF stores each log file type for analysis. More days require additional disk storage. For smaller systems, keep this value low.

An automatic log file cleaner runs continuously in the background. It clears large log files when disk space or RAM disk space becomes insufficient. This mechanism helps prevent a full log folder, which prevents TBF from storing new log files.

At the bottom of the **Miscellaneous** section, there are several maintenance options:

- Clear the cache
- Reapply settings
- Scan for new interfaces
- Reapply dynamic IP addresses
- Reset the dashboard

Basic configuration example

This topic covers a step-by-step guide for initial device setup and configuration.

When you initially set up a TBF device, assign it a recognizable host **Name** that identifies the device within the network. For example, when you have multiple TBF devices in the network, include their physical location as part of the hostname. This makes it easier to identify the correct device when you require physical access.

Perform the following steps:

1. To complete the fully qualified domain name (FQDN) of the device, change the **Domain** to the domain name of the company.
2. To encrypt the connection between the system and the TBF device when you access the GUI, enable **HTTP Forward**.
   Changes only take effect after you select **Save** at the bottom of the page and then refresh the browser.
3. Enter a **Timezone** and **Default Language** preference for the device.
4. When you require remote access to *console access* on page 55, enable SSH on the device.
5. To enable changes from the portal, enable portal connections.
6. Keep **Anti Lockout Rules** enabled.
   This creates automatic firewall rules that help prevent accidental lockouts because of incorrect firewall setups.
7. Keep **DNS Localhost** enabled unless you have a specific reason to disable it.
8. When the ISP offers a Dual-Stack (Lite) connection instead of a native IPv4 connection, enable **Prefer to use IPv4 even if IPv6 is available**.
   This routes traffic over IPv4 whenever possible instead of an automatic switch to IPv6 when available.
9. To save the setup, select **Save** at the bottom of the page.

Updates

Manage system updates, set up automatic update settings, and troubleshoot update issues.

Find the update settings under **System > Update**.

Overview

The update system provides manual online updates, manual offline updates using signed update images, automatic update scheduling, email notifications, and troubleshooting guidance.

Update tab

Check for online updates or install a signed offline update image.

Prerequisites:

For an offline update, download the signed `.swu` update image from the Belden Portal at *https://my.belden.com/*.

On the **Update** tab, check for online updates or upload and install a signed offline update image.

Note: Version checks and image installation use `tbf-swupdate` (SWUpdate).

Follow either the online update flow or the offline update flow.

Perform the following steps:

1. Navigate to the **Update** tab.
2. Perform an online update.
    a. Review the **Current Version**, **Latest Version**, and **Status** values.
    b. Select **Check for Updates**.
    c. When an update is available, select **Update**.
3. Perform an offline update.
    a. Under **Offline Update**, select **Choose File**.
    b. Select the signed `.swu` update image.
    c. Select **Upload and Install**.

Update settings

Set up automatic updates, email notifications, and proxy settings for the update system.

Settings

Auto Update

This setting enables automatic updates. Enabling this option displays additional options.

Weekday

The weekday on which the automatic updater runs.

Time

The time at which the automatic updater runs. It is a random time between 2 AM and 5 AM, and TBF automatically sets it.

Update Email Notification

This setting lets you receive update notifications by email. The default email address is `root@localhost.com`. You can change it after you enable it. Enabling this option displays additional options.

Send Update Information Report

This setting sends emails for update information reports.

Update Email Schedule

This setting sets the update email schedule. The default is **Weekly**.

Update Email Send all Updates

This setting sends all available updates. If you disable it, it only sends new updates compared to the last email.

Send Update Installation Report

This setting sends emails for update installation reports.

Enable Proxy

This setting uses an HTTP/HTTPS proxy for update downloads.

Proxy URL

This field sets the proxy URL. For example, `http://user:password@server:port`.

Miscellaneous settings

This topic describes additional system settings, including network preferences, security options, and interface settings.

Settings

Prefer to use IPv4 even if IPv6 is available

By default, when you set up IPv6 and a hostname resolves both IPv6 and IPv4 addresses, the system uses IPv6. If you enable this option, the system prefers IPv4 over IPv6.

Strict Reverse Path Filter

The current recommended practice in RFC3704 is to enable strict mode to help prevent IP spoofing from DDoS attacks. This is the default. If you use asymmetric routing or other complex routing configurations, then disable this setting.

Console Password

This field requires a password to access the console. You need the administrator password to unlock it.

Disable API

This setting disables the REST API functionality.

Dashboard Columns

This setting determines the number of columns to display on the dashboard. The default is **2**.

Notifications

This is an email notification system to receive alerts when TBF detects an issue.

Find the notification settings under **System > Settings > Notifications**.

To receive alerts when TBF detects an issue, set up email notifications.

Note: TBF disables notifications by default.

Define the following options as needed:

- Disable notifications
- Email server
- Email port
- From and to email
- Authentication

To verify the email setup, select **Send Test E-Mail**.

System messages

System message types and their notification triggers when email notifications are enabled.

Enable notifications to also enable system messages. System messages use the same email server setup as the general notification settings.

Message types

Interface up/down

This setting sends a notification when a network interface becomes available or unavailable.

Gateway up/down

This setting sends a notification when the default gateway becomes available or unavailable.

VirtualIP change

This setting sends a notification when the Virtual Router Redundancy Protocol (VRRP) state changes.

Disk mail root notifications

Built-in monitoring service that generates notifications for disk mail root directory issues.

A built-in service monitors the disk mail root directory and detects issues. Every 3 hours, TBF generates and displays notifications in the GUI.

Note: Messages appear only when you enable the mail notification setting. When enabled, TBF sends notifications to the given email address.

Advanced settings

Set advanced system settings, including performance optimization, security features, and network protocols.

Find the advanced settings under **System > Settings > Advanced**.

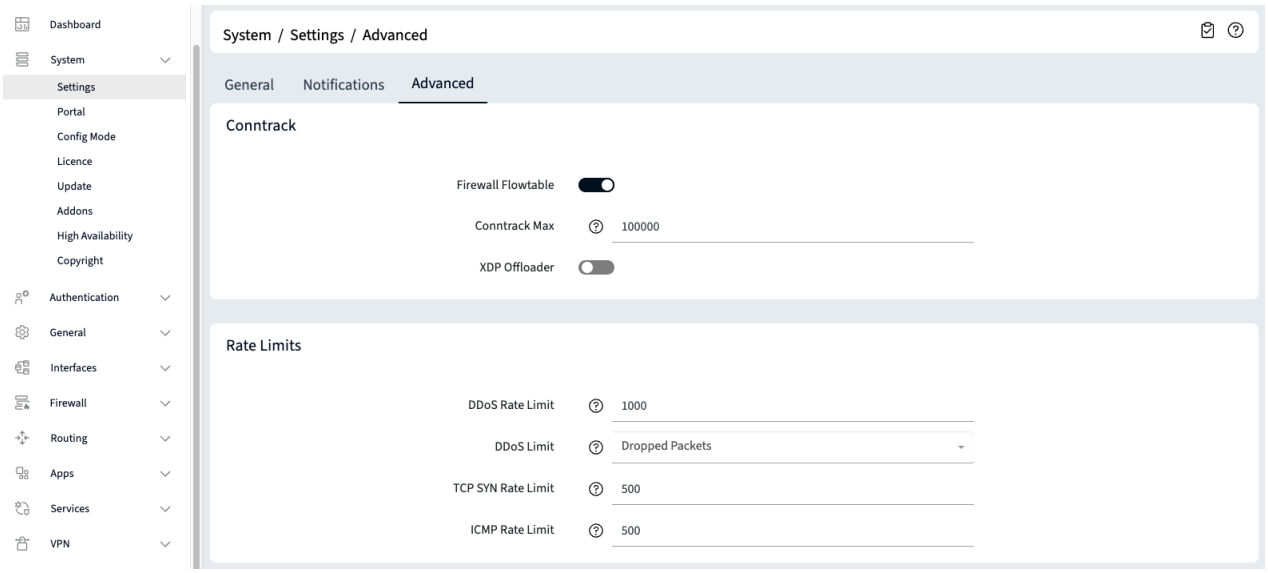


Figure 52: Advanced settings

Enable **XDP Offloader** here. Also enable **Firewall Flowtable** for XDP to work properly.

Rate limits settings

Set DDoS protection and rate-limiting settings to help protect against various types of network attacks.

DDoS Rate Limit

This field defines the maximum number of connections per IP address per minute before the system blacklists the IP address for 60 seconds. Enable this setting on each interface for it to be effective. The default is `1000`.

DDoS Limit

This setting can be either:

- **All**: This option also counts valid traffic against the limit. Select a limit that accommodates the expected legitimate traffic volume.
- **Dropped Packets**: This option is the default.

TCP SYN Rate Limit

This limit targets TCP SYN packets to help protect against SYN flood attacks. The default is `500`. A value of `0` disables this limit.

ICMP Rate Limit

This limit targets ICMP packets to help protect against ICMP flood attacks. The default is `500`. A value of `0` disables this limit.

Timeouts settings

Set firewall timeout settings for various network protocols and connection tracking.

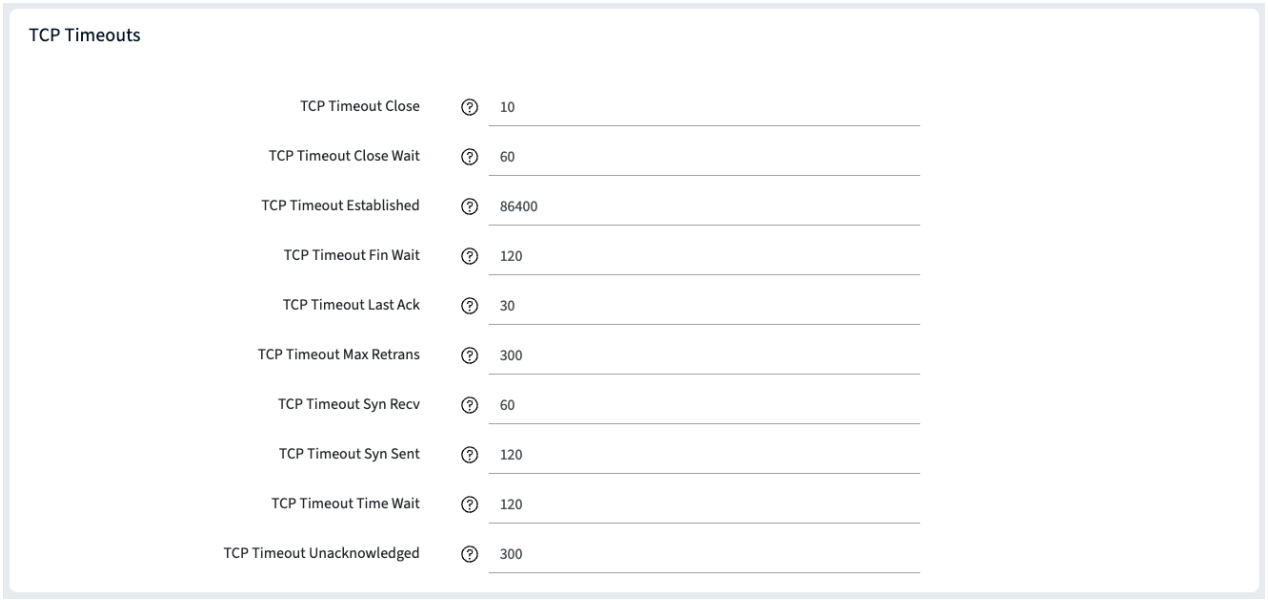


Figure 53: TCP timeouts



Figure 54: UDP and GRE timeouts

Set up firewall timeouts for the following protocols:

- ICMP
- ICMPv6
- TCP
- UDP
- GRE

You can also set up the connection tracking states table size.

Firewall helper settings

Firewall helpers assist with protocols that need to open additional random ports for operation.

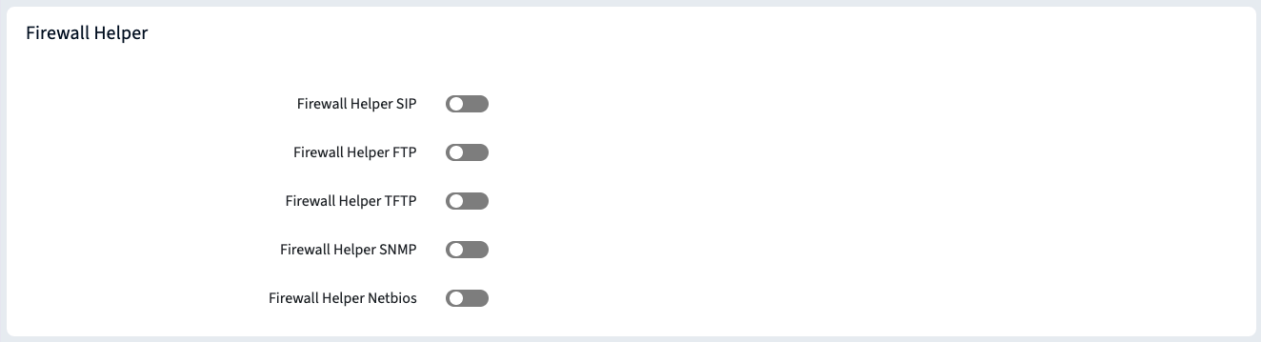


Figure 55: Firewall helper

There are four firewall helpers that assist with protocols that need to open additional random ports:

- SIP
- FTP
- TFTP
- SNMP

You can enable each helper individually. When a connection that uses one of these protocols opens an additional port, the firewall automatically tracks the port. This tracking removes the need to add a new firewall rule for that port.

Network interfaces settings

Set up hardware and software offload features for network interfaces to help improve performance.

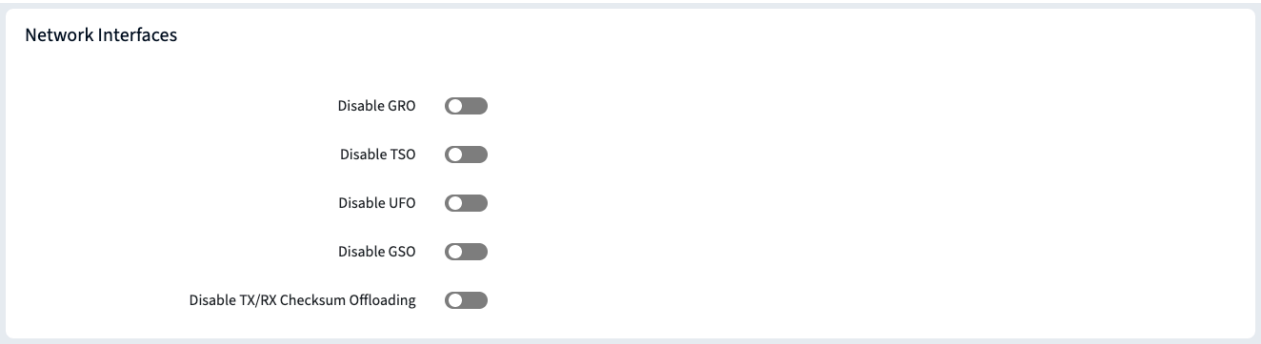


Figure 56: Network interfaces

You can enable or disable the following hardware and software offload features:

- GRO (Generic Receive Offload)
- TSO (TCP Segmentation Offload)
- UFO (UDP Fragmentation Offload)
- GSO (Generic Segmentation Offload)
- TX/RX Checksum Offload

Tip: You do not need to restart the system to enable or disable any of these settings.

Allowlisting settings

Set IP addresses and networks that bypass login blocking after unsuccessful authentication attempts.

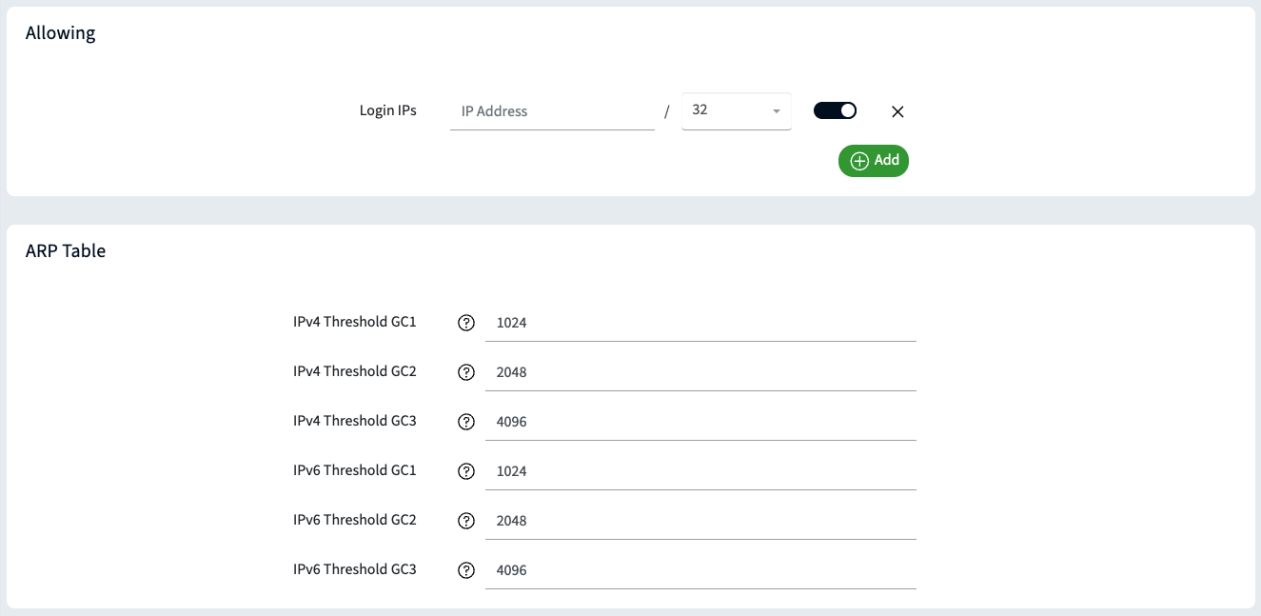


Figure 57: Allowlisting

You can add multiple IPv4 or IPv6 addresses or networks. The system does not block these addresses at the TBF login when the username or password is incorrect. Without this allowlist, the login system blocks access after three unsuccessful login attempts.

ARP table settings

Set ARP table cache thresholds for IPv4 and IPv6 address resolution.

Define the ARP table cache threshold for IPv4 and IPv6, as well as for GC1, GC2, and GC3. The default values are:

- GC1: `1024`
- GC2: `2048`
- GC3: `4096`

Tip: If you have many clients, then increase the threshold values.

Miscellaneous settings

This topic describes various system settings, which include gateway behavior, audit logs, CPU performance, and VRRP options.

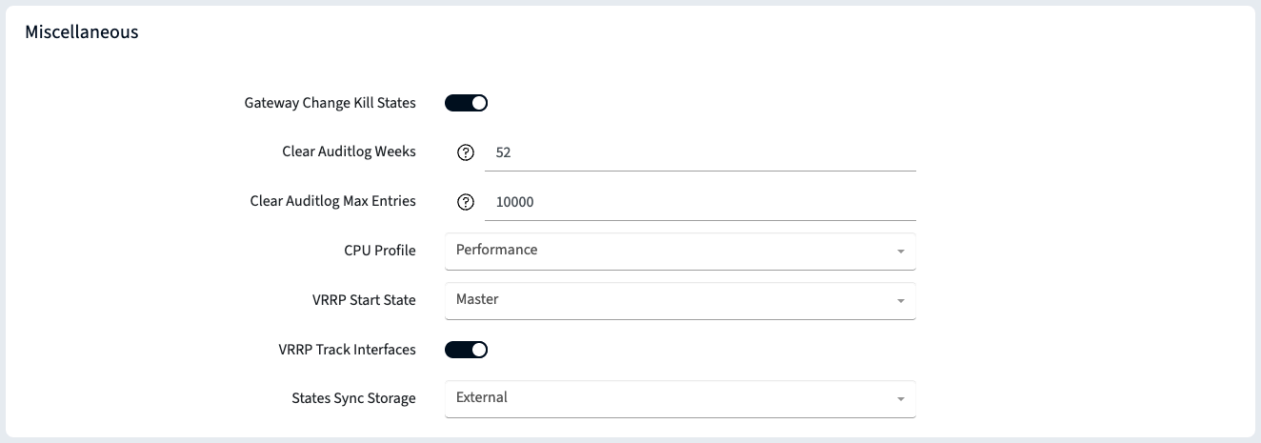


Figure 58: Miscellaneous

Gateway Change Kill States

This setting terminates connection states when the gateway changes. To force a gateway change, use this setting.

WARNING: This setting disrupts active connections.

Clear Auditlog Weeks

This field defines how long TBF retains audit logs with system actions. The default is `52`.

Clear Auditlog Max Entries

This field clears audit logs with system actions when there are more entries than this value. The default is `10000`.

WARNING: A large number of audit logs slows down the GUI performance. Use the text audit log from the logging settings instead.

CPU Profile

This setting determines the CPU performance and power profile of the system. The following options are available:

- **Performance**: This option uses the high available CPU speed but also uses more energy and generates more heat.
- **Dynamic (Schedutil)**: This option reduces CPU speed or puts CPUs to sleep when the system does not need them. While this profile is slower or takes longer for CPUs to become ready, the system uses less energy and stays cooler.

VRRP Start State

This setting is either **Master** or **Backup**. Set the primary TBF to **Master** and the others to **Backup**.

Important: When you do not set this to **Master**, the VRRP IP addresses disappear for a few seconds during changes or service reloads.

VRRP Track Interfaces

This setting monitors interfaces and fails over IP addresses when an interface goes down.

States Sync Storage

This setting is one of the following:

- **Internal**: This option stores states directly in the active state table. While this requires greater processing resources, it has faster failover time because the states are immediately available.
- **External**: This option stores states in an external table and loads them during failover. This option is more efficient but creates a delay between the failover and when the states become available.

Firewall flowtable

The firewall flowtable offers a fast-forwarding path for TCP/UDP packets and helps improve packet processing performance.

The flowtable is a fast-forwarding path for TCP/UDP packets that pass through the firewall. Packets first traverse the firewall through the normal path. After the system establishes a connection state, the system adds the connection to the flowtable. The system then sends each incoming packet directly from the incoming interface to the outgoing interface. This path bypasses the firewall infrastructure and reduces processing time.

This feature allows 2-3 times faster packet processing. This feature is compatible with QoS and logging and is enabled by default. When you enable IDS or IPS, the system only offloads bypassed traffic to the flowtable. When you enable the limiter within a firewall rule, the system does not add the matching traffic to the flowtable because limiter functionality is not available in the flowtable.

Tip: If you detect errors, disable this feature.

Important: Verify that the **MTU** is the same on each involved interface. Otherwise, connections experience low throughput.

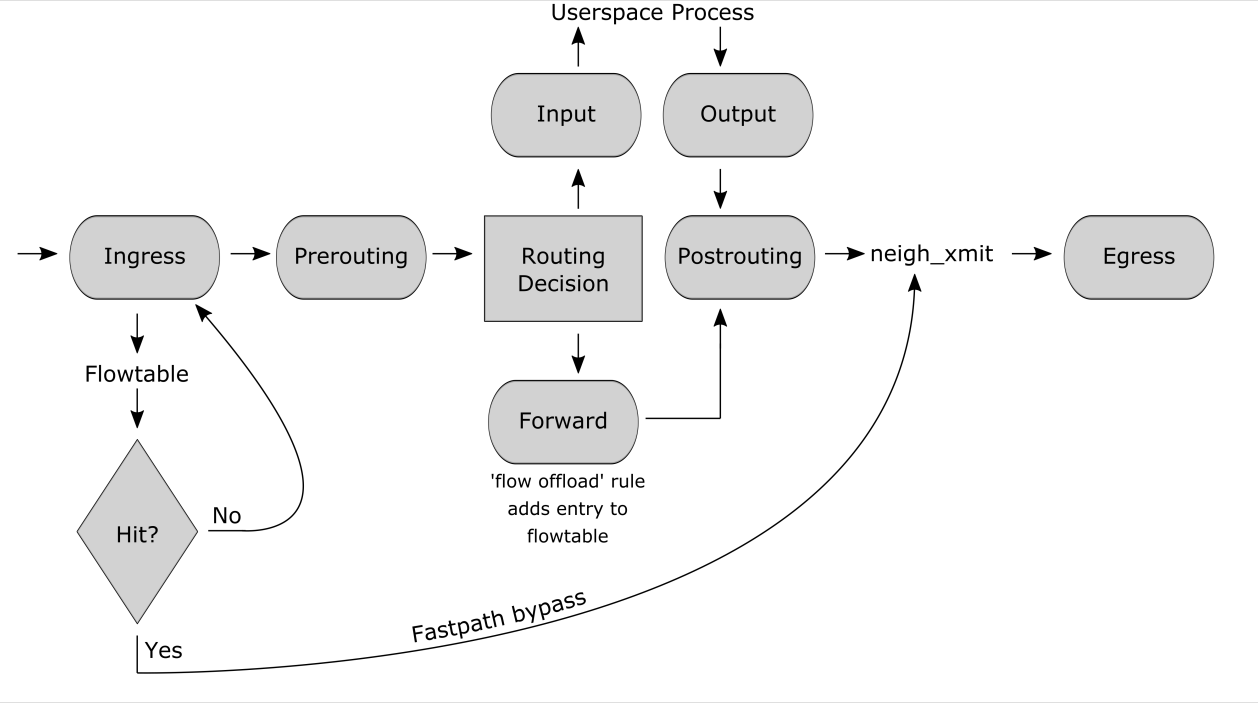


Figure 59: Netfilter hooks and flowtable interactions

General troubleshooting

Troubleshoot slow network performance by adjusting advanced firewall and offload settings.

You experience slow download or upload speeds.

Perform the following steps:

1. Navigate to **System > Settings > Advanced**.

2. **WARNING:** Disabling GRO and flowtable offload can improve network speeds, but reduces firewall performance.

   Try one of the following options:
   - Disable **Firewall Flowtable**
   - Enable **Disable GRO**

   This might improve the network speed.

Licence

This page describes the licence status and activation requirements.

Find the licence under **System > Licence**.

TBF requires a licence to operate. Without a valid licence, the GUI, terminal, and console are read-only.

Official TBF hardware models from Belden include a built-in licence that activates automatically after installation. You do not need to modify the licence in this case.

Portal settings

Portal settings enable a connection to the management portal and provide access to remote management features.

Portal settings let you connect the TBF device to the portal for remote administration and monitoring. Find the portal settings under **System > Portal**.

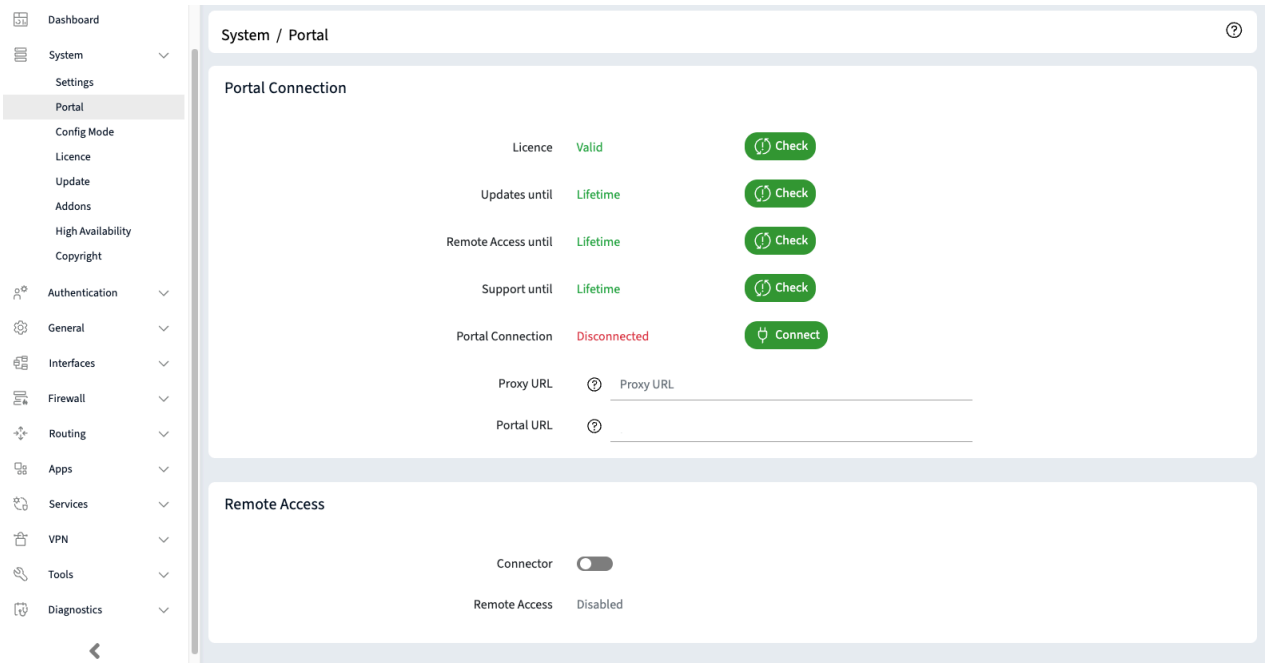


Figure 60: Portal settings

A successful portal connection enables remote device management, automatic updates, and device access from any internet-connected location.

Portal connection

Connect a device to the portal to enable remote administration, licence verification, and automatic updates.

This section shows the TBF licence status and displays when updates are available. This section also lets you verify a portal connection. To use remote access features, establish a portal connection.

Perform the following steps:

1. To manually verify the licence and check for updates, select **Check**.
2. To connect to the portal, enter the login credentials:
   - Email address
   - Portal password

   Note: The user account requires device management permissions in the portal.
3. To link TBF to the portal account, select **Connect**.

   When the portal connection succeeds, the system shows confirmation details in a separate information box.

The device is now connected to the portal and accesses remote management features.

Important: To use the remote access feature through the portal, a portal connection is required.

To disconnect from the portal, select **Disconnect**.

Remote access through the portal

Remote access through the portal assists in establishing security for GUI access from anywhere without firewall configuration changes.

If you have a licence for remote access, then enable the portal connector. The connector URL is `portal.voleatech.de`.

After you enable the portal connector, log in to the portal using the account credentials. Access the GUI of the firewall remotely. The portal connector establishes an encrypted connection to the portal and assists in establishing security for access from the portal back to the GUI.

Note: This remote access functionality requires no firewall rule configuration. Because the firewall initiates the connection, the internet does not need to access the firewall.

The remote access feature provides the following benefits:

- Assists in establishing an encrypted connection to the device
- No firewall configuration changes needed
- Access from any location with internet connectivity
- GUI functionality through the portal

Addons

Addons are optional extensions that enhance the functionality of TBF by creating new GUI items and expanding capabilities.

Find addons under **System > Addons**.

Addons are optional extensions for TBF that enhance the functionality of the system. When you install an addon, the addon creates a new GUI item and expands the capabilities of the system. TBF installs addons automatically in the background, which takes several minutes to complete.

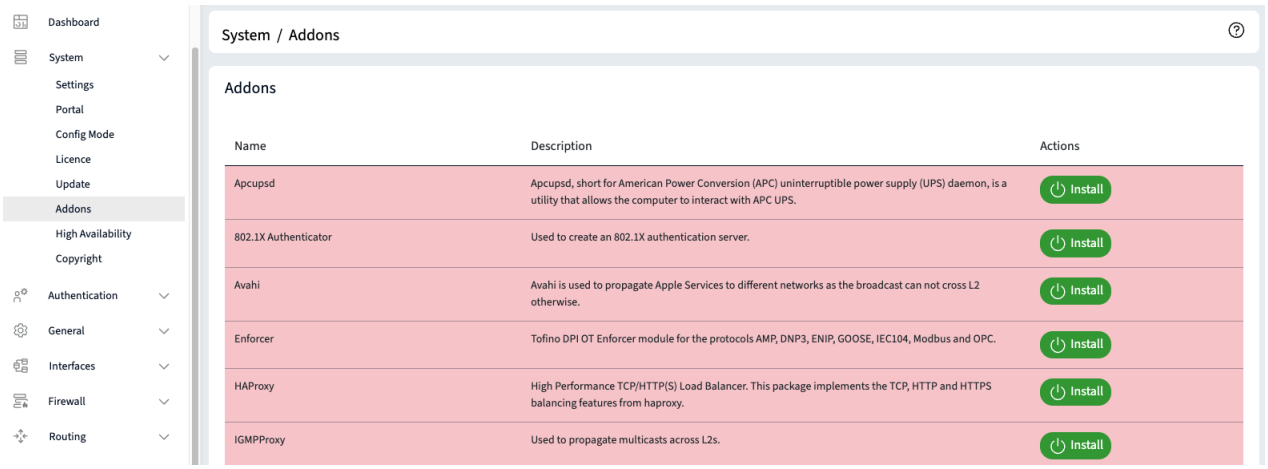


Figure 61: Addons

Note: Installing addons requires an active internet connection and a valid update subscription to connect to the update repository.

The following addons are supported:

- Apcupsd
- 802.1X authenticator
- Avahi
- HAProxy
- IGMPProxy
- NetFlow
- Ntopng
- PPPoE server
- Web application firewall
- WebVPN
- ZeroTier

At the bottom of the page, find the **Check Addons** button. This button verifies whether TBF has installed addon packages correctly. If the system detects any issues, TBF automatically attempts to reinstall the affected packages.

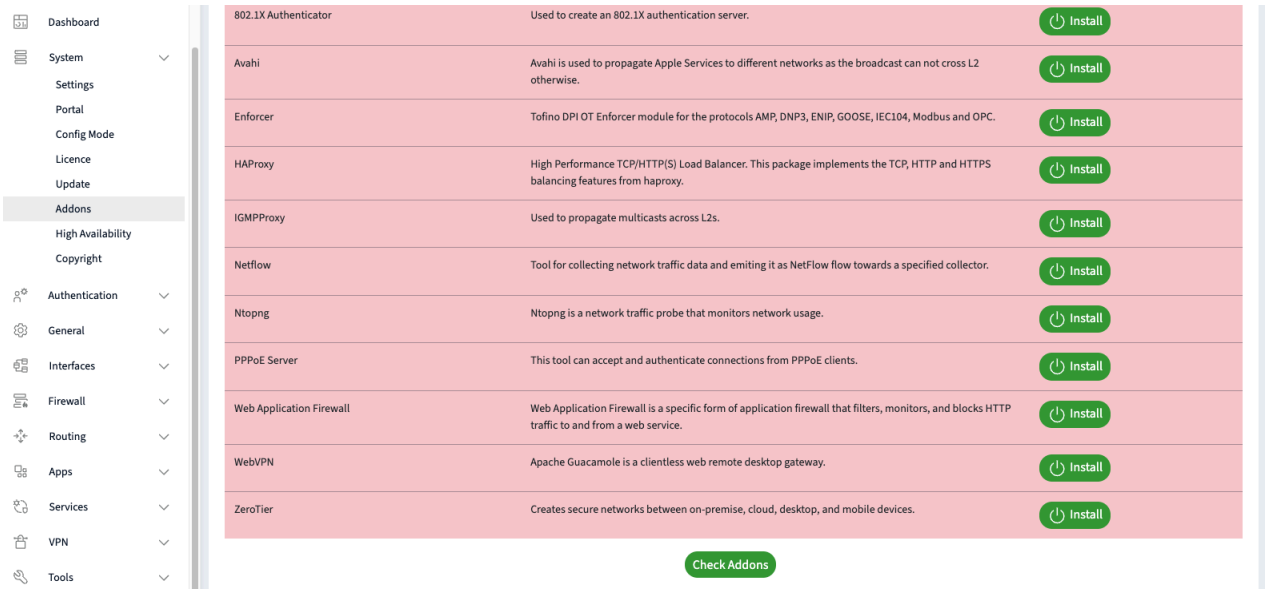


Figure 62: Check addons

Copyright

View copyright information for installed packages in TBF.

Find the copyright information for each installed package under **System > Copyright**.

The **Main** tab displays the packages that TBF directly uses for the GUI and management functions. The **Other** tab lists other packages.

To view more details about a specific package, select its licence name.

8. Authentication

User authentication offers user management, access control, and security features for system access and services.

User authentication encompasses aspects of user management, from basic user accounts to advanced security features. The authentication system controls access to the GUI, system services, and external applications through user and group management.

User authentication includes the following components:

- *User* on page 80: User account management, permissions, and profile settings
- *Group* on page 82: Group-based permission management and organization
- *Authentication server* on page 83: External authentication server integration
- *Identity awareness* on page 86: Link firewall rules with specific users and groups

Set up user authentication during initial system deployment to assist in establishing access control. TBF supports local authentication through the built-in database as well as integration with external authentication servers for enterprise environments.

User

User management controls user accounts, permissions, authentication, and access to system features.

User management encompasses aspects of individual user accounts, from basic account creation to advanced features that assist in establishing security. Find the user settings under **Authentication > User**.

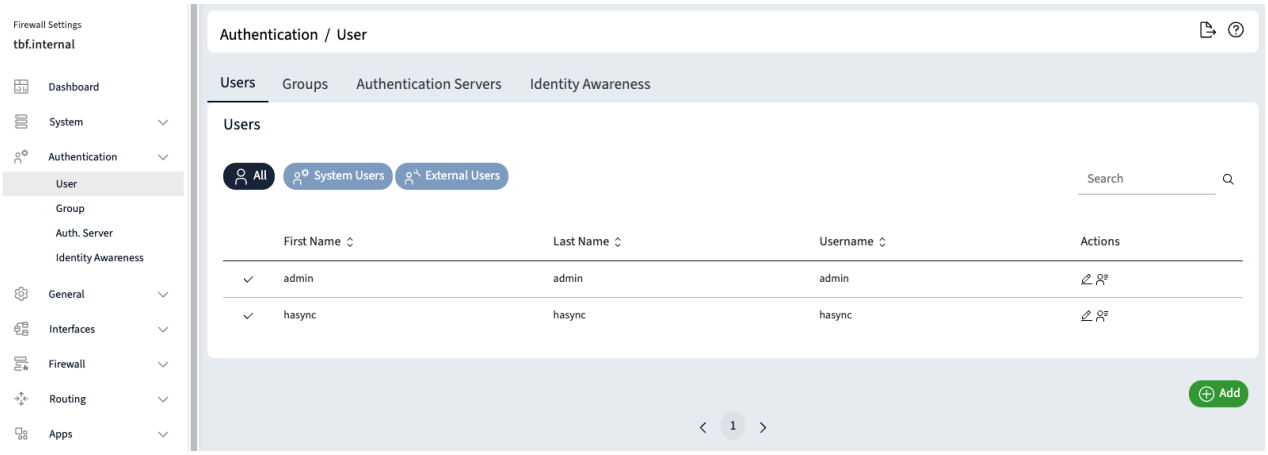


Figure 63: Users

The **Users** tab lets you modify user settings, including:

- Activating or deactivating users
- Deleting users
- Managing user permissions and group memberships

When you create users in the GUI, the Linux system disables them by default. To enable system access, explicitly activate users using the **System Access** option in the user settings.

Users can be members of multiple *groups* on page 82.

Note: To log in to the GUI, users require membership in either the **System Admin** or **System User** group.

Set up other users for services such as OpenVPN or WebVPN.

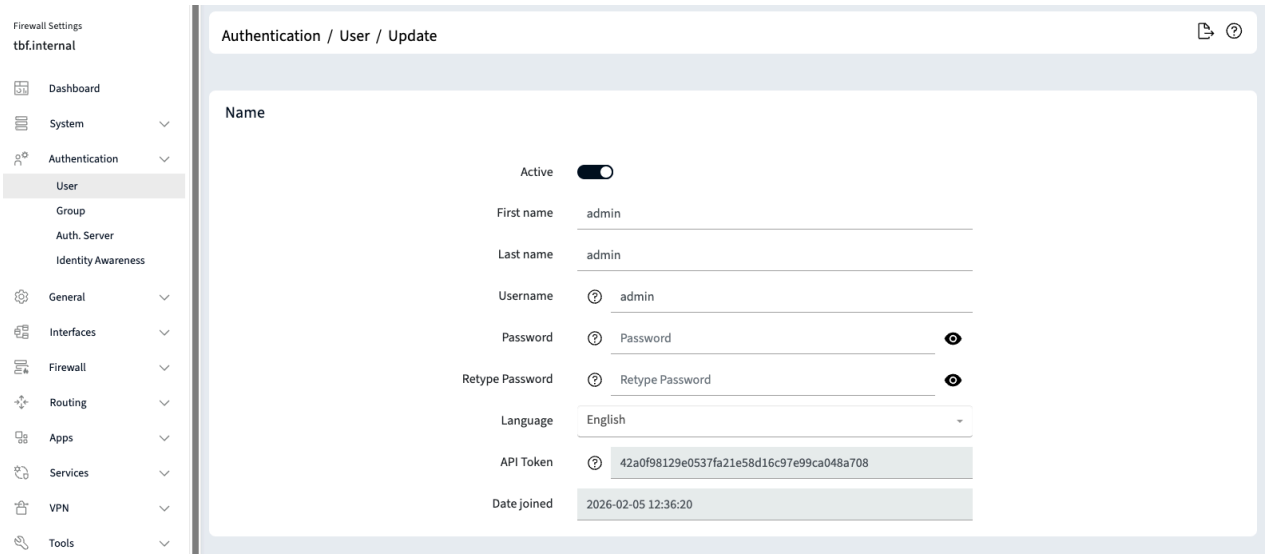


Figure 64: Update user

User management includes the following key areas:

- *User permissions* on page 81: Control access to system features through direct assignment or group membership
- *API token management* on page 81: Enable REST API access without passwords
- *SSH access configuration* on page 82: Set up command-line access using SSH keys
- *Two-factor authentication* on page 82: Add extra layers that assist in establishing security with TOTP and one-time passwords

Additional user features include language customization, bookmark management, OpenVPN profile downloads, certificate association, and profile editing capabilities.

User permissions

User permissions control access to system features.

Each user has a specific set of permissions that determine their access to system features and functions. You can assign permissions directly to individual users or through group memberships. This offers flexible access control options.

TBF combines permissions from both groups and individual users. The additive effect determines the complete set of permissions for each user. This approach offers granular control and keeps permission structures manageable.

Attention: The permissions assigned to each user affect system security. Verify the permissions for each user.

The `admin` user automatically has all permissions, regardless of any permissions you set up in the GUI. This user functions as a superuser with unrestricted access to system features.

API token management

API tokens enable REST API access without passwords while applying permission-based access control.

TBF automatically generates an API token for each user, enabling REST API access without requiring a password. This token-based authentication simplifies the authentication process for automated systems and applications.

Note: Users still require appropriate permissions to access resources through the API. The token serves as an authentication mechanism, but the underlying permission system controls what actions the user can perform.

While API tokens offer convenient access, users can still authenticate using their traditional username and password combination when needed.

SSH access configuration

SSH access configuration enables command-line access using SSH keys for password-free authentication.

Users can add their SSH keys to their profile to enable command-line access to the system. When you set the system settings correctly, users log in without entering a password by using public key authentication.

To allow SSH access, enable the **System Access** setting for the user account. This setting controls whether the user can access the underlying Linux system through SSH or other command-line interfaces.

For users with system access, you can also grant **Sudo Access** to allow them to become `root` and to perform administrative tasks.

CAUTION: Grant this elevated access carefully and only to trusted users who require administrative privileges.

SSH key authentication has several advantages:

- Decreases the possibility of password-based authentication vulnerabilities
- Offers stronger authentication through cryptographic keys
- Enables automated access for scripts and applications
- Allows for centralized key management and revocation

To set up SSH access, users first generate an SSH key pair and add their public key to their user profile. Also enable SSH key authentication in the *settings* on page 63.

Tip: SSH keys are an alternative to passwords for user authentication. Belden recommends rotating SSH keys at defined intervals and removing unused keys from user profiles.

Two-factor authentication

Two-factor authentication adds an extra security layer using one-time passwords and Time-based One-Time Password (TOTP) for GUI and OpenVPN access.

You can enable two-factor authentication for both GUI and OpenVPN access. TBF supports one-time passwords and TOTP as additional authentication layers beyond username and password.

The two-factor authentication system lets you:

- Create and delete authentication elements
- View the TOTP QR code
- Access one-time passwords

TBF automatically deletes used one-time passwords and generates new ones as needed.

Any TOTP-compatible mobile app can generate the required codes for authentication. Popular TOTP apps include Google Authenticator and Authy.

Important: TOTP functionality depends on accurate timekeeping. The TBF clock must be synchronized.

One-time passwords are also useful for granting temporary access to third parties without requiring a password reset. Provide the one-time password along with the regular password for temporary access.

Group

Groups organize users with the same permissions. The permissions you set for a group apply automatically to every user in the group.

Find the group settings under **Authentication > Group**.

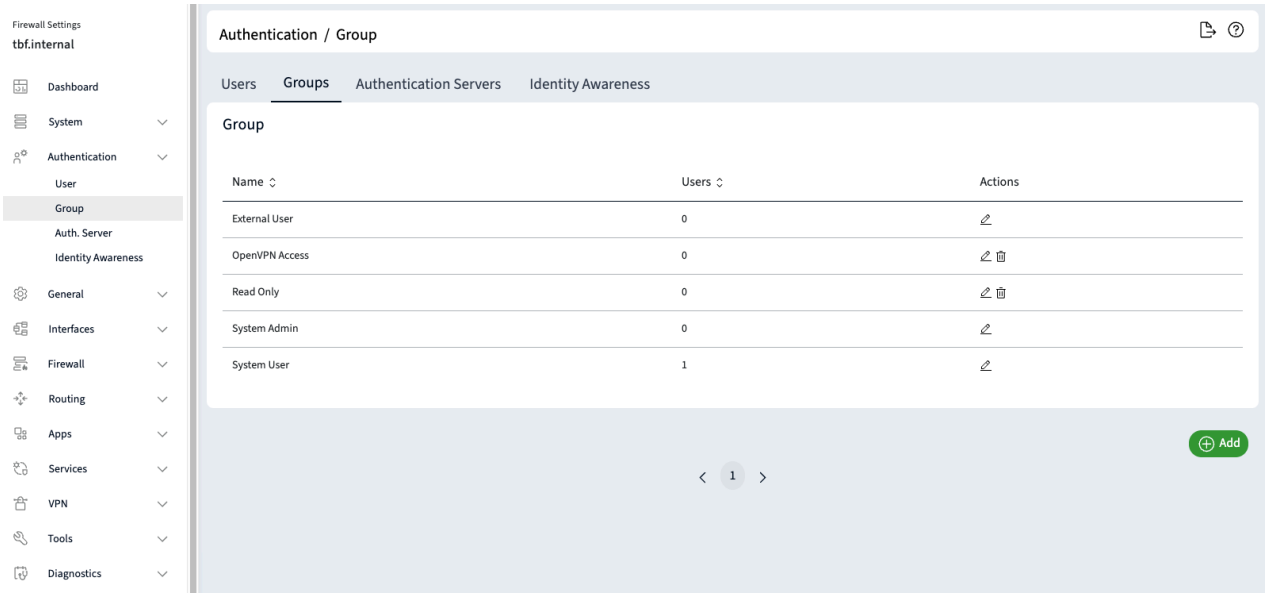


Figure 65: Groups

On the **Groups** tab, edit group settings such as:

- Activate or deactivate groups
- Delete groups

The main purpose of groups is to organize users with the same permissions. For each group, you set the permissions that TBF automatically applies to every user in the group.

For a user to log in to the GUI, TBF requires **System Admin** or **System User** group membership. Use other groups for services such as OpenVPN or WebVPN.

Note: Users in the **Read Only** group have read-only access to the GUI. Add users to the **System User** group for them to have any GUI access.

Authentication server

Authentication servers enable user authentication against external systems like LDAP and RADIUS instead of the local database.

Find the authentication server settings under **Authentication > Auth. Server**.

Authentication servers verify user credentials against external systems instead of the local TBF database. This approach integrates with existing user management systems and offers centralized authentication across the network.

TBF supports the following authentication server types:

- **LDAP**: The Lightweight Directory Access Protocol connects to directory services like Active Directory
- **RADIUS**: The Remote Authentication Dial-In User Service offers network access authentication

After you create an authentication server, assign the authentication server to specific users through the *user settings on page 80*. After you assign the authentication server, users log in with their existing credentials from the external authentication system.

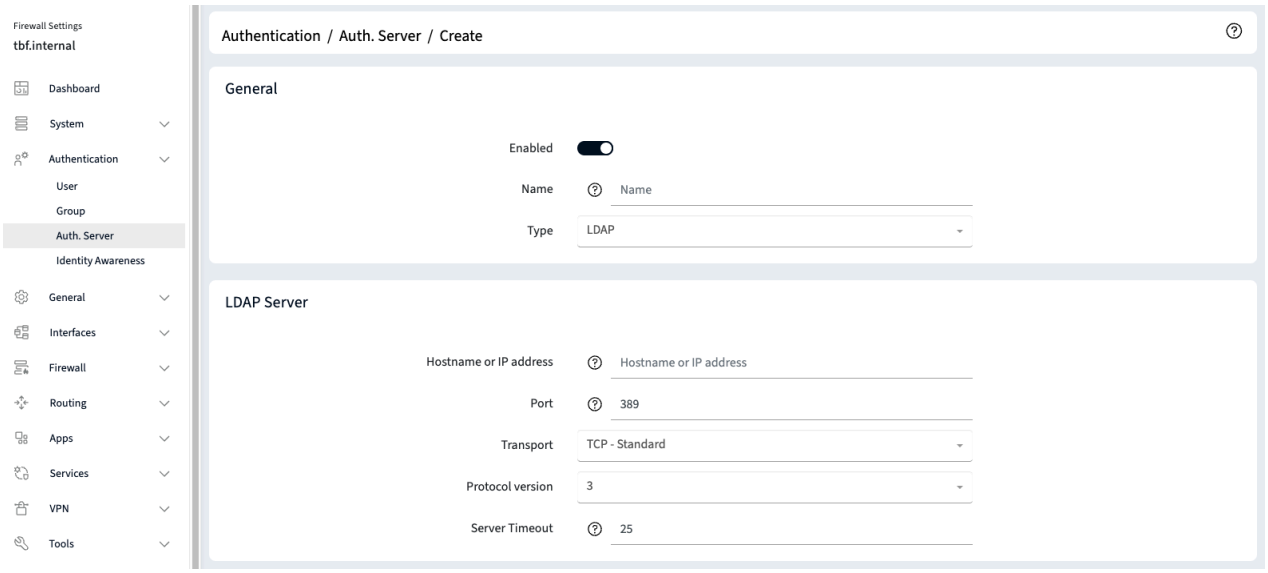


Figure 66: Authentication server creation

Authentication server setup involves several configuration areas:

- *LDAP server* on page 84: Set up connection details for LDAP directory services
- *LDAP scope* on page 85: Specify search parameters to locate user accounts
- *LDAP user* on page 85: Map LDAP attributes to system user properties
- *LDAP advanced* on page 85: Set up advanced LDAP options and group synchronization
- *RADIUS server* on page 86: Define RADIUS authentication parameters

LDAP server

Lightweight Directory Access Protocol (LDAP) server configuration enables authentication against directory services like Microsoft Active Directory or OpenLDAP.

LDAP is a protocol for accessing and maintaining distributed directory information services.

When you set up an LDAP server, specify the following:

Server

This field specifies the hostname or IP address of the LDAP server.

Port

This field specifies the port number for LDAP connections. The default is `389` for standard LDAP and `636` for LDAP over SSL (LDAPS).

Use SSL

Enable this setting to use LDAPS for encrypted connections to the LDAP server.

Bind DN

This field specifies the distinguished name (DN) of the user account that TBF uses to bind to the LDAP server.

Important: This account needs `read` access to the user directory.

Bind Password

This field specifies the password for the bind DN account.

LDAP scope

LDAP scope configuration determines where and how TBF searches for user accounts in the LDAP directory tree.

Settings

Base DN

This field specifies the base distinguished name (DN) is where the search for users begins. Set this DN to the organizational unit or container that contains the user accounts.

User Filter

This field specifies an LDAP filter that defines which objects in the directory are considered users. The default is `(objectClass=person)`.

Search Scope

This field defines the depth of the search:

- `base`: Search only the base DN object.
- `one`: Search the base DN and one level below.
- `sub`: Search the base DN and all levels below (subtree).

LDAP user

LDAP user configuration maps LDAP attributes to TBF user properties for retrieving user information from the directory.

This mapping determines how the system retrieves information from the LDAP directory.

Settings

Username Attribute

This field specifies the LDAP attribute that contains the username. Common values are:

- `uid`
- `sAMAccountName`
- `cn`

Email Attribute

This field specifies the LDAP attribute that contains the email address of the user. The default is `mail`.

Full Name Attribute

This field specifies the LDAP attribute that contains the full name of the user. Common values are:

- `displayName`
- `cn`

Group Attribute

This field is available when you enable group synchronization. This field specifies the LDAP attribute that contains group membership information.

LDAP advanced

Advanced LDAP settings offer additional configuration options for complex directory environments and assist in establishing security.

Settings

Group Synchronization

This setting automatically synchronizes LDAP groups with TBF groups. When you enable this setting, the system automatically assigns users to groups based on their LDAP group membership.

Group Base DN

This field is available when you enable **Group Synchronization**. This field specifies the base DN of the group objects in the LDAP directory.

Group Filter

This field is available when you enable **Group Synchronization**. This field defines an LDAP filter for group objects. The default is `(objectClass=group)`.

Connection Timeout

This field specifies the timeout in seconds for establishing a connection to the LDAP server. The default is `10`.

Search Timeout

This field specifies the timeout in seconds for LDAP search operations. The default is `30`.

RADIUS server

Remote Authentication Dial-In User Service (RADIUS) server configuration enables centralized authentication, authorization, and accounting management for network access.

Settings

Server

This field specifies the hostname or IP address of the RADIUS server.

Shared Secret

The system uses the shared secret key to encrypt communication between TBF and the RADIUS server.

Important: The shared secret is required to match the shared secret set up on the RADIUS server.

Authentication Port

This field specifies the authentication port. The default is `1812`.

Authentication Timeout

This field determines the time in seconds the RADIUS server may take to respond to an authentication request. The default is `5`.

Identity awareness

Identity awareness connects firewall rules to specific users and groups for user-based policy management and automatic IP address tracking.

Find the identity awareness settings under **Authentication > Identity Awareness**.

Identity awareness connects *firewall rules (forward and input)* on page 131 to specific users and groups. This capability helps you create user-based firewall rules and manage policies that recognize individual user identities.

Unlike network objects, TBF automatically updates user IP addresses when the IP addresses change. This reduces the need for manual updates.

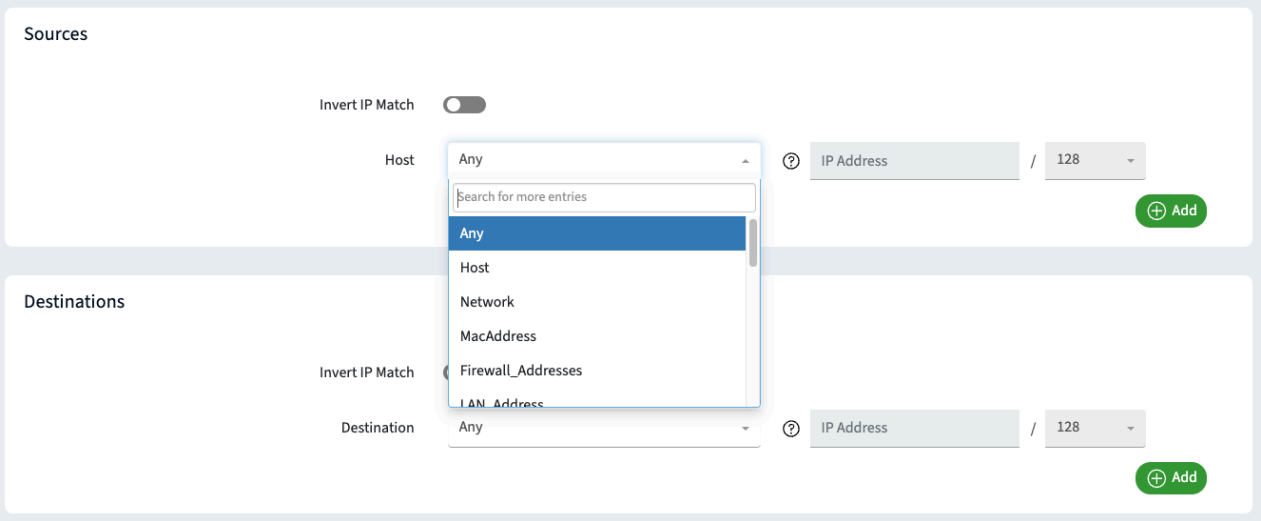


Figure 67: Identity awareness firewall rule

Before using a user in a firewall rule, first create the user or group in TBF.

For larger environments, you can integrate an LDAP or Active Directory server to synchronize users and groups as an *authentication server* on page 80.

Identity awareness settings

These system-wide settings control identity awareness behavior and PC client connections.

Identity awareness settings control how TBF handles user identification and PC client connections for firewall rule enforcement.

Enabled

This setting enables users and groups to be usable in firewall rules.

Captive Portal Enabled

This setting creates a captive portal that allows users to log in and link their current IP address with their user account in TBF. This feature enables dynamic IP address association. TBF automatically updates the IP address of a user when the IP address changes.

PC Client Filter IPs

This setting restricts access to only IP addresses within a specified network range from a firewall interface.

PC Client Connect Password

This field specifies the password that PC clients need to initially connect to the firewall.

Below is a list of registered **PC Clients**. Approve each client manually in the **Actions** column so it connects to TBF.

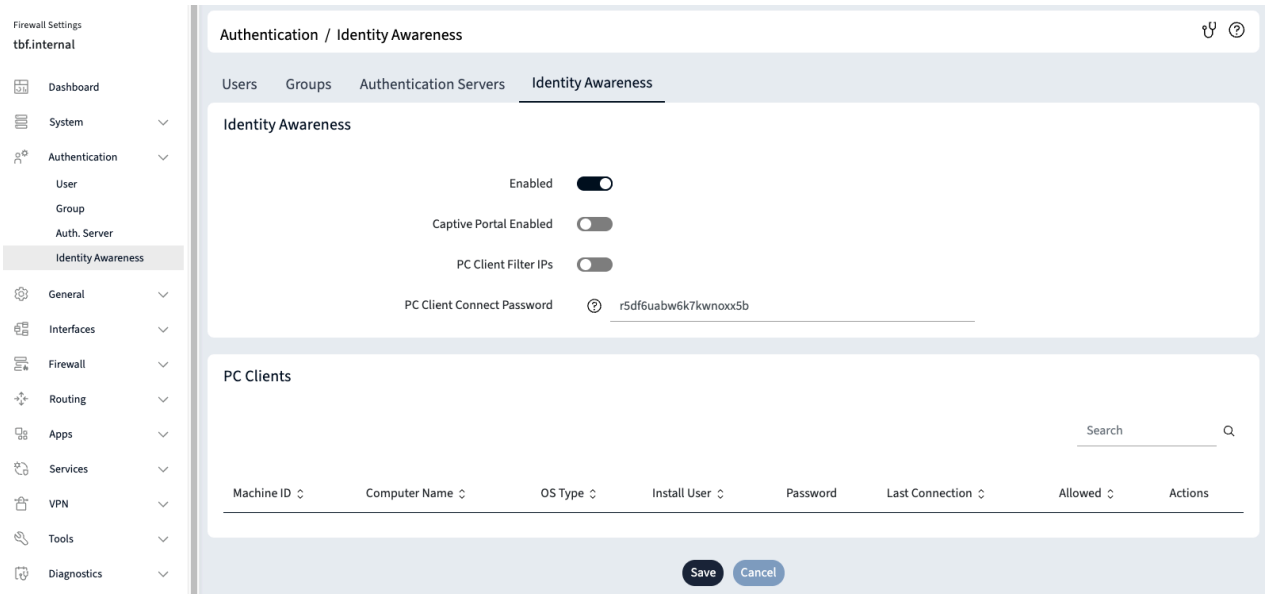


Figure 68: Identity awareness

Identity awareness user settings

Set individual user settings for identity awareness, including IP address assignment and DHCP integration.

With identity awareness user settings, you can assign IP addresses to specific users and connect with DHCP for automatic address management. This feature helps track users across the network and apply firewall rules based on user identity rather than only IP addresses.

Perform the following steps:

1. Navigate to the user settings and find the identity awareness configuration section.



Figure 69: User identity awareness list

2. Select how to assign IP addresses to the user.

   You have two options for IP address assignment:

- Manual IP assignment: Set a fixed IP address that does not change.
- DHCP integration: Allow TBF to assign IP addresses automatically.

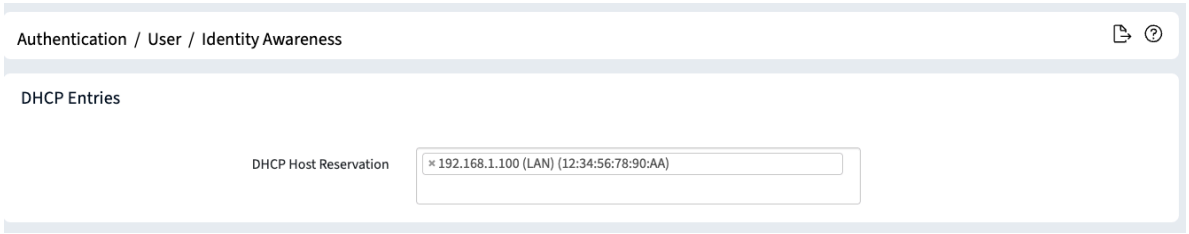


Figure 70: User identity awareness DHCP

3. For manual IP assignment, enter the specific IP address in the user settings.

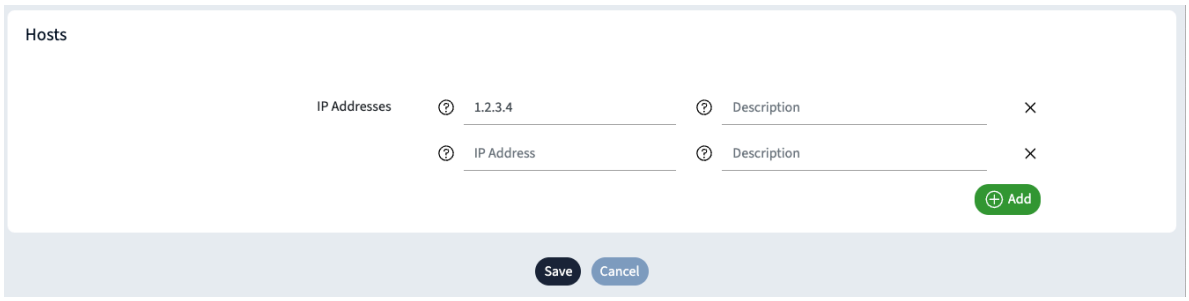


Figure 71: User identity awareness IP

Identity awareness is now enabled for the user. You can use the user in firewall rules with automatic IP address tracking. TBF monitors and applies security policies based on their identity.

9. Interfaces

Network interfaces offer connectivity between a TBF device and networks, and support both physical and virtual interface types for flexible network architecture.

Find the interface settings under **Interfaces**.

Interfaces in TBF use interface objects for abstraction. This abstraction layer separates the physical network hardware from the logical network setup. This separation helps provide flexibility in network design and management.

To enable each network interface, assign it to a TBF interface object. Navigate to **Interfaces > Assign** to complete this assignment process.

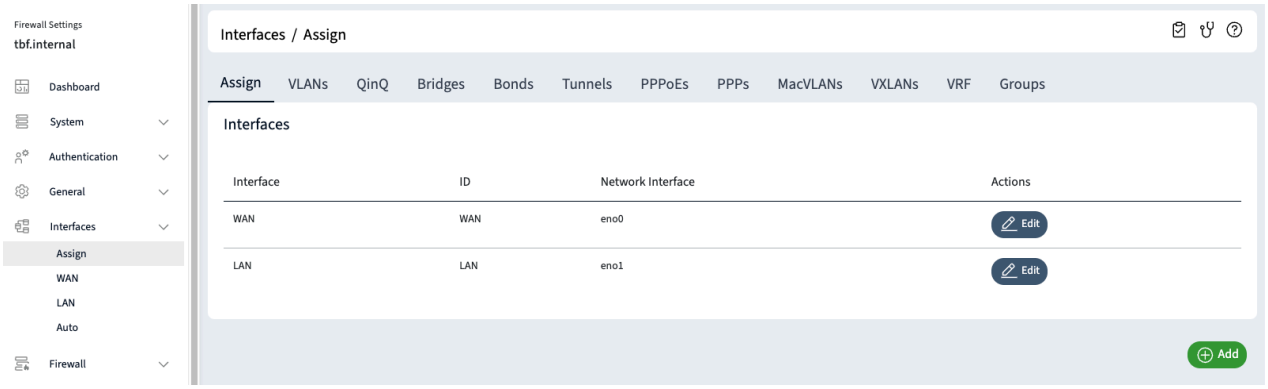


Figure 72: Interfaces

Interface management

Each assigned interface appears as an item in the menu below **Interfaces**, and offers direct access to configuration options for that specific interface.

TBF also automatically generates certain interfaces, such as the app bridge, OpenVPN, and VTI interfaces. View these automatically generated interfaces under **Interfaces > Auto**.

Interface categories

TBF supports several categories of network interfaces:

Basic interface management

- *Assign interfaces* on page 91: Interface assignment and role configuration
- *Set up interfaces* on page 92: Basic interface configuration settings

Virtual LAN interfaces

- *VLAN* on page 96: Virtual LAN configuration for network segmentation
- *QinQ* on page 98: Double VLAN tagging for service provider networks
- *MacVLAN* on page 115: MAC-based virtual LAN interfaces

Link aggregation and bridging

- *Bridge* on page 99: Layer 2 bridging between network segments
- *Bond (link aggregation)* on page 104: Combination of multiple interfaces for redundancy and bandwidth

Tunneling and VPN interfaces

- *Tunnel* on page 107: Generic tunneling interfaces for various protocols
- *VXLAN* on page 122: Virtual Extensible LAN for overlay networks

Point-to-point interfaces

- *PPPoE* on page 109: Point-to-Point Protocol over Ethernet
- *PPP* on page 110: Point-to-Point Protocol for direct connections

DSL interfaces

- *SHDSL* on page 115: Symmetric High-bit-rate DSL interfaces
- *VDSL* on page 119: Very-high-bit-rate DSL interfaces

Advanced networking

- *VRF* on page 121: Virtual Routing and Forwarding for network isolation
- *Interface groups* on page 124: Logical grouping of interfaces for management

Interface configuration workflow

Setting up network interfaces follows this sequence:

1. Review the available physical interfaces and plan the network topology.
2. Assign the interfaces to the appropriate network roles (LAN, WAN, DMZ).
3. Set the basic interface settings (IP addresses, subnet masks, gateways).
4. Set up advanced features like VLANs, bonding, or tunneling as needed.
5. Test the connectivity and monitor the interface performance.

Assign interfaces

Interface assignment enables you to create, modify, and manage logical interfaces that abstract physical network connections.

Find the interface assignment options under **Interfaces > Assign**.

Perform the following steps:

1. Access the interface assignment page through the navigation menu.
2. Create or modify interfaces to match the network layout.

   Note: You cannot delete the **WAN** and **LAN** interfaces, but you can disable them, which effectively deactivates the interface.

3. Set up interface properties including naming and firewall rule creation.

   Each interface has an internal ID. TBF assigns fixed IDs to the **WAN** and **LAN** interfaces. For other interfaces, the ID starts with `INT#`. The number increases sequentially with each new interface you create.

   You can rename any interface. This human-readable name does not affect the internal ID.

The **Internal Interface ID** corresponds to the actual network interface name in the system.

The **Create Default FW Rules** option determines whether TBF automatically creates default firewall rules for the new interface:

- DNS
- ICMP
- local
- internet

TBF creates these default rules for the **LAN** interface by default.

Interfaces in TBF can serve any role and are not restricted to internal, external, LAN, or WAN functions. These are only labels. You can set up interfaces for any functionality you need.

Interface abstraction

Interface abstraction separates logical interface configuration from physical network hardware, helping provide flexibility in network design and management.

Interface abstraction creates a logical layer between the network configuration and the physical network interfaces. This separation allows you to manage network settings independently of the underlying hardware.

Benefits of interface abstraction

Interface abstraction offers several advantages:

- **Hardware independence**: Network configurations remain valid even when physical interfaces change
- **Simplified management**: Rename and reconfigure logical interfaces without affecting physical connections
- **Flexible assignment**: Any logical interface can serve any network role (LAN, WAN, DMZ, etc.)
- **Consistent configuration**: Interface settings persist across hardware changes or replacements

Logical versus physical interfaces

TBF maintains two distinct layers:

Physical interfaces

The actual network ports and hardware connections on the device. These have fixed hardware identifiers.

Logical interfaces

Software-defined interfaces that you create and set up. Map these to any available physical interface.

Interface mapping

When you assign interfaces, you create a mapping between logical interface names and physical network ports. This mapping allows you to:

- Use descriptive names for interfaces, such as *Office Network* instead of *eth0*
- Change physical port assignments without reconfiguring network rules
- Maintain consistent interface references across different hardware configurations

Set up interfaces

Set up interface settings including general configuration, IPv4 and IPv6 settings, advanced options, and interface-specific features.

Set up interface settings under **Interfaces > INTNAME**, where `INTNAME` represents the name of the interface you want to modify.

Note: The available options and settings vary depending on the type of interface you are setting up.

Watch this video tutorial: *Interface Setup Tutorial*

General interface settings

Set basic interface settings including name, MAC address, MTU, and connection speed.

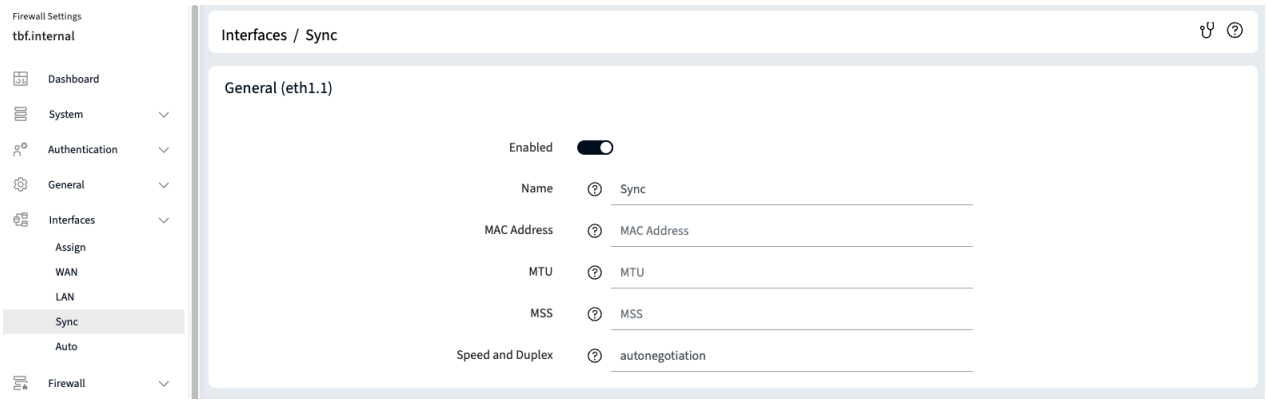


Figure 73: Interface settings

Settings

Enabled

This setting enables or disables the interface.

Name

This field assigns a custom name. The interface name appears throughout the GUI, such as when you define firewall rules for this interface. TBF also displays this name as `ALTNAME` in the shell for every interface.

MAC Address

This field overrides the default MAC address.

Note: Depending on the interface type, this change affects other interfaces. For example, VLANs on an interface share the same MAC address.

MTU

This field specifies the maximum transmission unit. The default is `1500`.

Note: If you plan to use IPv6, then the minimum cannot be less than `1280`.

MSS

This field limits the TCP connection at the specified size. You only need this with PPPoE and PPP connections. TBF generates this value automatically by default for these interface types.

Speed and Duplex

This field determines whether TBF automatically negotiates the connection speed on the interface. You can also manually select a specific speed within the capabilities of the hardware.

Note: For some interface types, such as VLANs, this option is grayed out because the connection speed depends on the underlying physical interface.

IPv4 interface settings

Set up IPv4 configuration including static, DHCP, PPP, and PPPoE options.

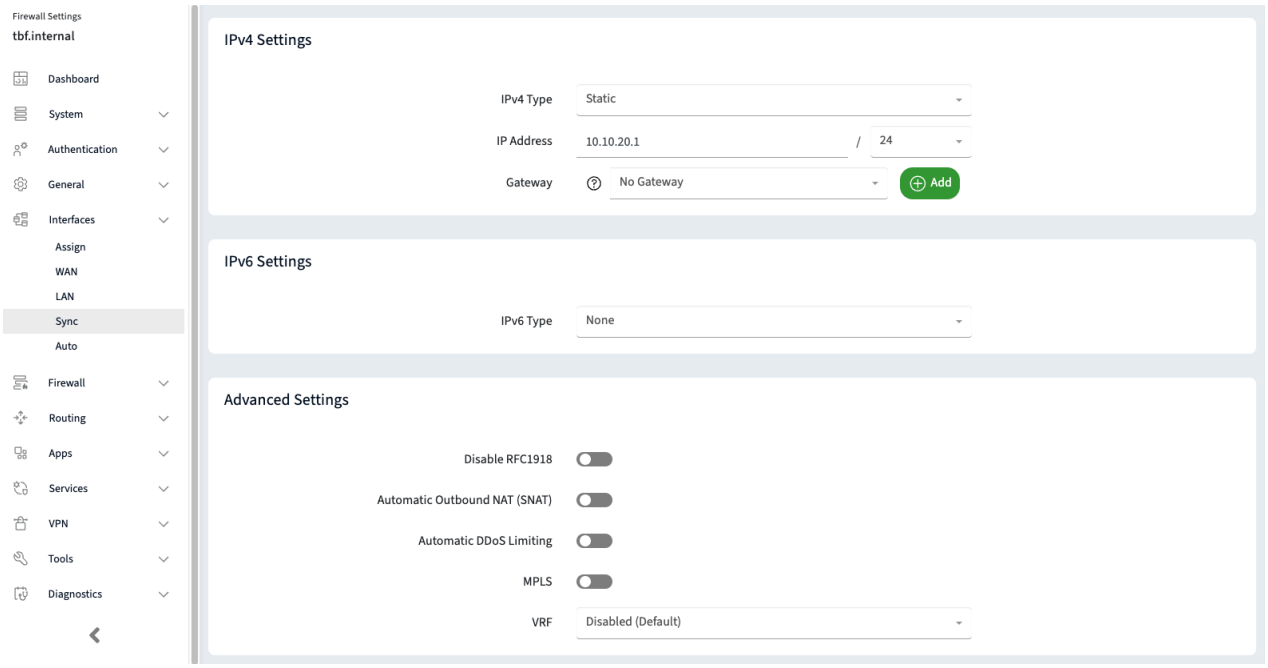


Figure 74: IPV4 settings

The IPv4 settings section contains the IPv4 setup type.

IPv4 Type is one of the following options:

- **None**
- **Static**
- **DHCP**
- **PPP**
- **PPPoE**

Note: Depending on the interface type, you only see a subset of the following options.

DHCP

This setting uses the DHCP client to automatically obtain an IP address and gateway for the interface.

Static

This setting manually defines an IP address and, when applicable, a gateway.

Gateway

This setting is available for static IP addresses and only affects the IP addresses of TBF itself. The IP addresses of TBF use this gateway. This behavior is especially relevant in multi-WAN setups to help keep the interface IP address responsive when the default route uses another interface. Otherwise, routing decisions are based on the **Main** routing table, including for interface IP addresses.

Lease Time

This field is available when **IPv4 Type** is **DHCP**. This field specifies the maximum DHCP lease time in seconds. TBF refreshes the DHCP address after this period expires.

Note: For cellular interfaces, a low value is necessary to accommodate changing cellular connections.

IPv6 interface settings

Set up IPv6 configuration including static, DHCP, SLAAC, and tracking settings.

The IPv6 settings section contains the IPv6 setup type.

IPv6 Type is one of the following options:

- **None**
- **Static**
- **DHCP**
- **SLAAC**
- **Track**

Note: Depending on the interface type, you only see a subset of the following options.

Static

This setting manuallys define an IP address and, when applicable, a gateway.

DHCP

This setting uses the DHCP client to obtain an IP address. This requires a router that advertises itself with router advertisement (RA). You can set **IA_NA** manually. Otherwise, TBF generates one automatically. This is necessary to obtain an IP address from the server. Enable **Prefix Delegation** to also obtain a prefix from the DHCP server that you can then specify on other interfaces via Track. You can set **IA_PD** manually, which is necessary to obtain an IP address from the server. **Prefix Delegation Size** is auto by default. You can also specify the size of the prefix you want to request.

SLAAC

This setting obtains the IP address using the IPv6 Stateless Address Autoconfiguration (SLAAC) mechanism. Confirm that at least one IPv6 router advertises itself in the network.

Track

This setting obtains an IP address via prefix delegation from another interface. Select the interface that receives a prefix. The **IPv6 Prefix ID** determines which of the delegated /64 subnets this interface uses when the ISP has delegated more than one prefix via DHCPv6. For example, if the ISP provides a /60 delegation, 16 /64 networks are available, allowing you to use prefix IDs from 0 through 15.

Router Advertisement (Client)

This setting either obtains the IP address when you enable **SLAAC** or it obtains the gateway. The RA mechanism obtains the gateway in IPv6, even when using DHCP. If you disable router advertisement, this mechanism is disabled.

Gateway

This setting is available for static IP addresses and only affects the IP addresses of TBF itself. The IP addresses of TBF use this gateway. This is particularly relevant in multi-WAN setups to help keep the interface IP address responsive when the default route uses another interface. Otherwise, routing decisions are based on the **Main** routing table, including for interface IP addresses.

Advanced interface settings

These settings set advanced interface options that include RFC1918 blocking, NAT, DDoS limiting, MPLS, and VRF.

Settings

Disable RFC1918

This setting automatically creates a firewall rule to block the IP addresses in the RFC1918 range. See the *built-in network ranges* on page 128 for a description of RFC1918. Private networks use these IP addresses. It is useful to block these IP addresses from communication on the **WAN** interface, for example.

Automatic Outbound NAT (SNAT)

This setting automatically creates an SNAT rule for this interface.

Note: This setting requires special consideration in high-availability (HA) setups. Disable this setting on the **WAN** interface and manually set this setting as described in *HA outbound NAT* on page 382.

Automatic DDoS Limiting

This setting creates a rule to limit connections per incoming host per minute. Set the number of connections per minute in the global settings.

MPLS

This setting enables the MPLS functionality. TBF disables this setting by default.

VRF

This setting specifies a virtual routing and forwarding (VRF). TBF disables this setting by default.

Wi-Fi client setup

Set up the SSID and the password to connect a Wi-Fi interface to a wireless network.

When the interface is a Wi-Fi interface, set up the SSID and the password to connect to a Wi-Fi network.

WPA authentication protocols

Set up WPA authentication with various protocols including EAP-TLS, EAP-PEAP, and EAP-TTLS.

To enable WPA authentication (802.1X), select a **Protocol**. The following protocols are available:

- EAP-TLS
- EAP-PEAP/MSCHAPv2 (both PEAPv0 and PEAPv1)
- EAP-PEAP/TLS (both PEAPv0 and PEAPv1)
- EAP-PEAP/MD5-Challenge (both PEAPv0 and PEAPv1)
- EAP-TTLS/EAP-MD5-Challenge
- EAP-TTLS/EAP-MSCHAPv2
- EAP-TTLS/EAP-TLS
- EAP-TTLS/MSCHAPv2
- EAP-TTLS/MSCHAP
- EAP-TTLS/PAP
- EAP-TTLS/CHAP
- EAP-MD5-Challenge
- EAP-MSCHAPv2

The **Anonymous Identity** is optional. For authentication purposes, TBF requires an **Identity** and a **Password**.

When you select a protocol that supports PEAP or TTLS, TBF requires a **Certificate Authority** and a **Certificate**.

Note: The command-line utility `wpa-cli` displays information regarding the WPA authentication status. Event-driven commands are available for authentication or deauthentication events. Belden support is available via *my.belden.com*.

Interface-dependent options

Some interfaces have additional options specific to their type, such as bridge, PPPoE, and PPP settings.

Some interfaces have additional options. For example, on a *bridge* on page 99, change the *RSTP* on page 99 settings as well as the *bridge port settings* on page 99.

On *PPPoE* on page 109 and *PPP* on page 110 interfaces, modify specific settings.

VLAN

These settings set up 802.1q VLANs on physical interfaces, bonds, and bridges to segment network traffic.

Find the VLAN settings under **Interfaces > Assign > VLANs**.



Figure 75: Vlans

Supported VLAN types

TBF supports 802.1q VLANs, which are an industry standard for VLAN tagging.

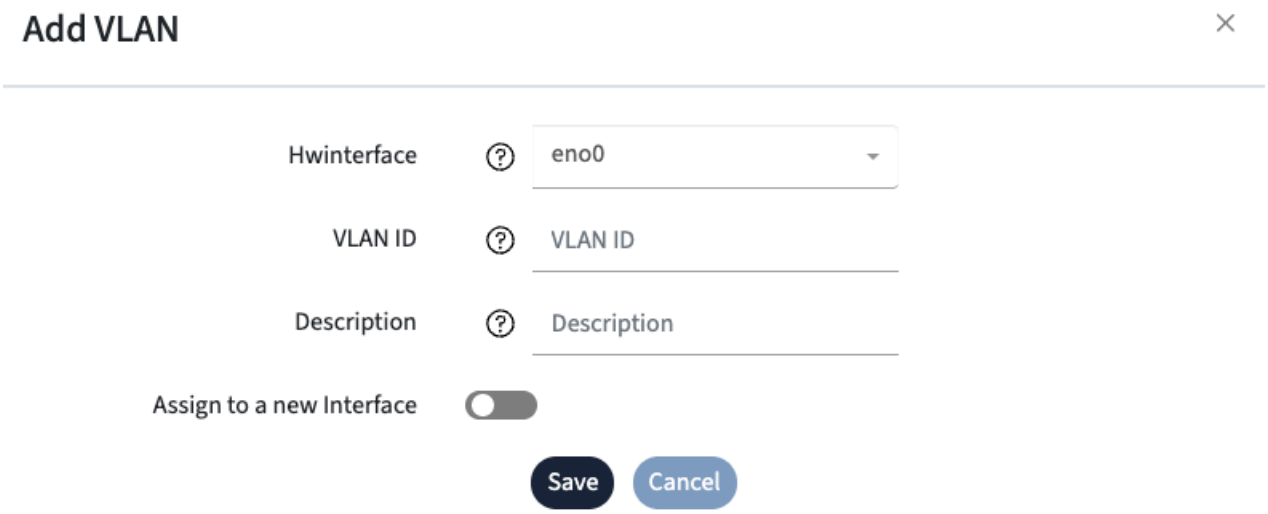


Figure 76: Add VLAN

Compatible interface types

You can create VLANs on these interface types:

- Physical interfaces
- Bonds
- Bridges

Note: Bridges handle VLANs differently than other interfaces. Set up their VLANs directly in the bridge settings rather than as separate VLAN interfaces.

A VLAN on top of a bridge is useful when you need an IP address in that specific VLAN. The bridge continues to route VLANs at layer 2 and forwards the VLANs through the set up bridge ports.

Settings

Hwinterface

This setting determines the parent interface on which to create the VLAN.

VLAN ID

This field specifies the VLAN identifier that tags the traffic on the selected interface. This value ranges from `1-4094`.

Description

This field adds an optional description to help identify the purpose of the VLAN.

Assign to a new Interface

This setting appears only when you create a new VLAN. This setting automatically assigns the VLAN interface as described in *assign interfaces* on page 91.

Video tutorial

Watch this video tutorial for a step-by-step guide to VLAN setup: *VLAN Setup Tutorial*

QinQ

Set up 802.1ad QinQ double VLAN tagging on physical interfaces, bonds, and bridges for service provider networks.

Find the QinQ settings under **Interfaces > Assign > QinQ**.

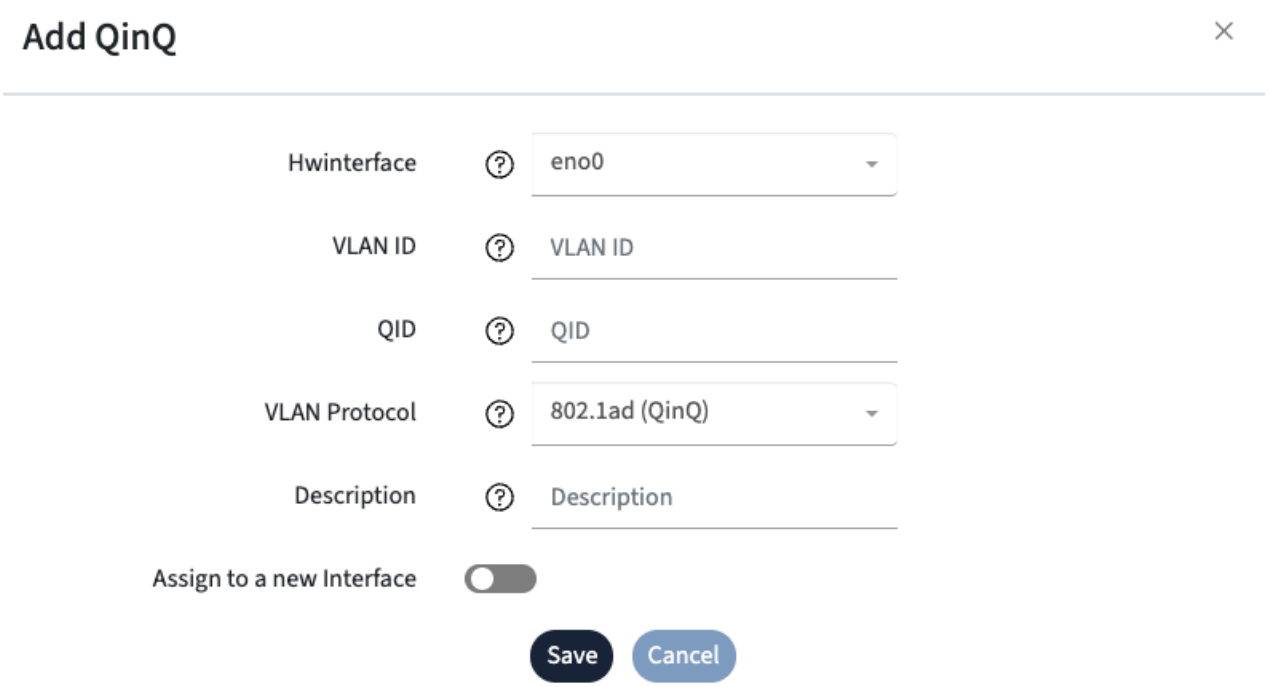


Figure 77: Add QinQ

Supported QinQ types

TBF supports:

- 802.1ad QinQs
- 802.1q VLANs

Compatible interface types

You can set up QinQs only on the following interface types:

- Physical interfaces
- Bonds
- Bridges

Settings

Hwinterface

The setting determines the interface for the QinQ.

VLAN ID

This field specifies the inner VLAN identifier for the selected interface. This value ranges from `1-4094`.

QID

This field specifies the outer VLAN identifier (Q-in-Q ID) for the interface. This value ranges from `1-4094`.

VLAN Protocol

This setting specifies the outer VLAN tag. The default is the QinQ tag, but some setups still use the VLAN tag.

Description

This field gives the interface a description.

Assign to a new Interface

When you create a new QinQ, this setting automatically assigns the interface according to the process described in *assign interfaces* on page 91.

Bridge

These settings set up layer 2 bridges to connect network segments and enable VLAN-aware switching with STP/RSTP support.

Find the bridge settings under **Interfaces > Assign > Bridges**.

Bridges support VLAN awareness, which lets you define VLANs and assign the VLANs to ports as either tagged or untagged traffic.

Note: Each port can have one untagged VLAN.

Also use a bridge as a tunnel endpoint. For more information, see *tunnel* on page 107 and *bridging scenarios* on page 103.

Create a new bridge

This task describes how to create a new bridge interface.

To create a new bridge, select **Add**.

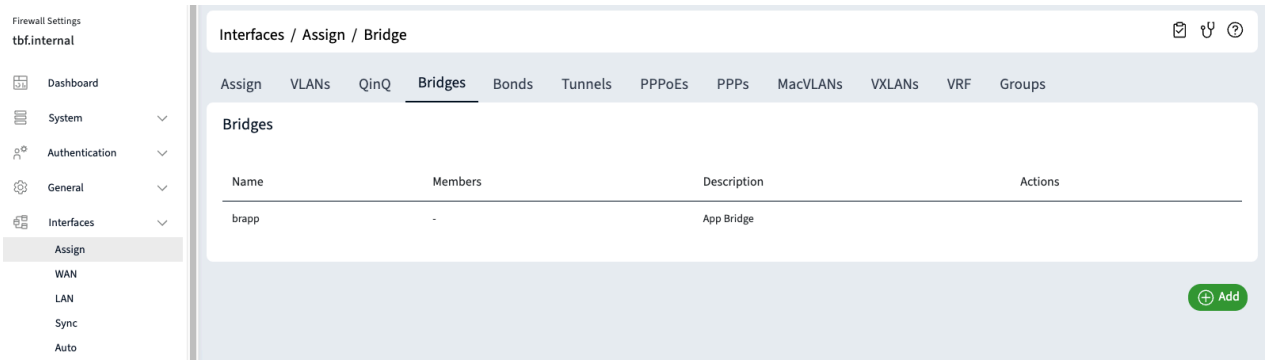


Figure 78: Bridges

Perform the following steps:

1. Enter a **Description** for the bridge.
2. Select the interfaces that you want to add to the bridge.
3. To enable the Spanning Tree Protocol, select **Enable STP**.

   When you select **Enable STP**, set the following settings:

   - **Protocol**: Selection of the STP protocol. The default is **rstp**.
   - **Priority**: Setup of the bridge priority. The default is `32768`.
   - **Max Age**: Setup of the maximum age for the bridge. The default is `20`.
   - **Hello Time**: Setup of the hello time for the bridge. The default is `2`.
   - **Forward Delay**: Setup of the forward delay for the bridge. The default is `15`.
4. To create the bridge, select **Save**.

After you create the bridge, assign the bridge to an interface. For more information, see *assign interfaces* on page 91.

Bridge VLAN

This page describes how to configure VLANs on a bridge interface.

Set up VLANs on a bridge. To do so, select the bridge and then select **Edit**.

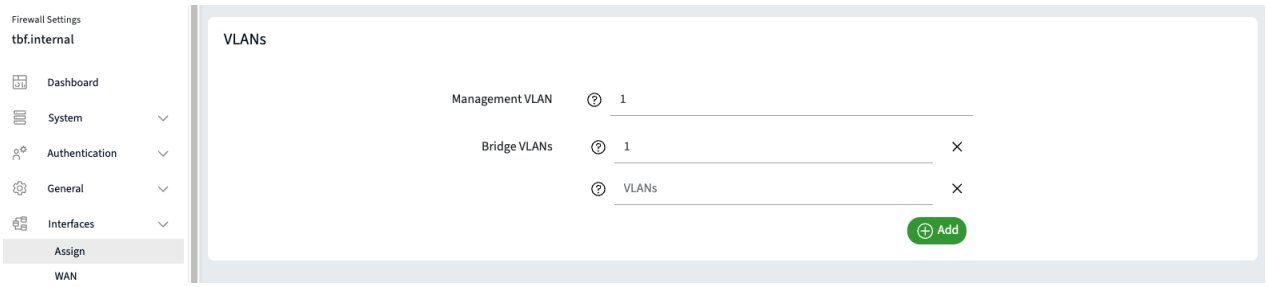


Figure 79: Bridge VLAN settings

Perform the following steps:

1. In the **VLANs** section, set the following settings:

   VLAN ID

   This setting specifies the VLAN ID for the bridge.

   Tagged Ports

This setting specifies the ports that are tagged for the VLAN.

Untagged Ports

This setting specifies the ports that are untagged for the VLAN.

2. To save the VLAN setup, select **Save**.

STP/RSTP

This page describes STP/RSTP settings for bridge interfaces.

Set STP/RSTP settings for a bridge. To do so, select the bridge and then select **Edit**.

Settings

Enable STP

This setting enables or disables STP for the bridge.

Protocol

This setting specifies the STP protocol. The default is **rstp**.

Priority

This field specifies the bridge priority. The default is `32768`.

Max Age

This field specifies the maximum age for the bridge. The default is `20`.

Hello Time

This field specifies the hello time for the bridge. The default is `2`.

Forward Delay

This field specifies the forward delay for the bridge. The default is `15`.

To save the STP/RSTP configuration, select **Save**.

Bridge port settings

This page describes the port settings for bridge interfaces.

Set the port settings for a bridge. To set the port settings, select the bridge and then select **Edit**.

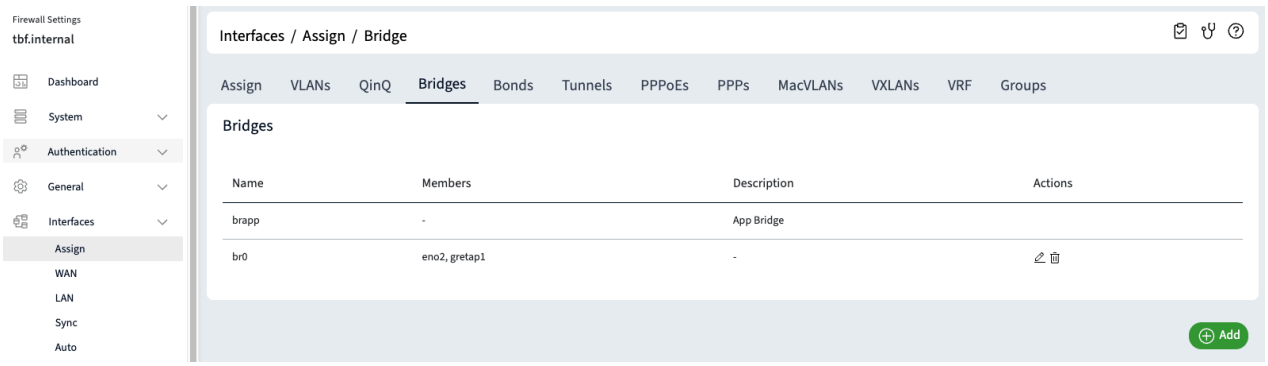


Figure 80: Bridge port settings

Settings

Port

This field specifies the port name.

Priority

This field sets the port priority. The default is `128`.

Path Cost

This field sets the path cost for the port. The default is `0` (auto).

Admin Edge

This setting enables or disables the admin edge for the port.

Auto Edge

This setting enables or disables the auto edge for the port.

P2P

This setting sets the point-to-point setting for the port.

To save the port settings, select **Save**.

Bridge port STP/RSTP settings

This page describes STP/RSTP settings for individual bridge ports.

Set the STP/RSTP settings for individual bridge ports. To set the STP/RSTP settings, select the bridge and then select **Edit**.

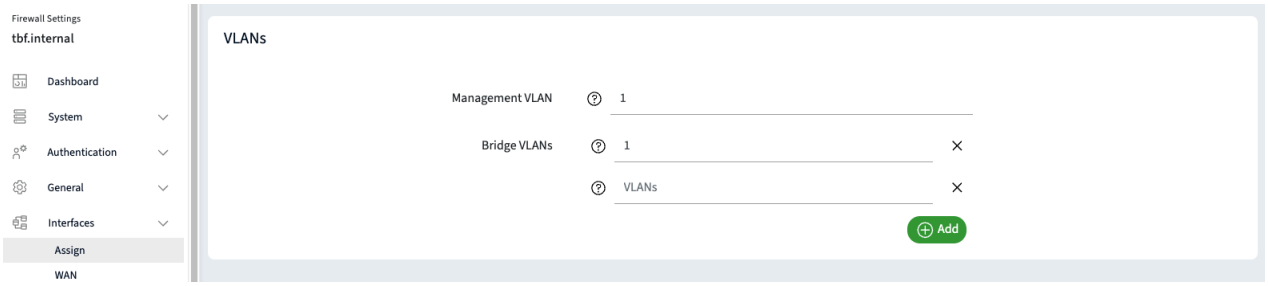


Figure 81: Bridge port STP/RSTP settings

Settings

Port

This field specifies the port name.

Enable

This setting enables or disables STP/RSTP for the port.

Priority

This field sets the port priority for STP/RSTP. The default is `128`.

Path Cost

This field sets the path cost for STP/RSTP. The default is `0` (auto).

To save the port STP/RSTP settings, select **Save**.

Compatibility

This page lists the interface types compatible with bridges.

Bridges are compatible with the following interface types:

- Physical interfaces
- VLANs
- Bonds
- Tunnels

You can also create VLANs on top of bridges.

Note: When you create a VLAN on top of a bridge, the device creates the VLAN as a separate interface. This behavior differs from setting up VLANs within the bridge itself.

Bridging scenarios

This topic describes common bridging scenarios and how to set them up.

Site-to-site bridging with GRETAP

Use GRETAP tunnels to create site-to-site bridges. This approach extends a layer 2 network across multiple sites.

To create a site-to-site bridge with GRETAP:

1. Create a GRETAP tunnel. For more information, see *tunnel* on page 107.
2. Create a bridge.
3. Add the GRETAP tunnel and local interfaces to the bridge.
4. Set firewall rules to allow GRE traffic between the sites.

To set firewall rules for GRETAP, allow GRE traffic between the tunnel endpoints. GRE uses protocol 47.

To create a firewall rule for a tunnel endpoint:

1. Navigate to **Firewall > Rules > WAN**.
2. Select **Add**.
3. Select **GRE** as the **Protocol**.
4. Enter the remote public IP address of the GRETAP tunnel as the **Host**.
5. Select **WAN_Address** as the destination.
6. Save the firewall rule.

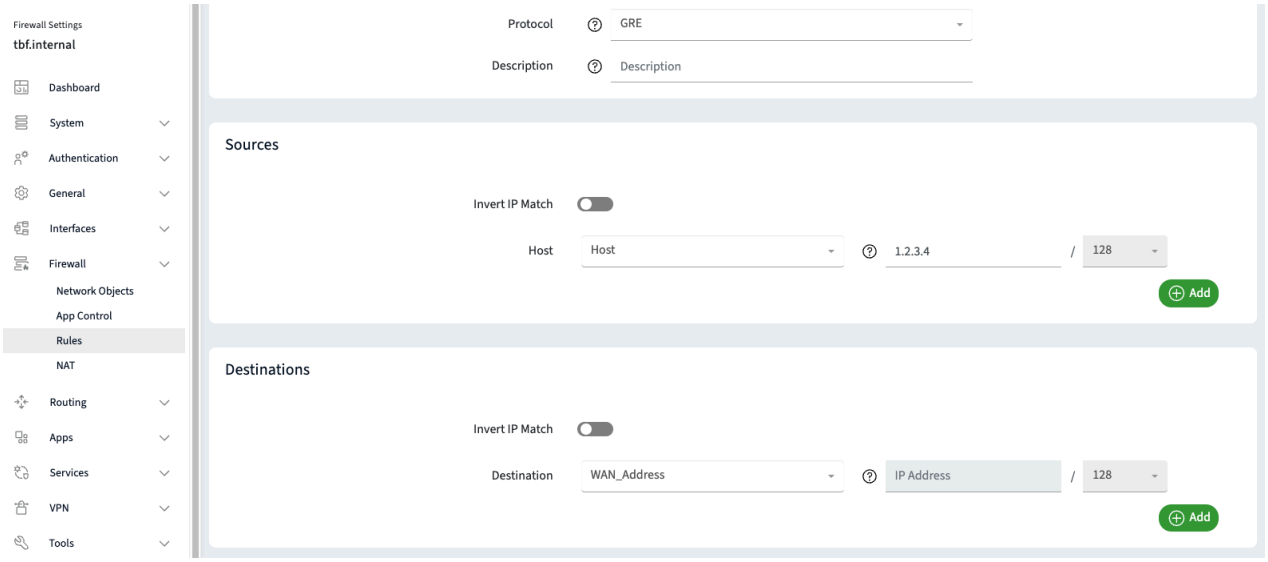


Figure 82: Firewall rule creation

Tip: To encrypt site-to-site traffic, implement an IPsec VPN on top of the GRETAP tunnel. For more information, see *GRE over IPsec* on page 335.

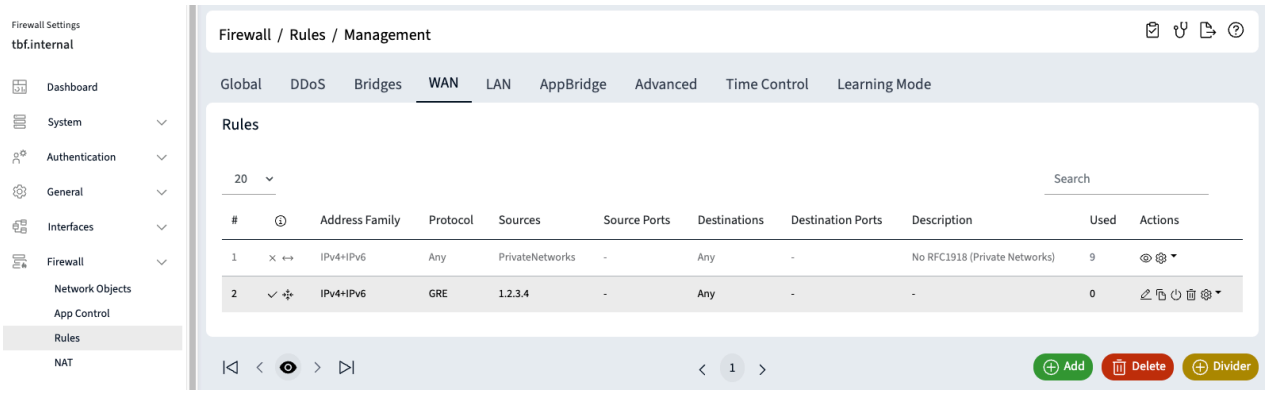


Figure 83: Firewall rule overview

Bond (link aggregation)

Bond interfaces combine multiple physical interfaces into one logical interface for increased bandwidth, redundancy, or both.

TBF combines multiple physical interfaces into a single logical interface through bonding. Bonding increases the maximum bandwidth, helps provide network redundancy, or delivers both benefits.

When you create a new bond, the system hides physical interfaces that existing interfaces like **WAN** or **LAN** already use. Bond only physical interfaces that remain unassigned to any interface.

Note: TBF automatically creates the **WAN** and **LAN** interfaces by default. To use a physical interface that **LAN** currently uses in a bond, set the **LAN** interface to `none` in its settings. Then, create the bond. This step frees up the physical interface for bonding.

Create and set up bonds

Create bond interfaces to combine multiple physical interfaces for increased bandwidth or redundancy.

This combination helps provide either increased bandwidth by connection aggregation or improved reliability through redundancy. Access the bond settings under **Interfaces > Assign > Bonds**.

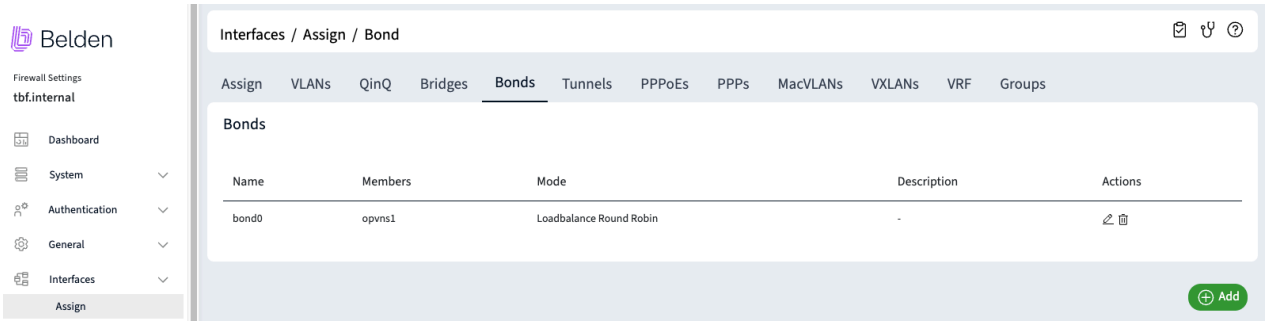


Figure 84: Bonds

Perform the following steps:

1. Navigate to **Interfaces > Assign > Bonds**.

   To start the creation of a new bond interface, select **Add**.

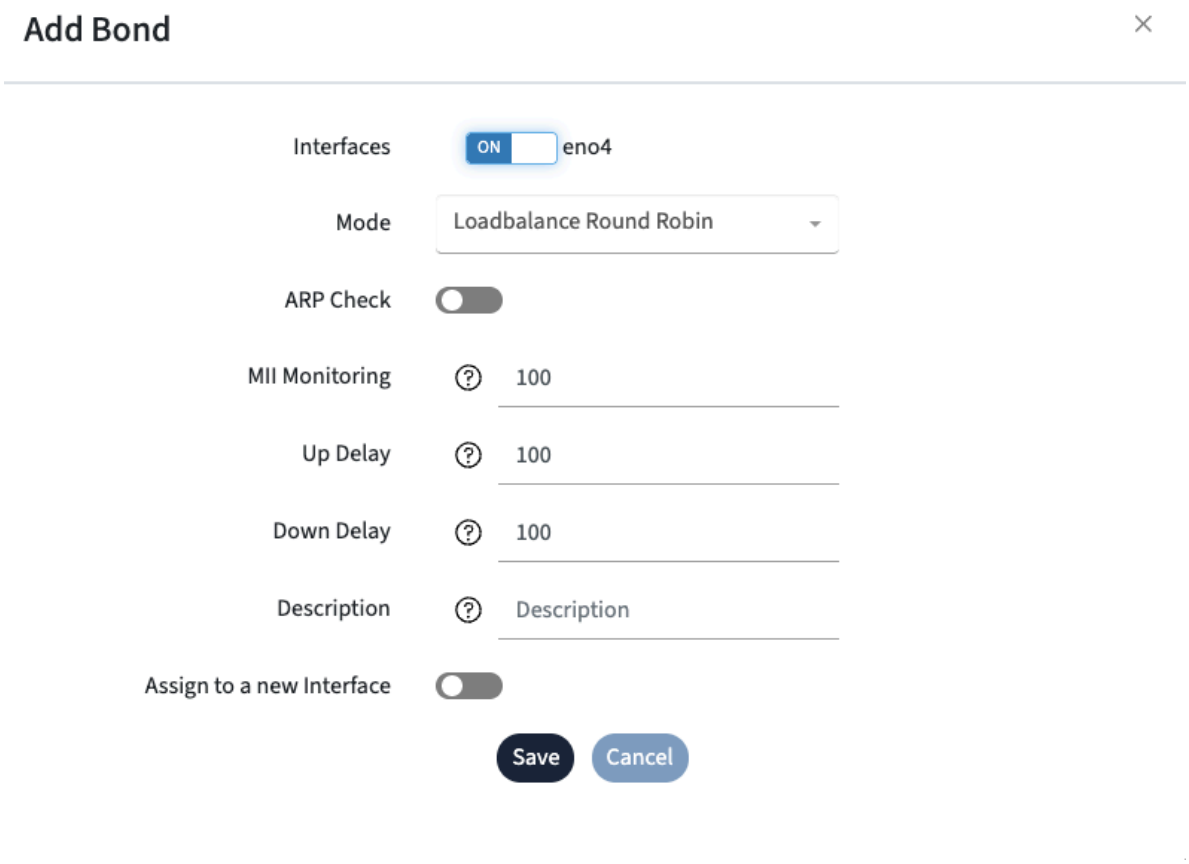


   Figure 85: Add bond

2. Set up the bond settings based on network requirements.

   The bond settings determine how the interfaces work together. For complete details about each setting and its purpose, see the *bond settings* on page 106.

3. Select which physical interfaces to include in the bond.

   Select the network interfaces that you want to combine into the bond. Modify this selection later with the **Edit** or **Add** options for the bond.

4. For **Active/Backup** mode, specify which interface serves as the primary active connection.

   In **Active/Backup** mode, only one interface carries traffic while others remain on standby. Select the interface to be active by default.

5. Optionally, assign the bond to a network interface automatically.

   To automatically set up the bond for immediate use, enable the **Assign to a new Interface** setting. For more information about interface assignment, see *assign interfaces* on page 91.

TBF creates the bond interface and makes it available for network traffic. The combined interfaces work together according to the bond mode you selected.

For a visual walkthrough of the bond setup process, watch this tutorial video: *Bond Interface Setup Tutorial*

Bond settings

This topic describes bond configuration settings and their available options.

Settings

Mode

This setting determines how the bond interface operates and distributes traffic across member interfaces. Select from these bond modes:

- **Loadbalance Round Robin** (`balance-rr`): This option distributes packets sequentially across all active interfaces.
- **Active/Backup** (`active-backup`): This option uses one active interface with others as backups.
- **Load Balance with XOR** (`balance-xor`): This option distributes packets using an XOR hash calculation.
- **LAG (802.3ad)** (`802.3ad`): This option creates dynamic link aggregation groups following the IEEE 802.3ad standard.
- **Adaptive transmit load balancing** (`balance-tlb`): This option balances outgoing traffic using the current load.
- **Adaptive Loadbalancing** (`balance-alb`): This option balances both incoming and outgoing traffic.

Hash Policy

This setting controls how the system distributes packets between bond member interfaces by specifying which packet information the system uses for load balancing calculations. This setting applies only to the following modes:

- **Load Balance with XOR**
- **LAG (802.3ad)**
- **Adaptive transmit load balancing**

Available options include:

- **Layer 2**: This option uses MAC addresses for distribution.
- **Layer 2 + 3**: This option uses MAC and IP addresses for distribution.
- **Layer 3 + 4**: This option uses IP addresses and port numbers for distribution.

ARP Check

This setting enables ARP monitoring that uses an IPv4 address instead of the default MII monitoring for detecting bond member outages. This setting works with the following modes:

- **Loadbalance Round Robin**
- **Adaptive transmit load balancing**

MII Monitoring

This field sets the interval in milliseconds for monitoring the physical link status of bond member ports using Media Independent Interface (MII) checks.

Up Delay

This field specifies the delay time in milliseconds before the system activates a bond member port after it becomes available. This field works in conjunction with **MII Monitoring** to help prevent rapid state changes.

Down Delay

This field specifies the delay time in milliseconds before the system deactivates a bond member port after it becomes unavailable. This field works in conjunction with **MII Monitoring** to help prevent rapid state changes.

Assign to a new Interface

This setting automatically assigns selected interfaces to the new bond during creation. For detailed information about interface assignment, see *assign interfaces* on page 91.

Tunnel

Tunnels are standalone connections that operate without requiring existing physical interfaces, with each tunnel needing a unique source IP address set up in the system.

Find the tunnel settings under **Interfaces > Assign > Tunnels**.

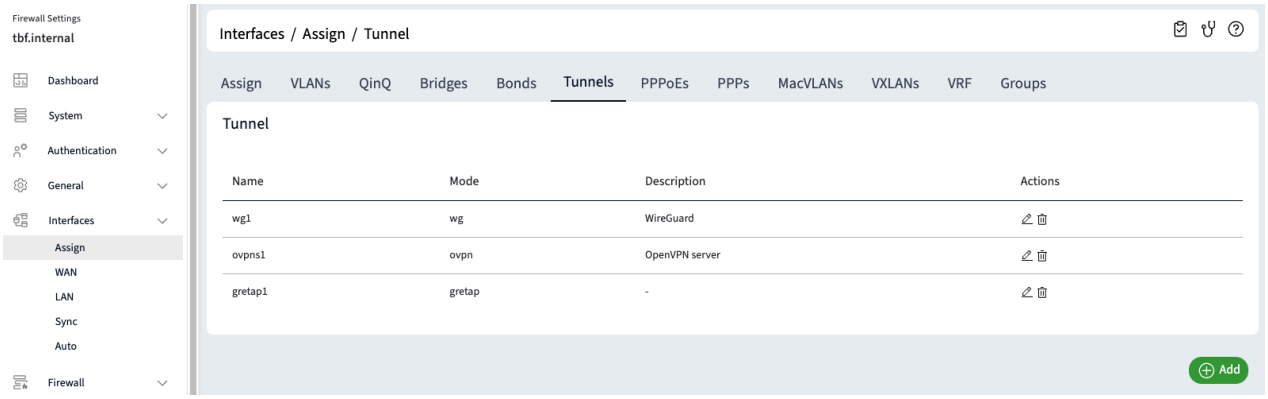


Figure 86: Tunnels

TBF supports these tunnel modes:

- **GRE**
- **GRETAP**
- **IPIP/GIF**
- **SIT**

GRE establishes a layer 3 tunnel for routing traffic between networks. However, **GRETAP** creates a layer 2 tunnel that connects devices as if they were on the same local network segment. This layer 2 capability makes GRETAP useful for bridging remote networks. For more information, see *create new bridge* on page 99.

Both **GRE** and **GRETAP** tunnels can use IPsec encryption. For detailed information, see *GRE over IPsec* on page 335.

For step-by-step instructions, see *create a new tunnel* on page 107. For setup details, see *interface tunnel settings* on page 108. For high availability setups, see *GRE with failover* on page 108.

Create a new tunnel

This task describes how to create a new tunnel interface in TBF.

To establish connections between networks that do not require existing physical interfaces, create tunnel interfaces.

Perform the following steps:

1. Navigate to **Interfaces > Assign > Tunnels**.
2. To create a new tunnel, select **Add**.
3. Set the tunnel settings according to the requirements.
   For detailed configuration options, see the *interface tunnel settings* on page 108.
4. Save the configuration.

TBF creates the new tunnel interface and makes it available for use.

Interface tunnel settings

This topic describes configuration options and parameters available for tunnel interfaces.

Tunnel modes

TBF supports these tunnel modes:

- **GRE**: Generic Routing Encapsulation for layer 3 tunneling
- **GRETAP**: GRE with Transparent Ethernet Bridging for layer 2 tunneling
- **IPIP/GIF**: IP-in-IP tunneling
- **SIT**: Simple Internet Transition tunneling

Settings

Each tunnel requires the following configuration:

- Unique source IP address
- Destination IP address
- Tunnel mode selection
- Optional encryption settings

Advanced settings

Additional configuration options include:

- MTU settings
- TTL configuration
- Key-based authentication
- IPsec integration

GRE with failover

This topic describes high availability (HA) configuration for GRE tunnels using failover mechanisms.

Set up GRE tunnels with failover capabilities to help provide HA and continuous connectivity between networks.

Failover mechanisms

TBF supports several failover approaches for GRE tunnels:

- Primary and backup tunnel configuration
- Load balancing across multiple tunnels
- Automatic failover based on connectivity monitoring

Setup considerations

When you set up GRE failover, consider these factors:

- Network topology and routing requirements
- Monitoring intervals and thresholds
- Failback behavior and timing
- Impact on existing network traffic

Best practices

These guidelines help provide improved GRE failover performance:

- Usage of diverse network paths for primary and backup tunnels
- Setup of appropriate monitoring and alerting
- Testing of failover scenarios on a defined schedule
- Documentation of the failover configuration and procedures

PPPoE

Point-to-Point Protocol over Ethernet (PPPoE) encapsulates Point-to-Point Protocol (PPP) frames within Ethernet frames for broadband internet access.

Find the PPPoE settings under **Interfaces > Set up > PPPoE**.

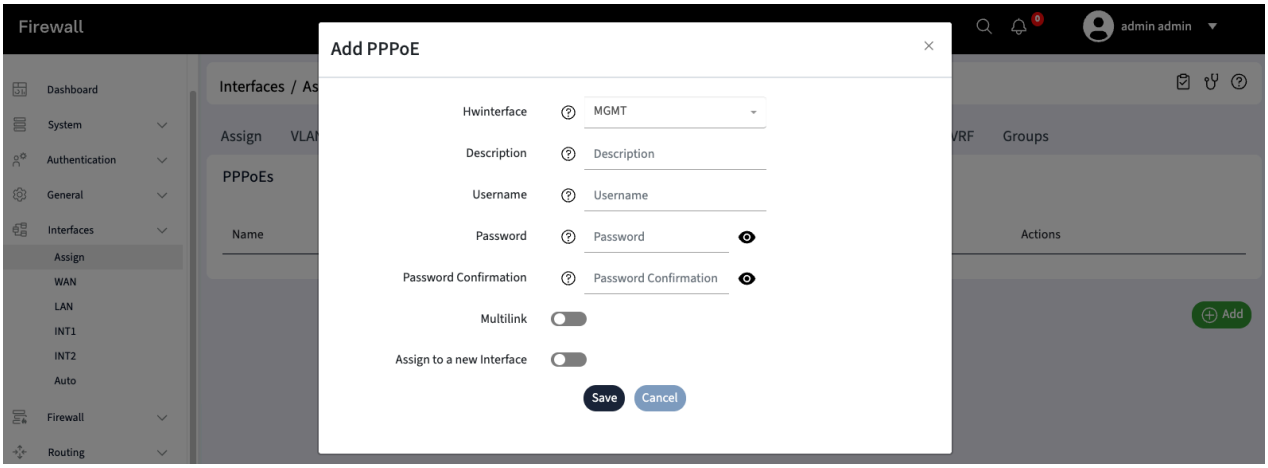


Figure 87: Add PPPoE

PPPoE is a network protocol that wraps PPP frames inside Ethernet frames. Internet service providers (ISPs) use PPPoE to deliver broadband internet access to customers through DSL or cable connections.

PPPoE creates an authenticated connection between the device and the network of the ISP. This protocol allows ISPs to manage user authentication, billing, and bandwidth allocation more effectively than traditional Ethernet connections.

Set up PPPoE in two ways: to connect to the network of the ISP as a client, or to provide PPPoE services to other devices on the network.

Before you use PPPoE, *assign an interface* on page 91.

PPPoE settings

These settings set up and authenticate the PPPoE interface.

Settings

Enable

This setting enables or disables the PPPoE interface.

Description

This field is an optional text field to identify the interface.

Username

This field specifies the authentication username the ISP provides.

Password

This field specifies the authentication password the ISP provides.

Service name

This field specifies an optional service name for the PPPoE connection. If the ISP does not require a service name, then leave the field blank.

Dial on demand

This setting establishes a connection only when traffic requires it, rather than maintaining a persistent connection.

Idle timeout

This field specifies the time in seconds before TBF disconnects an idle connection when **Dial on demand** is enabled.

Set up a DSL connection via PPPoE

This task describes how to establish a DSL internet connection with the PPPoE protocol configuration.

This video demonstrates how to set up a DSL connection with Point-to-Point Protocol over Ethernet (PPPoE). Internet service providers use PPPoE to manage DSL connections and authenticate users.

1. Watch the video tutorial for complete step-by-step instructions.

   View the video tutorial: *DSL PPPoE Setup Tutorial*

PPP

Point-to-Point Protocol (PPP) enables dial-up connections for cellular modems and serial interfaces.

Find the PPP settings under **Interfaces > Assign > PPPs**.

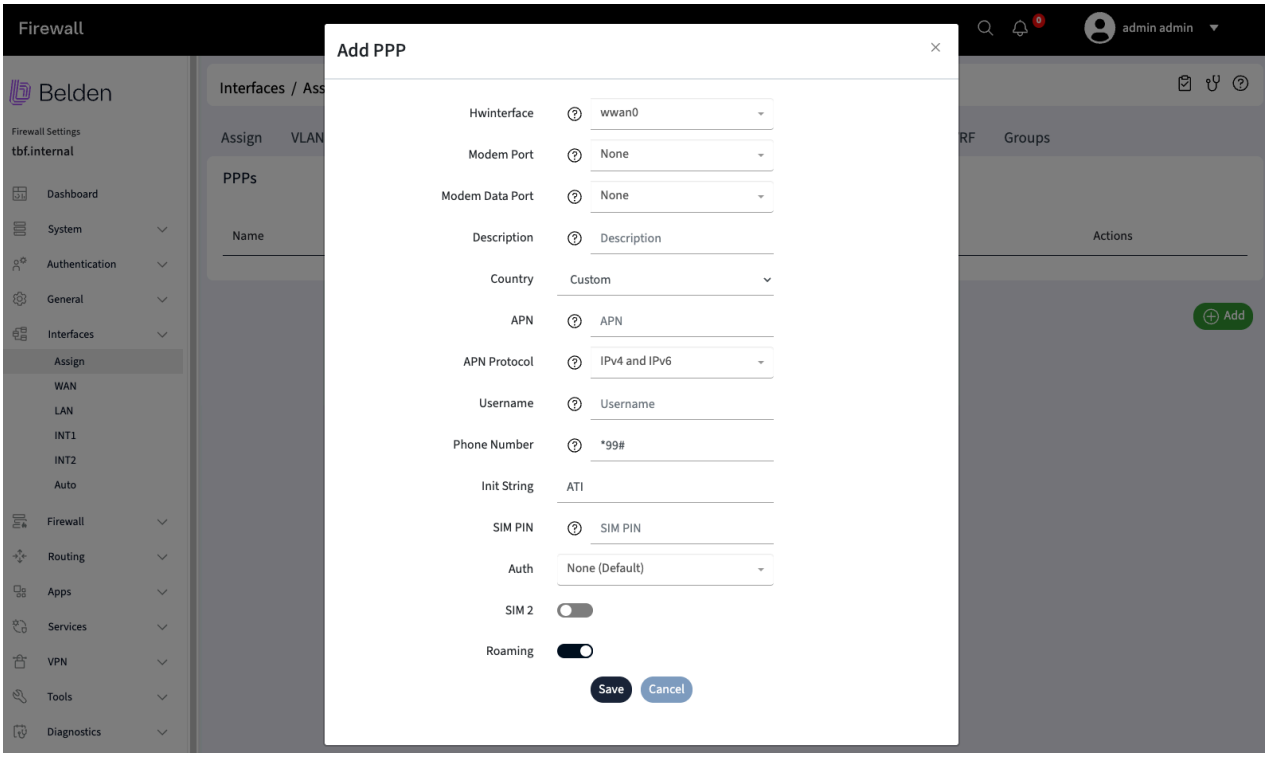


Figure 88: Add PPP

You can set up PPP only on these interface types:

- WWAN physical interfaces
- Modem serial ports

PPP is the standard protocol for establishing cellular modem dial-in connections.

Supported technologies

PPP works with various cellular technologies, including:

- LTE and 5G networks
- LTE450 networks for IoT applications
- Traditional cellular data connections

PPP settings

This topic describes settings for PPP connections including provider settings, authentication, and dual SIM options.

Provider settings

There are three drop-down lists:

- **Country**
- **Provider**
- **Plan**

When you make selections from these lists, TBF automatically fills in several related fields with the appropriate values:

Username

Enter the PPP username that the service provider requires for the connection.

APN

Enter the Access Point Name (APN) that the carrier uses to establish the data connection.

APN Protocol

Select the access protocol that matches the configuration of the provider network. To establish a successful connection, align this setting with the requirements of the carrier.

Auth

Select the authentication method that the provider requires. Select from these options:

- **None (Default)**
- **Chap**
- **PPP**

SIM PIN

If the carrier requires PIN authentication to access the network, then enter the PIN code of the SIM card.

Dual SIM settings

SIM 2

This feature is available on TBF devices that have two SIM card slots. When enabled, it activates dual SIM functionality. Only one SIM card can be active at a time. To switch to the backup SIM when the gateway of the primary connection becomes unavailable, set up automatic failover.

Auto Failover

To automatically switch to the backup SIM card when the primary connection gateway becomes inoperable or unreachable, enable this option.

Roaming

To allow the modem to connect through roaming services outside the home network coverage area, enable this setting.

Note: Dual SIM functionality requires a monitoring IP address and a disabled **Always Up** option for the *gateway* on page 214.

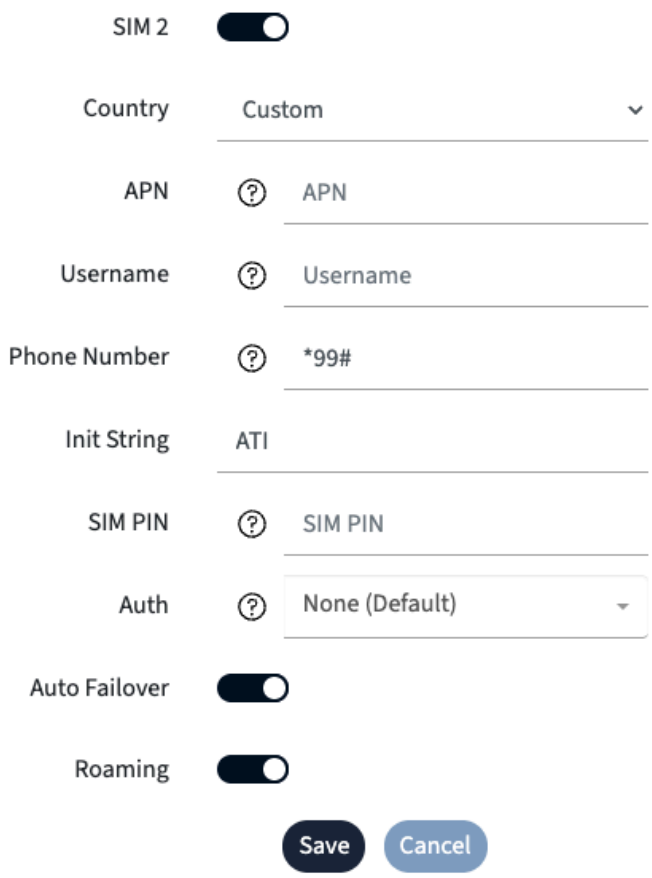


Figure 89: PPP configuration dual SIM

The settings described above are also available for the second SIM card.

Set up an LTE/5G PPP connection

Set up PPP connections for LTE and 5G cellular modems using specific hardware interface settings.

Verify that you have a TBF device with LTE or 5G capabilities.

LTE and 5G modems need specific hardware interface and port settings to create PPP connections. This process establishes a data connection through the cellular service provider.

1. Set the hardware interface to `wwan0`.

   This interface handles wireless wide area network connections.

   Hwinterface `wwan0`
2. Set the modem port to `/dev/cdc-wdm0`.

   This port manages communication with the cellular modem.

   Modem Port `/dev/cdc-wdm0`
3. Set the modem data port to `/dev/cdc-wdm0`.

   This port handles data transmission through the modem.

   Modem Data Port `/dev/cdc-wdm0`
4. Enter the dial-in settings of the provider.

Contact the cellular service provider for the correct Access Point Name (APN), username, and password settings. These vary by carrier and service plan.

5. Assign the PPP connection to an interface.

   This step creates the network interface that applications use for connectivity.

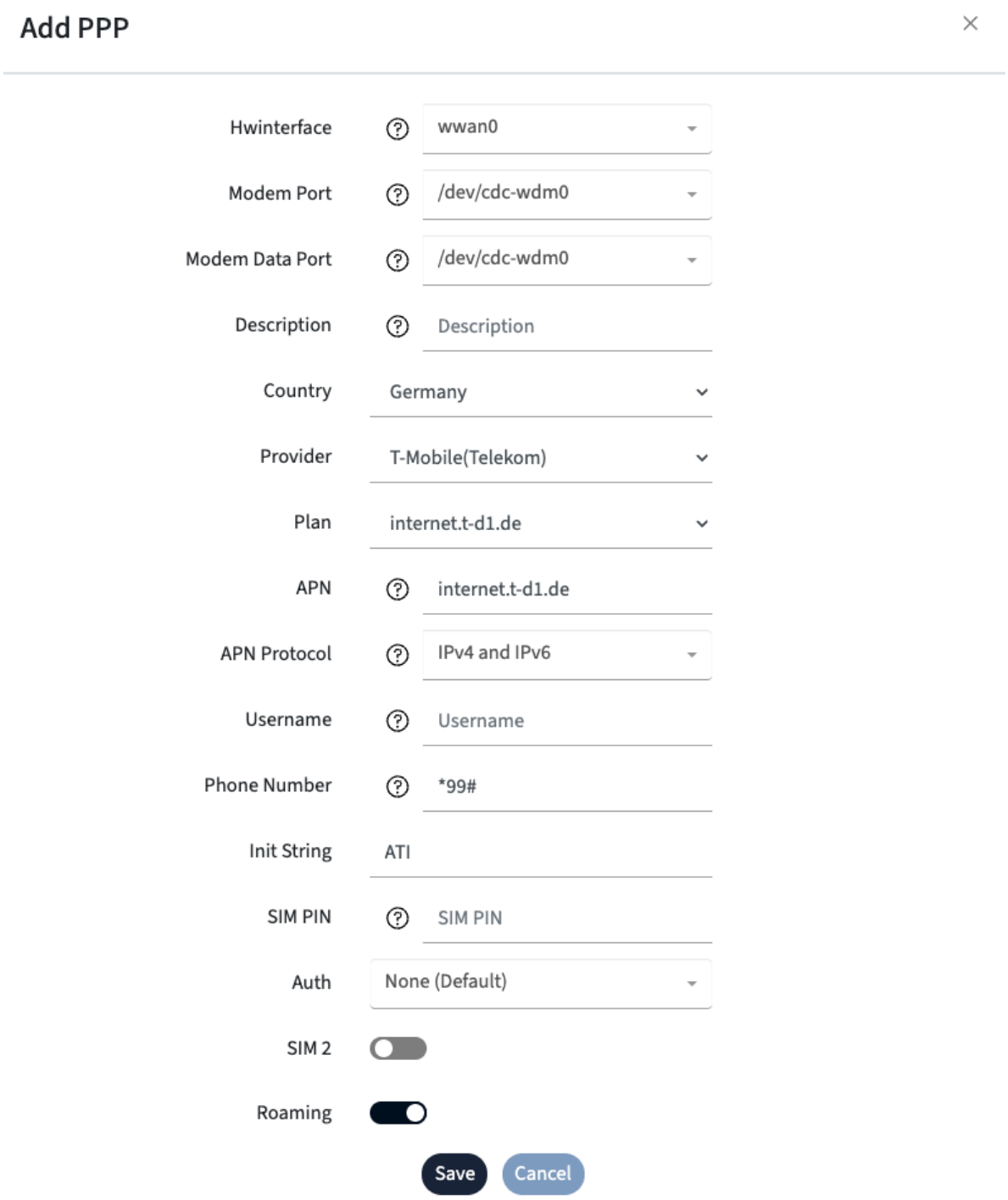


Figure 90: LTE/5G PPP configuration

After completing the setup, use the new interface for internet connectivity or as a backup connection. The connection establishes automatically when needed.

Video tutorial

Watch this video tutorial for a visual guide to LTE/5G configuration: *LTE/5G Setup Tutorial*

LTE450 overview

LTE 450 is a cellular network technology that operates on the 450 MHz frequency band, and delivers strong long-range connectivity for IoT applications.

LTE 450 is a cellular network technology that uses the 450 MHz frequency band to deliver long-range connectivity. LTE 450 offers enhanced coverage and signal penetration through buildings and other obstacles.

As the telecommunications industry moves toward LTE and 5G technologies, the 450 MHz band has adapted to support modern LTE 450 networks. The 450 MHz band works particularly well for IoT devices and applications, which include the following:

- Smart grid systems
- Smart meter services
- Public safety communications

Technical capabilities

LTE 450 supports both CAT-M and narrowband-IoT (NB-IoT) standards, which makes LTE 450 suitable for applications that need wide-area coverage. These standards allow cellular providers to offer comprehensive coverage and keep costs manageable.

Global adoption

Note: Today, 115 operators across 60 countries worldwide use the 450 MHz band.

Set up an LTE450 PPP connection

This task describes how to set up PPP connections for LTE450 cellular modems with specific port settings for 450 MHz band connectivity.

You need a TBF device with LTE450 capabilities.

LTE450 modems need specific port settings to create PPP connections on the 450 MHz frequency band. This frequency band helps provide coverage for rural and remote areas.

1. Set the hardware interface to `None`.

   Select **Hwinterface** `None` from the drop-down list.
2. Set the modem port to `/dev/ttyCELLULAR`.

   In the **Modem Port** field, enter `/dev/ttyCELLULAR`.
3. Set the modem data port to `/dev/ttyCELLULAR`.

   In the **Modem Data Port** field, enter `/dev/ttyCELLULAR`.
4. Enter the dial-in settings of the provider.

   Contact the LTE450 service provider for the correct APN, username, and password settings. These settings vary by provider and region.
5. Assign the PPP connection to a network interface.

   To route the LTE450 traffic, select an available interface from the list.

The LTE450 PPP connection is now active and ready for IoT applications and long-range connectivity in remote locations.

MacVLAN

MacVLAN creates virtual network interfaces that run on top of existing physical interfaces. Each MacVLAN has its own unique MAC address and appears as a separate network device.

A MacVLAN creates a virtual network interface that runs on top of an existing physical interface. Each MacVLAN has its own unique MAC address and appears as a separate network device to other systems on the network. This capability applies when the internet service provider (ISP) assigns IP addresses based on MAC addresses. The creation of multiple MacVLANs on the WAN port obtains additional IP addresses from the ISP.

Find the MacVLAN settings under **Interfaces > Assign > MacVLANs**.

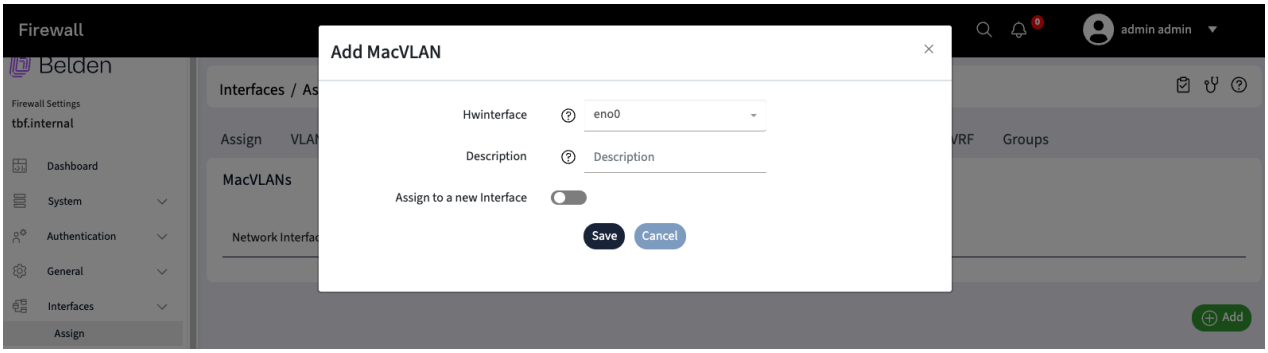


Figure 91: Add MacVLAN

Hwinterface

This setting selects the hardware interface that hosts the MacVLAN.

Description

This field specifies a descriptive name to help identify the purpose of the interface.

Assign to a new Interface

When you create a new MacVLAN, this setting automatically assigns the interface as described in *assign interfaces* on page 91.

SHDSL

Symmetric Digital Subscriber Line (SHDSL) transmits digital data over existing copper telephone wires and offers high-speed connections for business applications. Settings include mode, bitrate, signal quality, and modulation.

Find the SHDSL settings under **Interfaces > Assign > SHDSLs**.

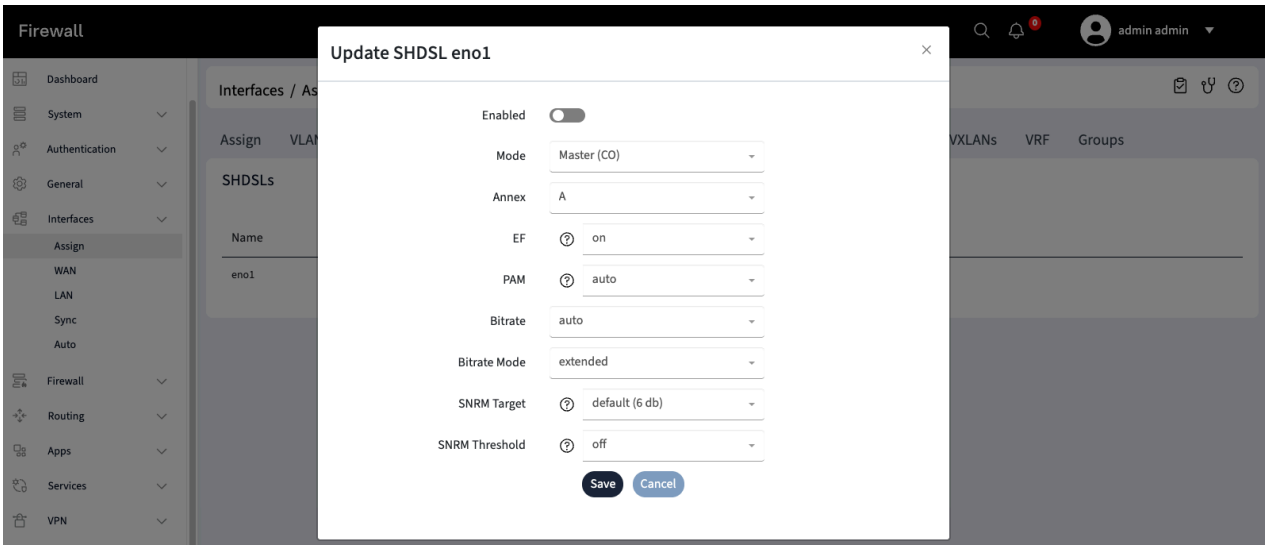


Figure 92: SHDSL settings

SHDSL transmits digital data over existing copper telephone wires. This technology offers high-speed data connections for business applications.

Settings

Enabled

This setting enables or disables the modem.

Tip: Deactivation of the modem when not in use reduces power consumption. An active modem that searches for connections uses significantly more power.

Mode

This setting determines the mode. Select from:

- **Master (CO)**: This option is for central office equipment.
- **Slave (CPE)**: This option is for customer premises equipment.

Annex

This setting determines the specific SHDSL implementation used. Select from:

- **A**
- **B**

EF

The error freeze feature pauses data transmission and maintains the connection when signal quality degrades. This helps prevent complete connection loss on poor-quality lines. The default is **on**.

PAM

The pulse-amplitude modulation setting controls how data signals are encoded. The **auto** default allows the system to select the applicable modulation method.

Bitrate

This setting is the data transmission speed setting. The **auto** default automatically negotiates the available speed based on line conditions.

Bitrate Mode

This setting determines the range of available bitrates. The **extended** default gives you access to higher speed options when line quality permits.

SNRM Target

This setting specifies the signal-to-noise ratio margin (SNRM) target value. This sets the desired signal quality level for operation. The default is **default (6 db)**.

SNRM Threshold

This setting specifies the SNRM threshold for connection maintenance. The **auto** default lets the system determine the appropriate threshold based on line conditions.

Related topics

For more information about SHDSL:

- *SHDSL tuning* on page 117: Information about the optimization of SHDSL connection parameters
- *Bitrate mode* on page 117: Reference for bitrate mode limitations and requirements
- *SHDSL configuration* on page 117: Step-by-step configuration instructions

SHDSL tuning

SHDSL modems automatically detect connection settings, but require manual tuning if detection takes longer than 5 minutes.

Cable quality determines the applicable PAM settings while cable length affects maximum bitrate. Adjust both settings on the master device to optimize performance.

The **Master (CO)** device controls the connection parameters. Make setting changes on the master device only.

Two cable factors affect the SHDSL performance:

- Cable quality determines the applicable PAM settings
- Cable length determines the maximum bitrate settings

Adjust one or both settings based on the specific cable characteristics.

The signal-to-noise ratio margin (SNRM) stays between 6 dB and 9 dB for normal environments, or up to 12 dB for environments with high interference. The SNRM threshold sets an upper limit when DSL line quality fluctuates. However, the system misses brief signal quality changes.

If automatic detection is unsuccessful, then manually adjust speed by changing both PAM and bitrate settings. Think of this as a two-dimensional model where amplitude (PAM) represents the y-axis and speed (bitrate) represents the x-axis of the DSL signal wave.

To increase connection speed, there are two options:

- If the cable quality is good, then use higher amplitude (PAM) settings
- If the cable length is short, then increase the bitrate

Bitrate mode

The **Bitrate Mode** determines maximum supported bitrates and PAM modulation compatibility for different DSL modes.

The **Bitrate Mode** has the following limitations based on the selected mode:

- **g.shdsl.bis** mode works with PAM 16 or PAM 32 modulation. With PAM 16, the maximum supported bitrate is 4 Mbit/s. With PAM 32, the maximum supported bitrate is 6 Mbit/s.
- **g.shdsl** mode only works with PAM 16 modulation and supports a maximum bitrate of 4 Mbit/s.
- **extended** mode has no specific PAM modulation or bitrate restrictions.

SHDSL configuration

This task describes how to set up SHDSL connections with the setup of master and slave devices with appropriate signal-to-noise ratio settings and connection parameters.

To set up an SHDSL connection on the TBF device, complete these steps:

Perform the following steps:

1. Navigate to **Interfaces > Assign > SHDSLs**.
2. Select **Edit**.

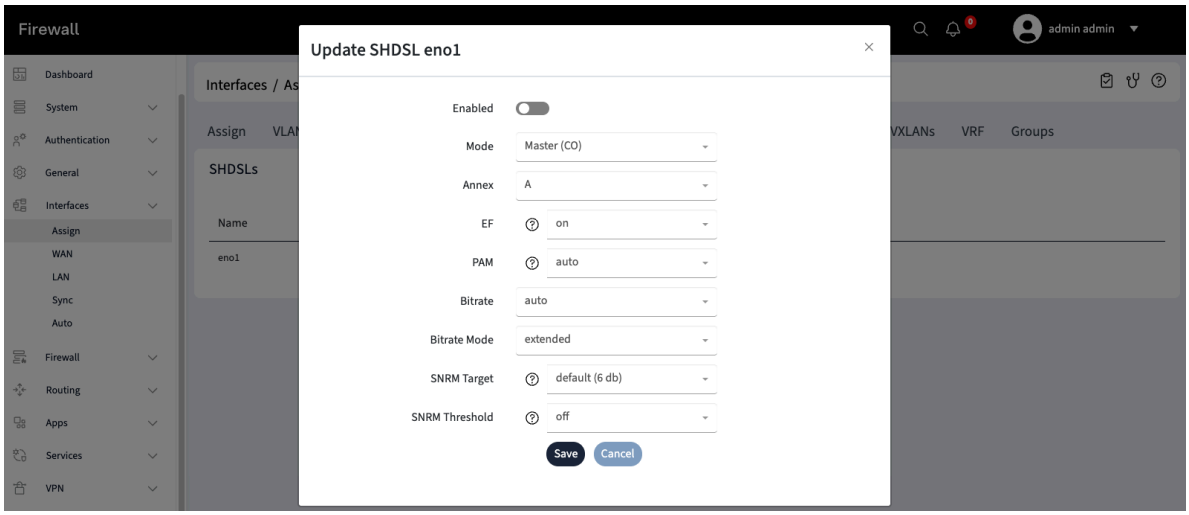


Figure 93: SHDSL setup

3. Set up each SHDSL modem individually.

   The configuration includes one or two SHDSL modems. Because SHDSL connections are point-to-point, you need one device as the **Master (CO)** and one as the **Slave (CPE)**. The master device controls the connection parameters.

4. Begin with the master device setup.

   Use the default settings initially. Optimize the connection as needed. First, set the desired signal-to-noise ratio (SNR).

   - Slower connections: SNR above 9 dB
   - Standard connections: SNR 6 dB
   - High-speed connections: SNR as low as 3 dB

   The auto **Bitrate** feature starts by testing the highest available connection speed and checks if the SNR is adequate. If not, the feature drops to the next lower speed and tries again. This process takes up to 20 minutes; TBF tests each speed for 1 minute. This also applies to active connections—if signal quality decreases, the device reduces the **Bitrate** by one step, and increases the bitrate when quality improves.

5. If auto **Bitrate** cannot establish a connection, set the connection parameters manually.

   Start with the modulation type (**PAM**). Lower numbers indicate simpler and more robust modulation. Check the current SNR under **Diagnostics > SHDSLs**.

Also set **Bitrate** to a manual speed. Lower numbers mean slower connection speeds. Find the right combination of modulation type and connection speed that meets the SNR requirements.

Note: The system only measures SNRM for received data. When one cable lead performs worse than the other, achievable speeds differ depending on the direction. When one direction runs faster than the other and more speed is needed on the slower cable, switching the leads is an option.

6. Set up the emergency freeze (**EF**) as required.

   EF keeps the current connection settings during brief connection drops. Disable this feature if needed. The maximum drop time varies based on the speed and ranges from a few milliseconds to several seconds.

Manual setup of connection parameters helps prevent connection outages because the system does not automatically change connection speeds.

After you establish a connection, navigate to **Interfaces > Assign** and create a new interface based on the SHDSL modem.

VDSL

Very High Speed Digital Subscriber Line (VDSL) enables high-speed data transmission over existing telephone network infrastructure. These VDSL settings set connection parameters that include management IP, multiplex settings, and virtual channel identifiers.

Find the VDSL settings under **Interfaces > Assign > VDSLs**.

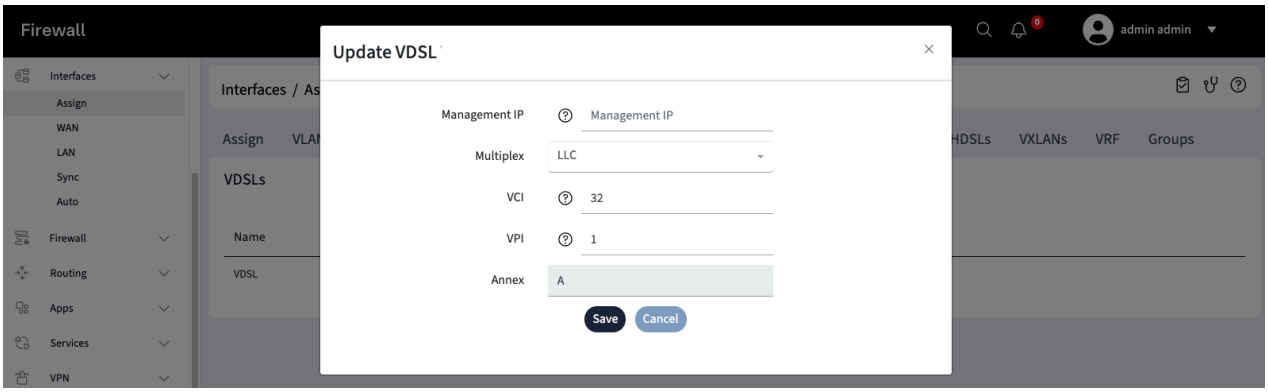


Figure 94: Update VDSL

The following settings control how the VDSL connection operates:

Management IP

This setting sets the IP address that the system uses internally to communicate with the modem and retrieve connection information. Change this address here if it conflicts with other devices on the network. Select an IP address within the `10.0.0.0/8` range. This helps avoid conflicts with public internet addresses.

Multiplex

This setting determines how the system organizes data packets for transmission. You have 2 options:

- **LLC**: (logical link control) offers additional error checking and serves as the default setting.
- **VCMUX**: (virtual circuit multiplexing) offers slightly better performance with less overhead.

VCI

This setting sets the virtual channel identifier, which helps route data through the correct path in the network. The default value of `32` works for most standard configurations.

VPI

This setting sets the virtual path identifier, which defines the main pathway for the connection through the network of the provider. The default value of `1` serves as the standard for most VDSL services.

Annex

This setting shows the annex type of the VDSL connection, which indicates the specific VDSL standard and frequency plan the line uses. TBF automatically detects this value and you do not change it.

Related topics

For more information about VDSL:

- *VDSL configuration* on page 120: Step-by-step instructions for setting up VDSL connections
- *Set up interfaces* on page 92: General interface configuration procedures

VDSL configuration

This task describes how to set up a VDSL connection on the device with the creation of a VLAN with the specific ID of the internet service provider (ISP) and the setup of PPPoE authentication with the credentials provided by the ISP.

To set up a VDSL connection on the TBF device, complete the following steps.

Note: The specific requirements for connecting to a VDSL service vary by ISP. The exact configuration details are available from the ISP. Most ISPs require a VLAN with a specific ID to connect to their VDSL service. This configuration example assumes the ISP requires this setup.

Perform the following steps:

1. Navigate to **Interfaces > Assign > VLANs**.
2. Create a VLAN with the specific VLAN ID of the ISP. For example, when the ISP requires the ID `7`, use this value.
3. Select the underlying interface that matches the modem type. When you have an *internal* modem, select the internal VDSL modem hardware interface. When you use an *external* modem, select the WAN hardware interface.
4. Leave **Assign to a new Interface** disabled.
5. To complete the VLAN setup, select **Save**.
6. Navigate to **Interfaces > Assign > PPPoEs**.
7. To create a new PPPoE interface, select **Add**.
8. Select the VLAN interface you just created as the **Hwinterface**.
9. Enter the username and password credentials that the ISP provided.
10. Enable **Assign to a new Interface**.
11. To complete the PPPoE setup, select **Save**.

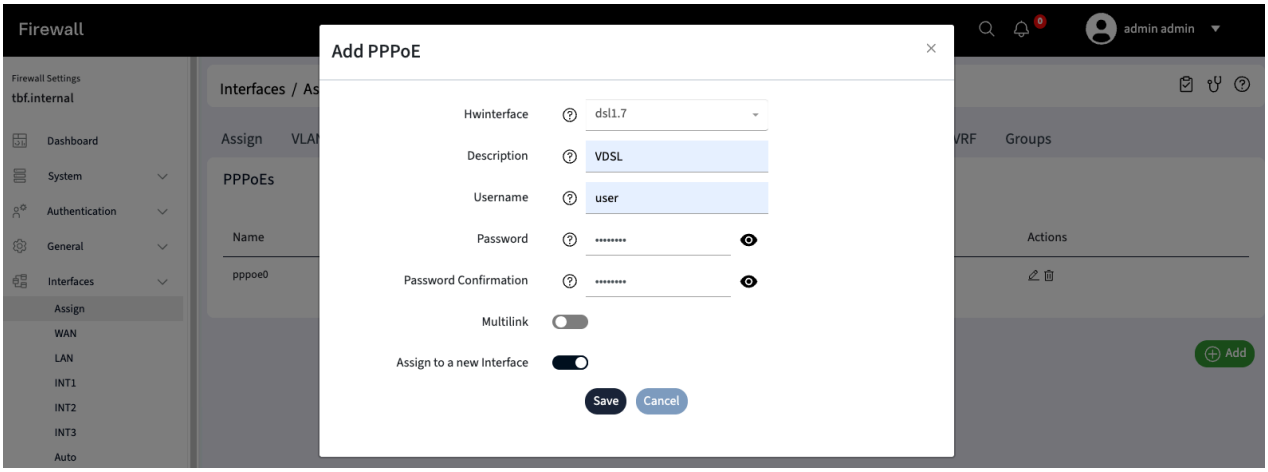


Figure 95: Add PPPoE

TBF creates a new PPPoE interface that you can now set up. For detailed interface setup options, see *set up interfaces* on page 92.

VRF

Virtual routing and forwarding (VRF) allows you to run multiple virtual routers on a single physical router by creating separate virtual router instances (VRIs).

Find the VRF settings under **Interfaces > Assign > VRF**.

VRF allows you to run multiple virtual routers on a single physical router. This technology creates separate VRIs that keep interfaces and routes logically separated from each other.

Each VRF unit works independently, which means different VRF instances cannot communicate directly with each other. This separation lets you use the same IP addresses and routes in different VRFs without creating conflicts.

Note: Not all TBF services support VRF functionality and do not work correctly in VRF environments.

VRF technology enables multiple routing tables to exist on the same router at the same time. Each table maintains its own routing information.



Figure 96: VRF

Each VRF instance has its own unique routing table.

Note: Multiple VRFs cannot share the same routing table, which helps provide separation between virtual routing environments.

VRF technology offers several specialized features:

- *VRF firewall integration* on page 121: How VRF interfaces work with firewall rules
- *VRF management network* on page 122: Out-of-band management capabilities
- *VRF layer 3 VPN capabilities* on page 122: VPN solution support

VRF firewall integration

TBF automatically groups the interfaces assigned to a virtual routing and forwarding (VRF) instance into a single interface group for firewall management.

When you set firewall settings, TBF automatically groups the interfaces that you assign to a VRF instance into a single interface group.

Note: This grouping means you cannot create separate firewall rules for individual interfaces within the same VRF. Instead, the firewall treats the VRF-assigned interfaces as one unified group for security management.

This grouping approach streamlines firewall rule management. Rather than management of multiple individual interface rules, apply security policies to the entire VRF group at once. This application makes the firewall configuration more efficient and easier to maintain.

VRF management network

Virtual routing and forwarding (VRF) enables you to create out-of-band management networks through a dedicated **MGMT** VRF.

VRFs allow you to create an out-of-band management network. The system automatically creates a default **MGMT** VRF for this purpose.

The **MGMT** VRF helps provide a dedicated network path for administrative access that operates separately from the production traffic. This separation helps improve network isolation and helps ensure that management connectivity remains available even when production networks experience disruptions.

VRF layer 3 VPN capabilities

Virtual routing and forwarding (VRF) technology enables layer 3 VPN solutions when it works with dynamic routing protocols like BGP or OSPF.

This feature helps you build secure, isolated network connections between multiple sites. Each VPN instance maintains its own separate routing table. This separate routing table helps provide that traffic from different customers or departments cannot interfere with each other.

VXLAN

Virtual Extensible LAN (VXLAN) is a network virtualization technology that addresses scalability challenges in large cloud computing deployments.

VXLAN is a network virtualization technology that addresses the scalability challenges in large cloud computing deployments. VXLAN uses a VLAN-like encapsulation technique to encapsulate OSI layer 2 Ethernet frames within layer 4 UDP datagrams. The default IANA-assigned destination UDP port number is 4789.

VXLAN endpoints, which terminate VXLAN tunnels and may be either virtual or physical switch ports, are known as VXLAN tunnel endpoints (VTEPs).

Find the VXLAN settings under **Interfaces > Assign > VXLANs**.

Note: VXLAN interfaces operate only on physical interfaces.

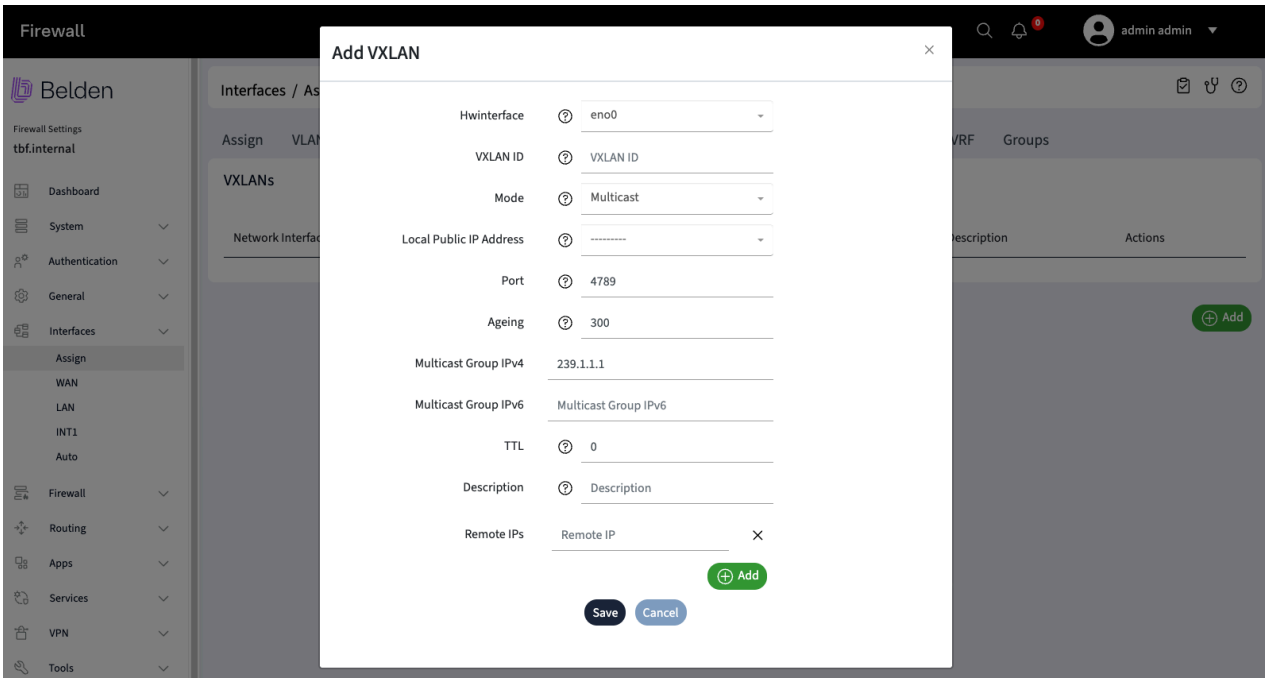


Figure 97: Add VXLAN

VXLAN settings

This reference describes the VXLAN interface settings.

Settings

Name

This field is the name of the VXLAN interface.

Description

This field is a description of the VXLAN interface.

Interface

This setting is the physical interface on which you create the VXLAN.

VNI

This setting specifies the VXLAN network identifier, which is a 24-bit segment ID to identify the VXLAN segment. The valid range is `1-16777215`.

Local IP

This setting specifies the local IP address of the VXLAN tunnel endpoint (VTEP).

Remote IP

This setting specifies the remote IP address of the VTEP.

Port

This setting specifies the UDP port number for VXLAN encapsulation. The default is `4789`.

TTL

This setting specifies the time to live value for VXLAN packets. The valid range is `1-255`.

TOS

This setting specifies the type of service field for VXLAN packets. The valid range is `0-255`.

Learning

This setting enables or disables MAC address learning for the VXLAN interface.

Proxy

This setting enables or disables ARP/ND proxy for the VXLAN interface.

RSC

This setting enables or disables route short circuit for the VXLAN interface.

L2miss

This setting enables or disables L2 miss notifications for the VXLAN interface.

L3miss

This setting enables or disables L3 miss notifications for the VXLAN interface.

UDP checksum

This setting enables or disables UDP checksum calculation for VXLAN packets.

UDP zero checksum6 TX

This setting allows zero UDP checksum for IPv6 transmit connections.

UDP zero checksum6 RX

This setting allows zero UDP checksum for IPv6 receive connections.

Remote checksum TX

This setting enables remote checksum offload for transmit.

Remote checksum RX

This setting enables remote checksum offload for receive.

GBP

This setting enables group-based policy extension for VXLAN.

GPE

This setting enables generic protocol extension for VXLAN.

Usage notes

Note: Bridge configurations require an existing VXLAN interface.

Interface groups

An interface group is a collection of interfaces that lets you apply NAT rules to multiple interfaces simultaneously.

Interface groups help streamline network management by letting you treat multiple network interfaces as a single unit. Instead of setting NAT rules individually for each interface, apply the same rules to all interfaces within a group at once.

Find the interface groups settings under **Interfaces > Assign > Groups**.



Interface groups simplify network configuration by reducing repetitive tasks and helping ensure consistent rule application across multiple interfaces.

10. Firewall

TBF offers stateful packet filtering and network address translation to control traffic flow and help protect the network from unauthorized access.

TBF organizes rules by interface, where the interface refers to the incoming interface where traffic first enters the system. For example, when traffic flows from the LAN to the internet, you need a rule on the LAN interface. The firewall automatically tracks connection states and allows the response traffic back through, eliminating the need for a return rule.

Key firewall capabilities

- **Rule management**: Create and manage firewall rules for different interfaces and traffic types
- **Network address translation**: Set up DNAT, SNAT, and BiNAT for address translation
- **Network objects**: Define reusable aliases for hosts, networks, and services
- **Advanced features**: Set up time control, application control, and DDoS assistance
- **Learning mode**: Automatically generate rules based on observed traffic patterns

Firewall setup areas

The firewall section includes these main setup areas:

- *Firewall general* on page 125: Basic firewall setup and global options
- *Network objects* on page 128: Reusable definitions for hosts, networks, and services
- *Firewall rules* on page 131: Interface-specific forward and input rules
- *Global firewall rules* on page 139: Rules with flexible interface matching
- *DNAT* on page 141, *SNAT* on page 144, and *BiNAT* on page 149: Network address translation
- *Advanced features* on page 152: Additional firewall capabilities

Access firewall settings

Access firewall settings through the web interface under the firewall section. Each area contains specific tools for managing different aspects of network access control and traffic control.

Firewall general

This page describes the general firewall rule settings.

Find the firewall rule settings under **Firewall > Rules**.

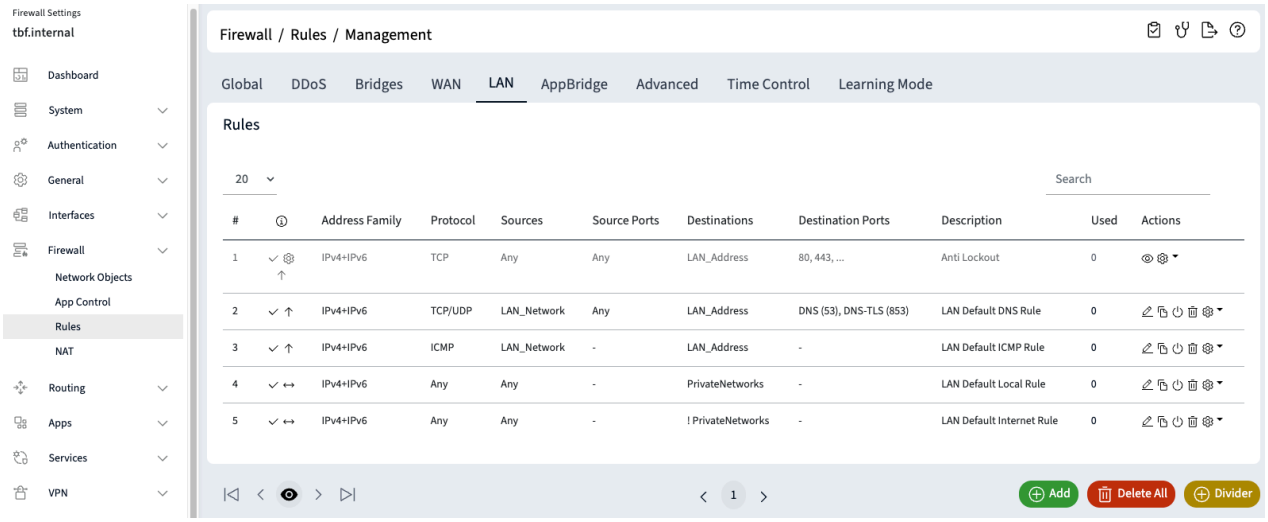


Figure 98: Firewall rules overview

Watch this video tutorial to learn about firewall rules: *Firewall Rules Tutorial*

The following topics explain key firewall concepts that help you understand how the system works:

- *Stateful versus stateless* on page 126: Learn the differences between firewall types and their key features
- *Firewall flow* on page 126: Understand how the system processes firewall rules through different stages
- *Firewall processing* on page 127: Discover how rules are processed in order and their priority levels
- *Flowtable bypass* on page 127: Explore performance optimization techniques for established connections
- *eXpress Data Path* on page 128: Learn about advanced performance acceleration technology

Stateful versus stateless

This page explains the stateful firewall behavior and when stateless rules apply.

What type of firewall is TBF? TBF operates as a *stateful* firewall. This means TBF monitors and remembers active network connections. Established connections continue without re-evaluating firewall rules for each packet.

Stateful firewall

A stateful firewall is a network security device that monitors the state and characteristics of network connections that pass through the firewall. The firewall maintains a record of active connections and distinguishes between legitimate packets for different connection types. Only packets that belong to known, established connections pass through the firewall. This approach helps improve packet filtering by understanding the context of network traffic. (*stateful firewall*)

Stateless firewall

Stateless firewalls examine network traffic and allow or block packets based on static criteria such as source and destination addresses. They do not track connection states or understand traffic patterns. A stateless firewall applies simple rule sets without considering whether a packet is part of an established connection. This makes them vulnerable to certain attacks where malicious packets appear to be legitimate responses to requests you made. (*stateful vs. stateless firewalls*)

Firewall flow

This page describes how the system processes network traffic through the firewall.

Understanding how the system processes firewall rules helps you create more effective security policies. The following diagram shows the path network traffic takes through the firewall system.

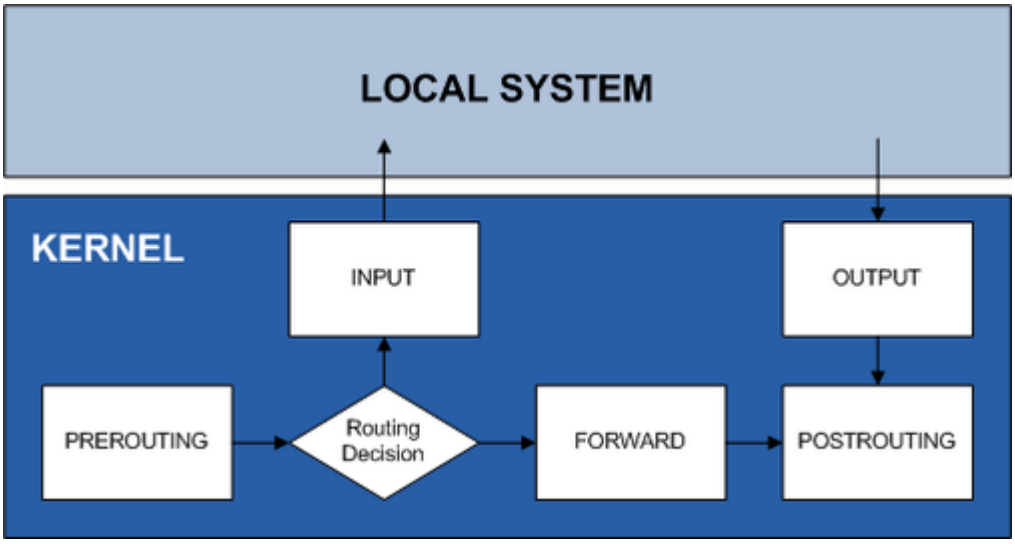


Figure 99: Linux network flow

The system performs specific operations at each stage of the traffic flow:

- `PREROUTING`: Applies Destination Network Address Translation (DNAT) to incoming packets
- `FORWARD`: Applies firewall rules to traffic that passes between different network interfaces
- `INPUT`: Applies firewall rules when traffic targets the firewall device itself, such as DNS server requests or management interface access
- `OUTPUT`: Applies firewall rules when the firewall device generates traffic, such as DNS server responses or management interface communications
- `POSTROUTING`: Applies Source Network Address Translation (SNAT) to outgoing packets

This chapter examines each stage in detail.

Note: TBF does not filter `OUTPUT` traffic. The firewall device can send outbound traffic without restrictions.

Note: In `INPUT` rules that target the firewall device, the destination is the network object **Address** or **Network** that is assigned to each interface. Entering the interface IP address as the destination is insufficient. The system treats this as a forward rule rather than an input rule.

Firewall processing

This page describes how the system evaluates global and interface-specific firewall rules.

In TBF, the system evaluates global firewall rules first, then moves to interface-specific rules. When the system finds the first rule that matches the traffic, the system applies that rule and stops checking additional rules. This means the sequence of firewall rules is required for proper operation.

Flowtable bypass

This page describes the flowtable bypass feature that improves firewall performance for established connections.

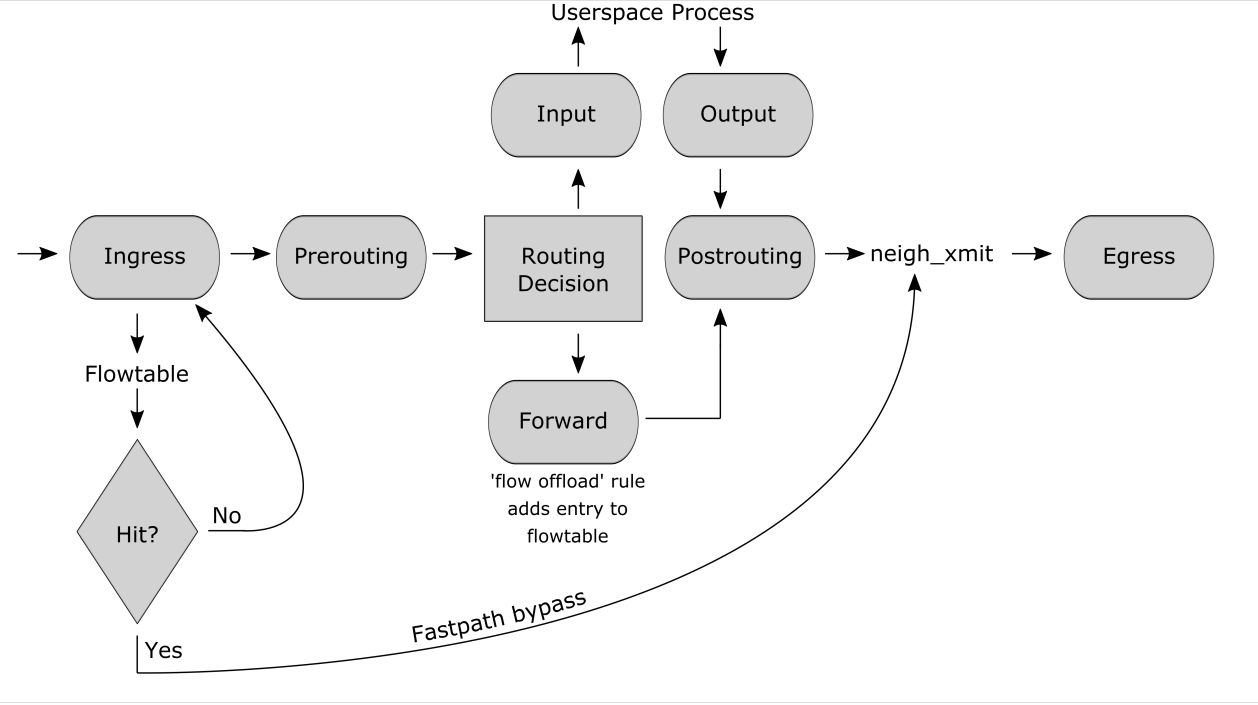


Figure 100: Netfilter hooks and flowtable interactions

The flowtable bypass feature improves firewall performance by doubling processing speed. This feature skips the standard network stack for established connections and routes traffic directly from the input interface to the output interface.

Note: Flowtable bypass works only with stateful connections that use TCP or UDP protocols.

The feature activates after the system establishes the initial packet flow. For long-running connections, flow offload stays active only while traffic flows through the connection. When no traffic passes through for several minutes, the connection returns to normal processing mode. The system reactivates flow offload when the system detects new traffic on the connection.

The system automatically disables flowtable bypass for firewall rules that include logging or rate-limiting features, as these functions cannot operate with the bypass mechanism.

eXpress Data Path

This page describes XDP offload technology for accelerating firewall rule processing.

The XDP offload technology significantly improves network performance by accelerating data processing. XDP increases flowtable bypass speed by 2.5 times and normal firewall speed by 5 times. This enhancement uses XDP and eBPF technology to allow established TCP or UDP connections to forward data directly from the network driver. The connections skip traditional processing layers.

Activating XDP disables QoS functionality. The performance improvements require bypassing QoS logic in the TBF firewall. This trade-off is necessary to achieve this speed increase.

If you need improved firewall performance and can operate without QoS features, then enable XDP.

For implementation details, see *XDP accelerator* on page 20.

Network objects

This page describes reusable network objects that define network elements for firewall rules.

Network objects are reusable components that define network elements for firewall rules and other security policies. Find the network objects settings under **Firewall > Network Objects**.

TBF supports 5 types of network objects:

- *Hosts (single IP addresses or hostname)* on page 129
- *Hosts (network range)* on page 129
- *Ports* on page 130
- *MAC addresses* on page 130
- *DNS domains* on page 131

Note: If you create nested network objects, then verify that all objects share the same type. For instance, you cannot add a MAC address object to a port object because the two object types differ. Additionally, avoid creating circular references where two network objects reference each other, as this creates an invalid loop structure.

For host objects, specify either IP addresses or hostnames (DNS entries). If you use hostnames, then verify that the service that uses the network object resolves the hostname to an IP address.

After you save network object entries, reorder the entries with the drag-and-drop functionality.

Note: This order is purely visual and does not affect the operational behavior of the network object.

Network objects offer flexibility because they allow reuse across multiple locations, such as firewall rules and other security configurations.

TBF automatically creates network objects for interface IP addresses, network ranges, and virtual IP addresses to streamline configuration.

Hosts (single IP addresses)

This page describes how to create single IP address objects for firewall rules.

Specify individual IP addresses or network objects that contain single IP addresses. TBF supports both IPv4 and IPv6 address formats.

Instead of IP addresses, use hostnames, which are DNS entries that resolve to specific IP addresses.

The following table lists the built-in hosts and their corresponding IP addresses that are automatically recognized by the system.

Host name	IP addresses
OSPF	The OSPF protocol uses the following multicast addresses: • `224.0.0.5` - All OSPF routers (IPv4) • `FF02::5` - All OSPF routers (IPv6) • `224.0.0.6` - OSPF designated routers (IPv4) • `FF02::6` - OSPF designated routers (IPv6)

Hosts (network range)

This page describes how to create network range objects for firewall rules.

Specify network ranges or other network objects that contain network ranges. These ranges support both IPv4 and IPv6 formats.

To find out how to set up dynamic block lists, see *dynamic block lists* on page 129.

The following table lists the built-in hosts and their corresponding IP addresses that are automatically recognized by the system.

Name	Network range
LOOPBACK	`127.0.0.0/8`
RFC1918_A	`10.0.0.0/8`
RFC1918_B	`172.16.0.0/12`
RFC1918_C	`192.168.0.0/16`
MULTICAST	`224.0.0.0/4`
RFC4193	`fd00::/7`
PrivateNetworks	• RFC1918_A • RFC1918_B • RFC1918_C • RFC4193

Dynamic block lists

This page describes dynamic block lists that automatically retrieve data from external URLs.

Dynamic block lists automatically retrieve current data from a specified URL to update network object entries. This feature helps maintain up-to-date security policies without manual intervention.

Perform the following steps:

1. Enable the **DBL** feature.
2. Enter the source **URL** that contains the updated network object data.
3. Set the **Update Interval** to either daily or hourly. This depends on how frequently you need the data refreshed.

The system automatically runs a background job at your specified interval and updates the network object entries with the current data from the URL.

Ports

This page describes how to create port objects and port groups for firewall rules.

Enter ports or other network objects that contain ports. You can also add built-in ports.

The following table lists the built-in ports.

Name	Port
BGP	179
CIFS	3020
CITRIX-ICA	1494
DNS	53
DNS-TLS	853
ESP	4500
FTP	21
FTP-DATA	20
HTTP	80
HTTPS	443
IEC104	2404
IMAP	143
IMAPSSL	993
ISAKMP	500
KERBEROS	88
LDAP	389
LDAPS	636
LPD	515
MODBUS	502
NETBIOS-SSN	139
NFS	2049
OPENVPN	1194
POP3	110
POP3S	995
PPTP	1723
RADIUS	1812
RADIUS-ACCT	1813
RSH	514
RTSP	554
SIP	5060
SIP-TLS	5061
SMTP	25
SMTPTLS	465
SMTPSSL	587
SNMP	161
SNMPTRAP	162
SQLNET	1522
SSH	22
TELNET	23
UUCP	540
WHOIS	43

MAC addresses

This page describes how to create MAC address network objects.

Create MAC address network objects. TBF uses MAC address network objects only for source MAC addresses in firewall rules. For MAC addresses, you can also use dynamic block lists.

CAUTION: Use MAC address network objects with care because layer 3 routers change the source MAC address.

DNS domains

This page describes how to create DNS domain network objects.

Create DNS domain network objects. The system supports 3 different types:

- Direct subdomains (`*.test.de`)
- All subdomains (`**.test.de`)
- Exact match (`www.test.de`)

You cannot query all subdomains because DNS servers do not allow zone transfers. Therefore, DNS servers do not allow you to crawl the subdomains.

TBF automatically learns the subdomains when it observes the responses in the built-in DNS server.

Direct subdomains only observe first-level subdomains. For example, the system observes `www.test.de` but not `test.www.test.de`.

TBF limits results to 256 entries. After TBF reaches the limit, it removes the oldest entry and adds a newer one.

Important: The clients behind the firewall use the *DNS* on page 263 server of the firewall and not a third-party DNS server. For the same reason, use only DNS domain network objects as **Destination** in global and interface firewall rules.

The DNS domain dynamically adds wildcard domains to firewall rules when it observes DNS results.

Firewall rules (forward and input)

Firewall rules form the core of TBF and organize traffic control by interface with built-in and user-created rules.

When you access the rules, TBF displays firewall rules for each interface. An interface represents the entry point where network traffic first enters the firewall.

This is how traffic flow works. When a computer connects through the LAN and the internet connects through the WAN, website traffic enters the firewall through the LAN interface. The LAN needs a firewall rule to permit this traffic. A return rule is unnecessary because the firewall maintains a connection state that tracks active sessions. The system automatically permits return traffic from WAN to LAN.

TBF also has *global firewall rules (forward and input)* on page 139. Global rules offer greater flexibility because you specify both incoming and outgoing interfaces or set the interfaces to accept any interface.

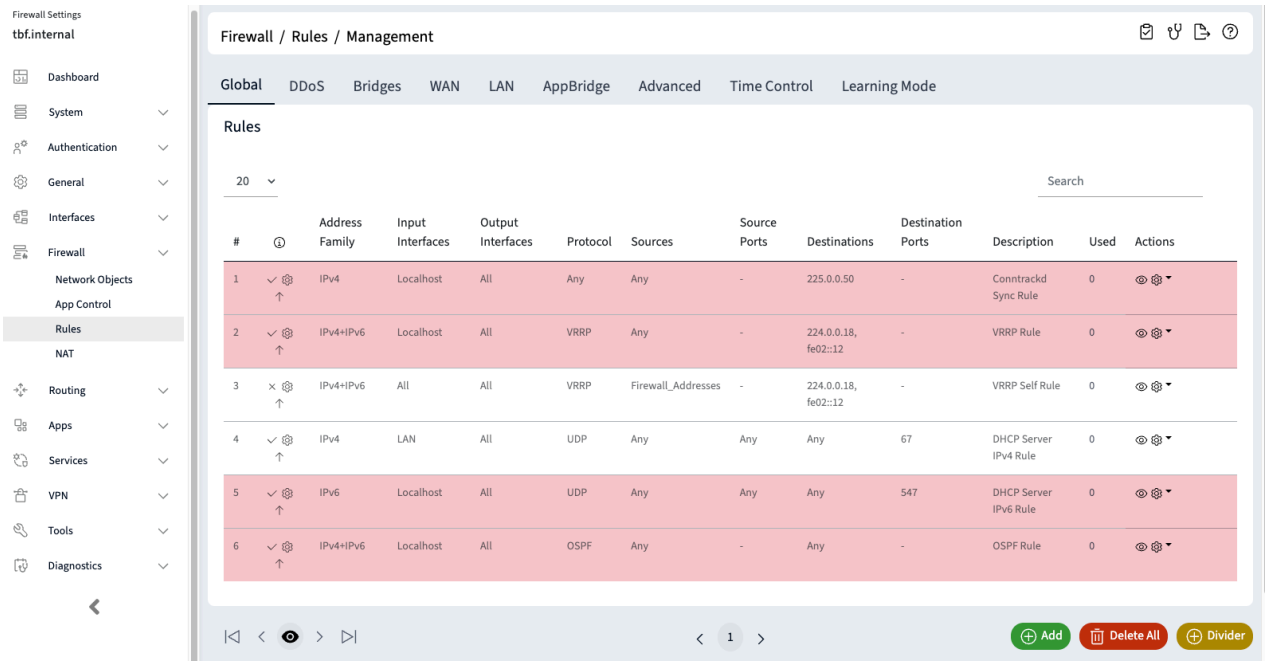


Figure 101: Firewall rules overview

Each firewall rule page displays built-in rules with gray backgrounds. The system manages these rules. You cannot modify or reposition the rules. Edit only user-created rules.

To export the settings as an Excel spreadsheet, use the export option in the top-right corner.

Rule management tasks

- *Create and update firewall rules* on page 132: Add new rules with various configuration options
- *Move firewall rules* on page 133: Reorder rules within interfaces or between pages
- *Delete firewall rules* on page 133: Remove individual rules or all rules from an interface
- *Search firewall rules* on page 134: Find specific rules with various criteria
- *Convert firewall rules* on page 134: Change between global and interface rules
- *Create and manage firewall rule dividers* on page 167: Organize rules with visual dividers to manage them better

Rule configuration and monitoring

- *Firewall rule settings* on page 134: Configuration options and parameters
- *Firewall rule usage and changes* on page 137: Usage statistics and modification history
- *Hostnames in firewall rules* on page 138: Use DNS names to set up rules

Diagnostic tools

- *Use TCPDump with firewall rules* on page 138: Monitor network traffic for specific rules
- *Trace firewall rules* on page 138: Monitor how packets match in real time

Create and update firewall rules

Create new firewall rules on an interface with various configuration options.

When you select **Add**, TBF creates a new rule for the current interface. Customize this rule with several configuration options. TBF organizes the rule settings into the following categories:

- *General settings* on page 134
- *ICMP settings* on page 135
- *Sources* on page 135
- *Destinations* on page 135
- *Advanced* on page 135

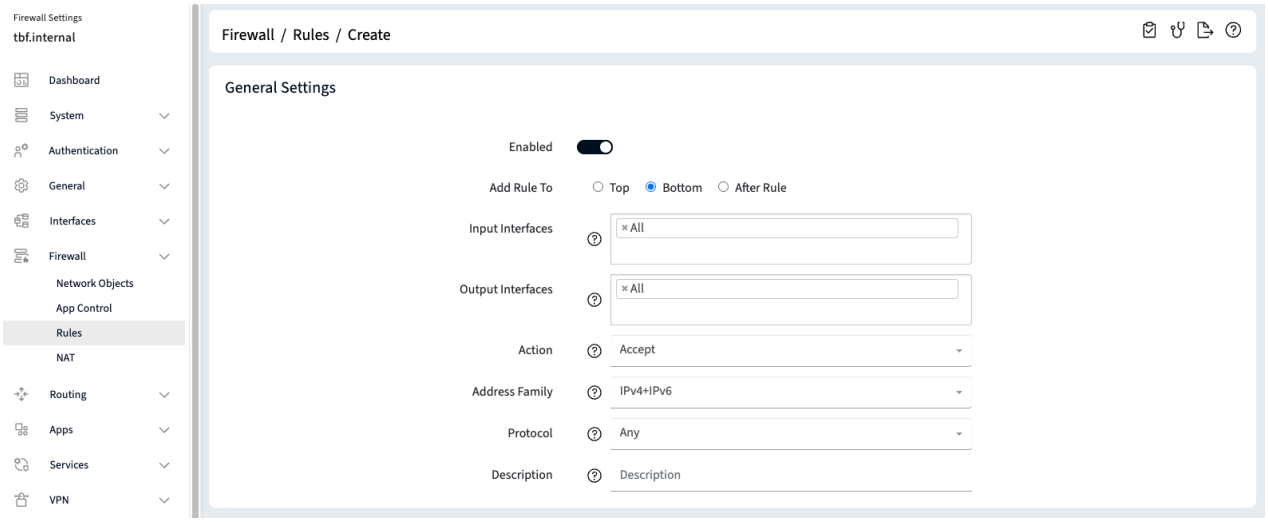


Figure 102: Create a firewall rule

Perform the following steps:

1. To create a new rule, select **Add**.
2. Set the rule settings according to network security requirements.
   For detailed information about each setting, see *firewall rule settings* on page 134.
3. To apply the changes, save the rule.

Move firewall rules

Move user-created firewall rules to different positions within the interface or between pages.

TBF organizes firewall rules by interface and displays the rules in groups of 20 rules per page. You can reposition user-created rules when you drag and drop the rules to a new location. After you move a rule, save the new position of the rule when you select **Save**.

Perform the following steps:

1. Select the firewall rule when you select the first cell in the row.
2. Use the navigation arrows at the bottom left to move the rule between pages.
   The arrows allow you to move the rule to the next page, previous page, first page, or last page. Hover over each button to see the function of the button.
3. Select **Save** to confirm and save the new position of the rule.

Delete firewall rules

Remove firewall rules from a specific interface or tab. You can delete the rules at once or remove individual rules as needed.

You can remove firewall rules from any interface or tab through the GUI. Deleting rules is permanent and you cannot undo this action.

Perform the following steps:

1. To remove the firewall rules from the current interface, select **Delete All** in the lower-right corner.
2. To remove a specific rule, select the firewall rule to delete. Then select **Delete All**.
   When you select a specific firewall rule before you select the delete button, TBF removes only that rule instead of the rules on the interface.

WARNING: Removing firewall rules does not terminate existing active connections that the system established under those rules. To close active connections, refer to *states diagnostics* on page 424.

Search firewall rules

Search for specific firewall rules on the overview page.

In the upper-right corner of the overview page, search for rules. As search values, use:

- Protocol
- Source
- Destination
- IP address
- Port
- Description

Perform the following steps:

1. Navigate to the firewall rules overview page.
2. Locate the search field in the upper-right corner.
3. Enter the search criteria.

Convert firewall rules

Convert between global firewall rules and interface firewall rules with the **Actions** column.

In the **Actions** column on the right side of the overview page, convert firewall rules. Convert a global firewall rule to an interface firewall rule and vice versa.

Perform the following steps:

1. Navigate to the firewall rules overview page.
2. Locate the rule you want to convert in the **Actions** column.
3. Select the convert option for the rule.
4. When you convert a global firewall rule, select an interface from the pop-up window.

Firewall rule settings

This topic describes configuration options and parameters for firewall rules, organized by category.

General settings

Enabled

This setting enables or disables the firewall rule.

Add Rule To

This setting determines where the firewall rule appears in the **Rules** list on the overview page.

Interfaces

This setting changes the interface assignment for the firewall rule. When you modify this setting, TBF places the rule at the end of the rule list for that interface.

Action

This setting determines the firewall rule action. Select from:

- **Accept**
- **Drop**: This option silently discards the packet.
- **Reject**: This option sends back a reject ICMP message.
- **Match**

Address Family

This setting specifies the IP address family. When you select IPv4+IPv6, TBF might not generate rules for both protocols, depending on the sources and destinations you define.

Protocol

This setting specifies the layer 2 protocol for the firewall rule.

Description

This field adds a descriptive label for the firewall rule.

ICMP settings

If you select ICMP as the protocol, then filter traffic by ICMP type.

Sources

The source settings include options for **Host** and, when applicable, **Source Ports**. Use hostnames (DNS entries) instead of source IP addresses. Add multiple entries and mix IPv4 and IPv6 addresses. TBF automatically determines the appropriate rule configuration.

Invert IP Match

This setting reverses the matching logic for IP addresses, MAC addresses, and ports.

Destinations

The destination settings include options for **Destination** and, when applicable, **Destination Ports**. Use hostnames (DNS entries) instead of destination IP addresses. Add multiple entries and mix IPv4 and IPv6 addresses. TBF automatically determines the appropriate rule configuration.

Note: To create INPUT rules that target the firewall itself, use the network objects *Address* or *Network* of each interface. It is insufficient to enter the interface IP address as the destination. The system creates a forward rule instead of an input rule.

Invert IP Match

This setting reverses the matching logic for IP addresses and ports.

Because of the structure described in *firewall flow* on page 126, if you target the firewall itself as the destination, then explicitly select the interface IP network object. TBF does not recognize custom IP addresses or network objects for this purpose.

Advanced

The advanced settings provide the following configuration options:

Log this Rule

This setting records traffic that matches this rule and adds a prefix for easier identification in logs.

WARNING: Firewall logging consumes significant system resources and generates numerous log entries.

Advanced Settings

This setting displays additional configuration options when enabled.

Force Input Rule

This setting forces this rule to become an input rule and makes the firewall the destination regardless of the IP address that is specified in the destination field. Normally, TBF automatically determines whether to place the rule in input, forward, or both chains. This setting overrides that detection and places the rule only in the input chain.

Force Output Rule

This setting forces this rule to become an output rule and makes the firewall the source regardless of the IP address that is specified in the destination field. Normally, TBF automatically determines whether to place the rule in output, forward, or both chains. This setting overrides that detection and places the rule only in the output chain.

Stateless

This setting helps prevent connection state tracking for this rule. This requires you to create a separate rule for return traffic. Use this setting primarily for asymmetric routing scenarios.

Trace

This setting displays packets that match this rule in real time. The trace function slows rule processing but helps with troubleshooting. Disable the trace function after debugging. View packets that match in *diagnostics firewall trace* on page 398.

Limit

This setting restricts rule activity when you set a maximum number of matches within a time period that is specified. Set limits by speed (KBit, Mbit, Byte, KByte, or MByte) or time-based packets per second, minute, or hour. See also *QoS* on page 296.

TCP Flags

When you use TCP protocol, this option filters traffic by specific TCP flags.

TCP MSS

When you use TCP protocol, this option sets the maximum segment size (MSS). This is necessary when the MTU is smaller than standard. This setting is also available per interface in the interface configuration.

Routing Table Load Balancing

This setting enables load balancing between two routing tables, useful for scenarios like internet load balancing.

Routing Table

This setting uses an alternative routing table for traffic that matches instead of the default main routing table. This enables *policy routing*, where traffic that matches follows the selected routing table.

Routing Table Reverse Main

This setting uses the main routing table for the reverse direction of the connection. This is necessary for policy routing because the local LAN route exists in the main routing table. In this context, the reverse direction refers to WAN-to-LAN traffic.

Bypass IDS/App Control

When enabled, this setting helps prevent the intrusion detection system or application detection engine from inspecting traffic that matches.

Limit

This setting sets the maximum match rate for this rule. TBF drops packets that exceed this rate.

Note: When you use **MByte/s**, providers measure speed in MBit (8 MBit = 1 MByte).

Note: Rate limits apply to forward and input rules.

Limit per Host

When enabled, this setting applies the limit to each individual host rather than all hosts that match this rule collectively.

Limit By

This setting specifies whether to limit by download or upload traffic.

QoS Class Input

This setting determines the class that is used for input traffic shaping on any interface with QoS that is enabled that this traffic passes through. Verify that you assign directions correctly. If you create a rule on LAN for shaping on WAN, then verify that QoS is active on WAN and select the appropriate class for WAN. Unlike QoS output, only set up QoS input for non-bridge member interfaces.

QoS Class Output

This setting determines the class that is used for output traffic shaping on any interface with QoS that is enabled that this traffic passes through. If you create a rule on LAN for shaping on WAN, then verify that QoS is active on WAN and select the appropriate class for WAN.

Enforcer

This setting assigns an operational technology (OT) protocol enforcer to this rule. Each enforcer restricts which values the rule accepts in the **Source**, **Destination**, and **Destination port** fields. For more information, see *Enforcer rule requirements* on page 170.

DSCP Type

This setting sets the differentiated services code point (DSCP) for QoS in the IP or IPv6 header. Match different DSCP types with this option. TBF only evaluates this option when it creates a new firewall state on the first packet of a connection. TBF does not recognize subsequent DSCP type changes for an existing firewall state.

Time Control

This setting activates the firewall rule only during time periods that are specified. First create time objects in the **Advanced Settings**. For more information, see *Firewall time control* on page 157.

No XDP

This setting disables eXpress Data Path (XDP) offload for this connection. This is useful when you require QoS functionality.

Raw Data

This setting allows you to append raw data to the firewall rule that is generated. Incorrect input prevents TBF from loading firewall rules. Follow `nftables` syntax for proper formatting.

Enable Expiry Date

This setting enables the expiration date feature for the firewall rule.

Expiry Date

This field specifies the expiration date. Enter a date in the format `dd.mm.yyyy`.

Firewall rule usage and changes

View usage statistics and modification history for firewall rules.

Rule usage statistics

Each firewall rule page displays usage statistics that show how frequently TBF applies a specific rule. Hover over the usage number to view detailed information including:

- Total number of connection states
- Currently active states
- Number of packets the rule processes
- Total data size in bytes

This information helps you understand which rules actively help protect the network and their impact on traffic flow.

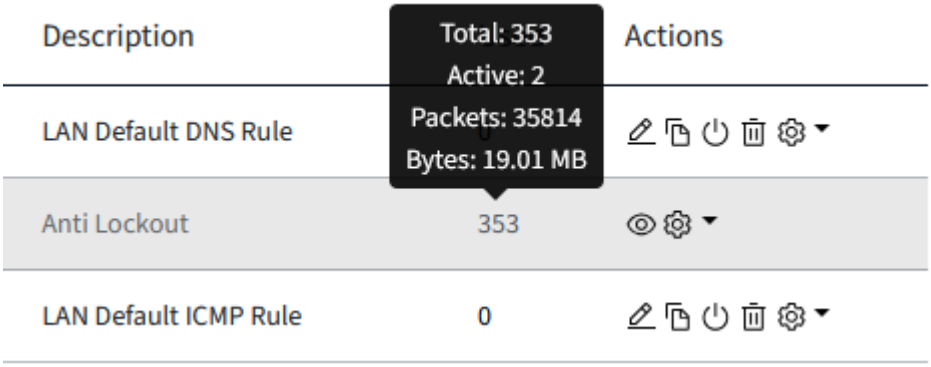


Figure 103: Firewall rules used

Rule modification history

At the bottom of each rule, track its change history through 3 key pieces of information:

- **Created date**: When someone first established the rule
- **Modified date**: When someone last updated the rule
- **Modified user**: Which user made the most recent changes

This audit trail helps you track configuration changes and troubleshoot rule-related issues.

Hostnames in firewall rules

Use hostnames as destinations or sources in firewall rules with DNS resolution requirements.

TBF supports hostnames in rules as destination or source. Confirm that the system can resolve them when the system loads, applies, or reapplies the firewall rules. The system cannot update the firewall rules when there is no working DNS. The reload is unsuccessful and leaves the former ruleset in place.

Use TCPDump with firewall rules

Start a TCPDump for a firewall rule to monitor network traffic in real time.

In the **Actions** column on the right side of the overview page, start a TCPDump for a firewall rule.

Perform the following steps:

1. Navigate to the firewall rules overview page.
2. Locate the rule you want to monitor in the **Actions** column.
3. Select the TCPDump option for the rule.

A pop-up window appears and shows the result in real time.

Trace firewall rules

Start a trace for a firewall rule to monitor packet matching in real time.

In the **Actions** column on the right side of the overview page, start a trace for a firewall rule.

Perform the following steps:

1. Navigate to the firewall rules overview page.
2. Locate the rule you want to trace in the **Actions** column.
3. Select the trace option for the rule.

A popup window appears and shows the result in real time. The window only shows data when a packet matches for the first time. Otherwise, the window remains empty.

Global firewall rules (forward and input)

Global firewall rules offer flexible traffic control by letting you specify both input and output interfaces without tying the rule to a specific interface.

Firewall rules serve as the core foundation of TBF. When you navigate to the rules section, the system displays firewall rules organized by interface.

Find the global firewall rules under **Global**.

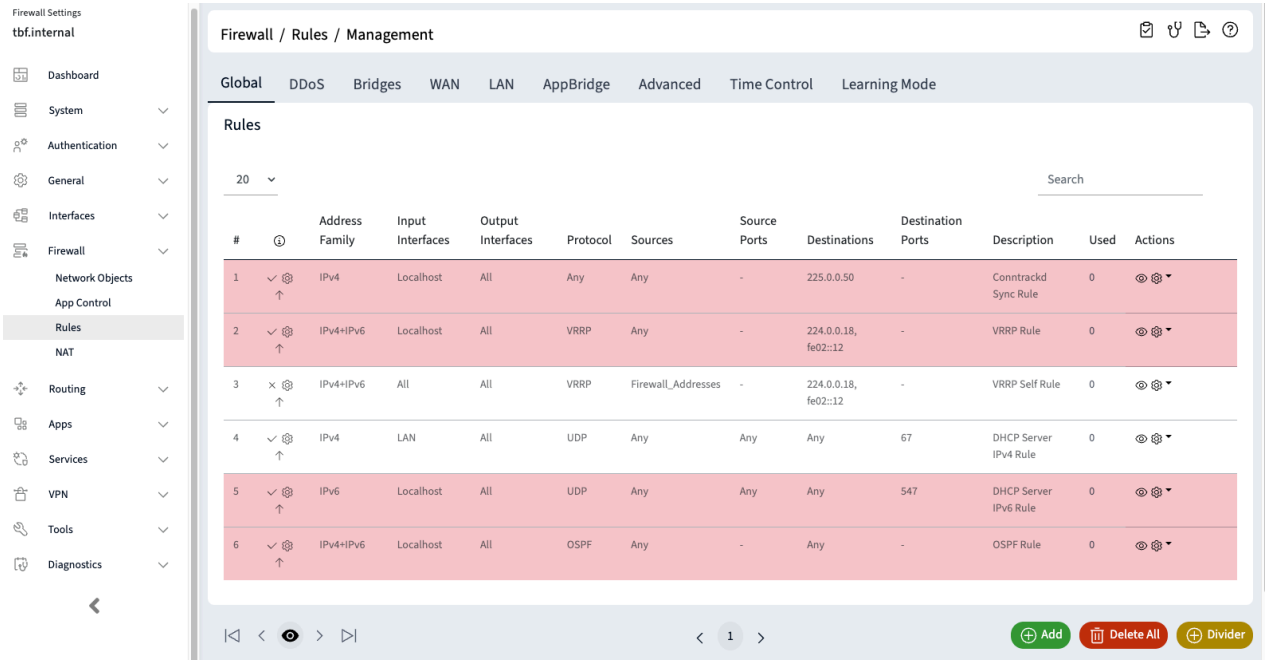


Figure 104: Firewall rules overview

Related topics

- *Firewall rule evaluation process* on page 139
- *Firewall rule search criteria* on page 139

Firewall rule evaluation process

TBF evaluates firewall rules in a specific order, and global rules take precedence over interface-specific rules.

The firewall evaluates rules in a two-tier hierarchy. First, the firewall checks global firewall rules that apply to each network interface. If a global rule matches the incoming traffic, TBF immediately applies that rule and stops processing. TBF evaluates no additional rules.

If no global rules match, TBF then evaluates interface-specific rules for the relevant network interface. This evaluation order allows global rules to override interface-specific settings. Network administrators gain centralized control over key traffic policies and granular control at the interface level.

Firewall rule search criteria

The firewall rule search function accepts specific criteria to help you locate rules quickly.

Search for specific firewall rules with the search function in the upper-right corner of the overview page. The search function recognizes and filters results based on the following criteria:

- **Protocol**: The communication protocol (such as TCP, UDP, or ICMP)
- **Source**: The originating network address or range
- **Destination**: The target network address or range
- **IP address**: Any specific IP address in source or destination fields
- **Port**: The network port number or port range
- **Description**: Any text contained in the description field of the rule

Enter a search term in the search field. TBF displays only the rules that match the criteria. This helps you quickly identify relevant rules in environments with many firewall configurations.

Bridge firewall rules (forward)

Bridge firewall rules filter traffic within bridge interfaces through a dedicated forwarding path that bypasses normal host processing.

Bridge firewall rules are a specialized type of firewall rule that work differently from standard firewall rules. When you create a bridge interface, TBF adds a new **Bridges** tab to the firewall **Rules**.

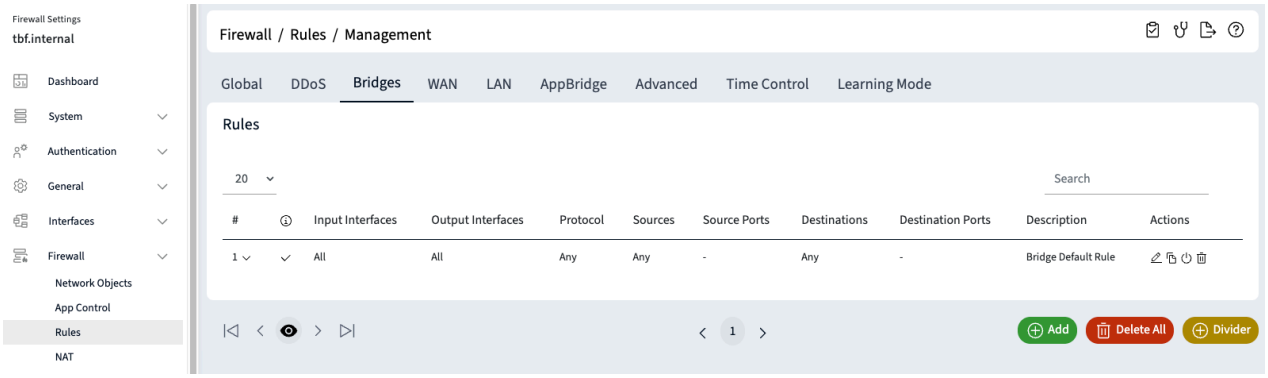


Figure 105: Firewall bridge rules

The main difference between standard firewall rules and bridge firewall rules is where they filter traffic. Bridge firewall rules filter traffic directly within the bridge interface. The bridge forwards this traffic through a separate, dedicated path that operates independently from normal forwarding in TBF. Bridge firewall rules only apply to traffic that flows through this dedicated forwarding path.

Standard global firewall rules do not affect bridge traffic unless you create a VLAN interface on top of the bridge. A single bridge supports multiple VLANs simultaneously. Bridge firewall rules let you match and control traffic that exists outside of any VLAN interface.

Important: For bridge firewall rules to work, select the bridge interface or its member interfaces as the input and output interfaces.

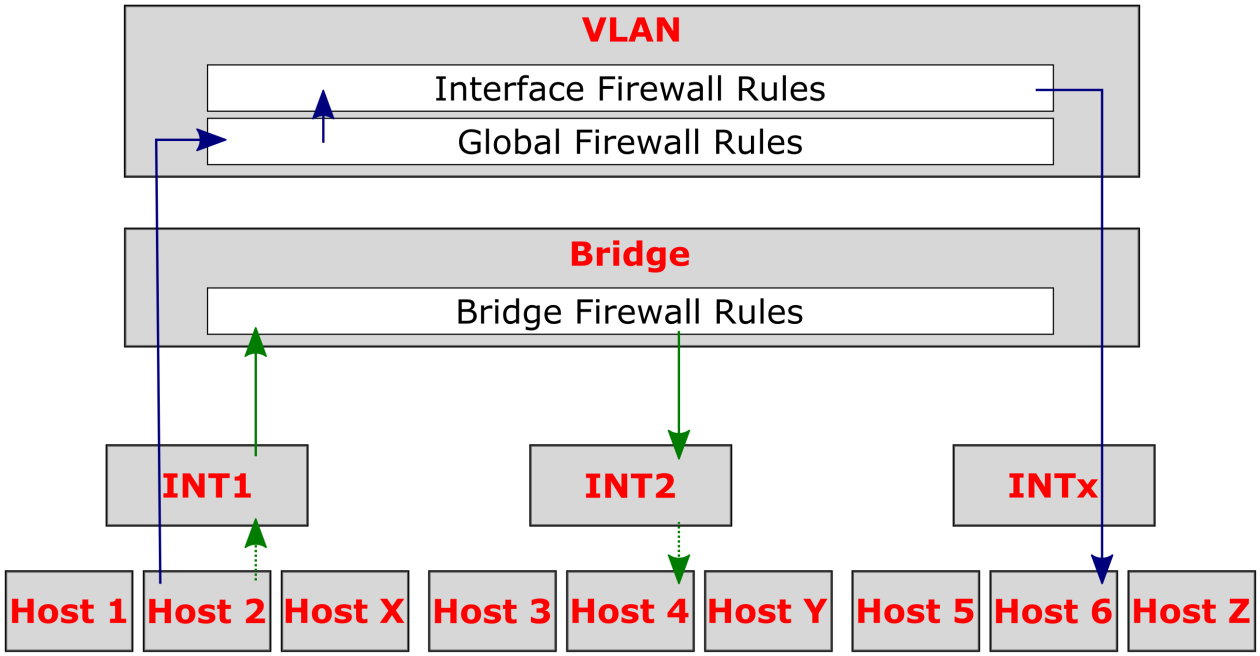


Figure 106: Firewall bridge rules

Attention: In high availability (HA) configurations, verify that interface names are identical on both systems. If the names differ, then exclude these rules from HA synchronization to help avoid synchronization outages.

Bridge firewall rule settings are similar to standard firewall rules, but they do not include advanced options such as:

- Traffic limiters
- Routing table selection

This limitation exists because the system performs routing at layer 2 for bridge traffic.

Related information
Firewall rules (forward and input) on page 131
Firewall rule search criteria on page 139

DNAT (prerouting)

Destination Network Address Translation (DNAT) modifies the destination IP address of incoming network packets to redirect traffic to internal hosts.

Find the DNAT rule settings under **Firewall > NAT**.

DNAT changes the destination IP address of network packets as the packets arrive at the firewall. This technique allows you to redirect incoming traffic from a public IP address to a specific private IP address on the internal network. For example, you can direct web traffic from the internet to an internal web server.

There are 3 tabs for different translation types:

- **Inbound (DNAT)**
- **Outbound (SNAT)**
- **Both (BiNAT)**

To view and manage DNAT rules, select the **Inbound (DNAT)** tab.

Search DNAT rules

To quickly locate specific rules, use the search field in the upper-right corner of the overview page. Search by any of these criteria:

- Protocol
- Source
- Destination
- IP address
- Port
- Description

Move DNAT rules

Reorganize DNAT rules by dragging and dropping them to different positions or using navigation buttons to move them between pages.

TBF organizes rules by interface and displays them in groups of 20 rules per page.

1. To move a rule within the current page, drag and drop user-created rules to different positions.
2. To save the new position, select **Save** in the bottom navigation.

The rule appears in its new position in the rules list.

You can also move a rule to the next or previous page, or to the first or last page. To do this, select the first cell of the firewall rule. Then use the arrow buttons in the bottom left. To see the descriptions, hover over the buttons.

Create and update DNAT rules

Add new DNAT rules or modify existing rules to redirect incoming traffic to internal hosts.

When you select **Add**, you create a new DNAT rule on the interface. You have various settings available for the rule. The rules contain the following sections:

- General settings
- Sources
- Destinations
- NAT settings
- Advanced
- Changes

1. Navigate to **Firewall > NAT**.
2. Select the **Inbound (DNAT)** tab.
3. To create a new rule, select **Add**. To modify an existing rule, select the rule.
4. Set the rule settings in each section as needed.

   For detailed information on each configuration section, see the related reference topics:

   - *DNAT general settings* on page 142
   - *DNAT sources settings* on page 143
   - *DNAT destinations settings* on page 143
   - *DNAT NAT settings* on page 144
   - *DNAT advanced settings* on page 144
5. To apply the rule configuration, select **Save**.

TBF creates or updates the DNAT rule and applies the rule to the selected interface. The rule appears in the rule list and becomes active immediately.

DNAT general settings

These settings set up basic DNAT rule properties that include enablement, interface assignment, NAT type, and address family settings.

General settings

Enabled

This setting enables or disables the rule.

Add Rule To

This setting determines where the rule appears in the **Rules** list on the overview page.

Interface

This setting specifies the interface for this rule. When you change the interface, TBF adds the rule to the end of the rule list of that interface.

NAT Type

This setting displays the NAT type.

No NAT

This setting excludes matching traffic from DNAT processing. This exclusion is useful for the creation of exceptions.

Address Family

This setting specifies whether the rule applies to IPv4, IPv6, or both. When you select IPv4+IPv6, TBF does not always generate rules for both protocols. This generation depends on the sources and destinations you define.

Protocol

This setting specifies the layer 2 protocol for the rule.

Description

This field adds a description for the rule.

DNAT sources settings

Set source IP addresses and ports for DNAT rules, including options for inverting match logic and mixing IPv4 and IPv6 addresses.

Sources settings

Add multiple source IP addresses and source ports, and mix IPv4 and IPv6 addresses. TBF automatically sets the appropriate rules.

Invert IP Match

This setting inverts the matching logic for IP addresses, MAC addresses, and ports.

Source IPs

This setting allows you to select the source IP addresses.

DNAT destinations settings

Set destination IP addresses and ports for DNAT rules. You can also invert match logic and mix IPv4 and IPv6 addresses.

Destinations settings

Add multiple destination IP addresses and destination ports, and mix IPv4 and IPv6 addresses. TBF automatically sets the appropriate rules.

Invert IP Match

This setting inverts the matching logic for IP addresses and ports.

Destination IPs

This setting allows you to select the destination IP addresses.

DNAT NAT settings

Translation IP addresses and ports for DNAT rules define how the system rewrites destination traffic, with options for static port assignment.

NAT translation settings

Translation IP

This field specifies the IP address that the system uses to rewrite the destination address of network packets.

Translation Port

This field specifies the port that the system uses to rewrite the destination port of network packets, when applicable.

Static Port(s)

This setting uses a static port instead of a randomly selected port. Some protocols, such as VoIP, require a static port.

DNAT advanced settings

These settings set advanced DNAT rule options that include logging, routing tables, input interfaces, and expiry dates.

Advanced settings

Log this Rule

This setting logs the traffic of the rule and adds a prefix to help you identify the traffic more easily.

WARNING: Firewall logging is a resource-intensive operation that generates many log entries.

Advanced Settings

This setting displays additional advanced settings.

Routing Table

This setting applies the DNAT rule to traffic that uses the specified routing table. TBF requires a firewall rule to first match the traffic and set the routing table for the connection.

Input Interface

This setting specifies the input interface that packets must match.

Enable Expiry Date

This setting sets an expiration date for the rule.

Expiry Date

This field specifies the expiration date. Enter a date in the format `dd.mm.yyyy`.

SNAT (postrouting)

Source NAT (SNAT) changes the source IP address of outgoing network packets. The firewall applies SNAT as the final processing step when handling new network packets.

SNAT helps you hide the original sender of network packets by replacing the source address with a different IP address that you specify. This technique is useful for maintaining network isolation and managing traffic flow.

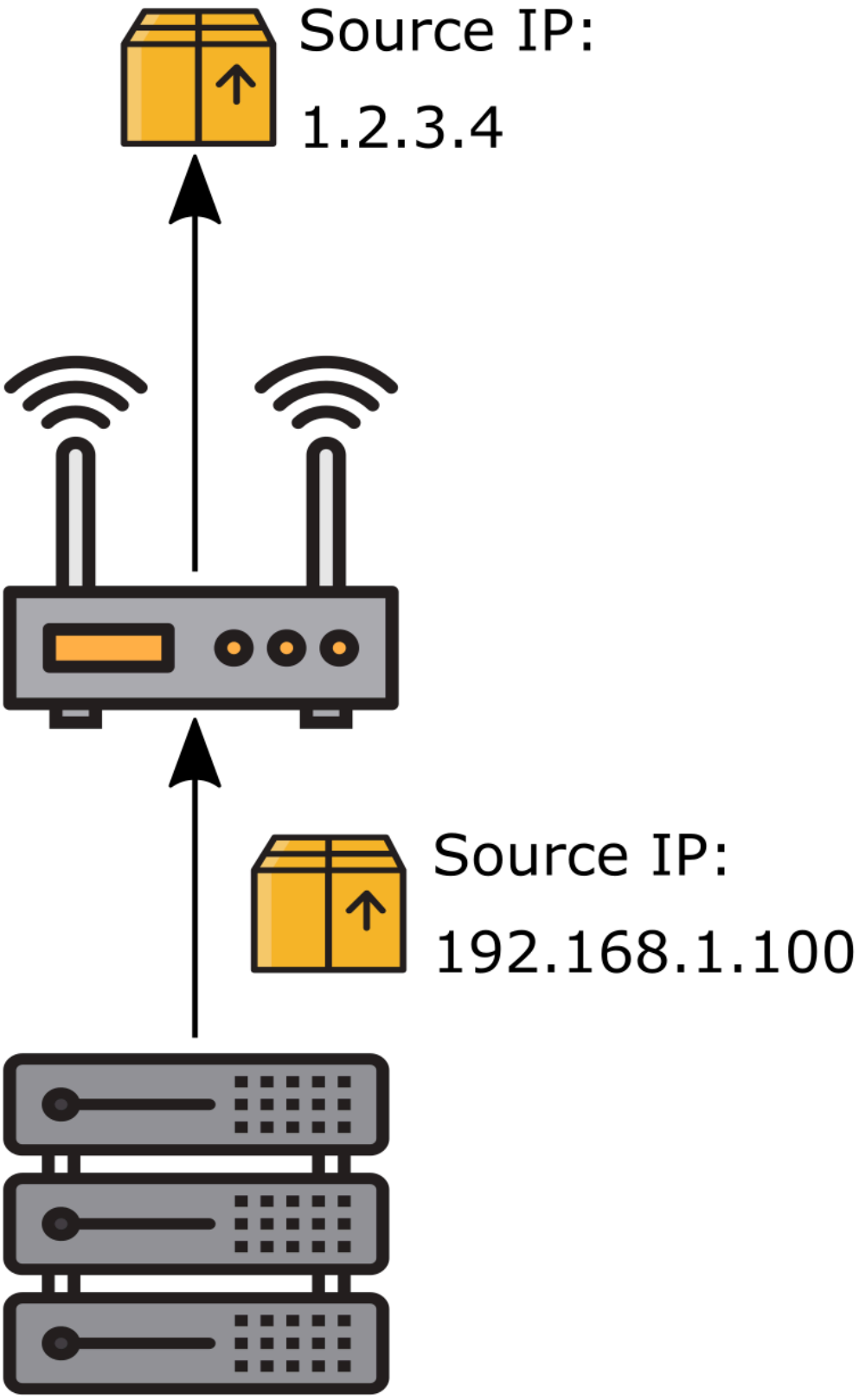


Figure 107: SNAT

srip and *Good Ware* from *Flaticon* created the icons for this image.

By default, when traffic leaves the local network for external destinations, the system automatically assigns a new source IP address that matches the WAN IP. However, this default behavior does not work for all network configurations.

For example, in high-availability setups, you need a shared virtual WAN address instead. See *HA setup examples* on page 386 for more details.

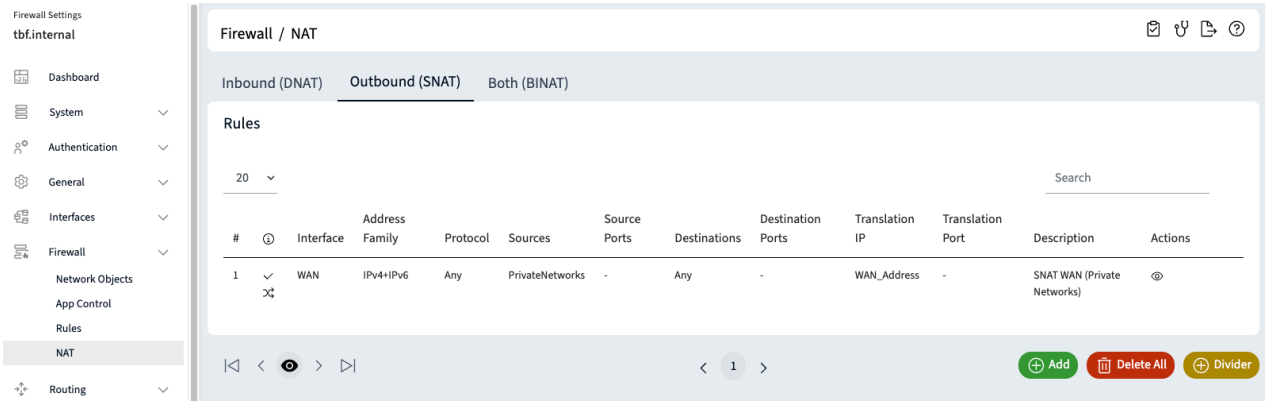


Figure 108: SNAT default rule

Find the SNAT rule settings under **Firewall > NAT**.

The NAT section contains three tabs:

- **Inbound (DNAT)**
- **Outbound (SNAT)**
- **Both (BiNAT)**

To view and manage SNAT rules, select the **Outbound (SNAT)** tab.

Move SNAT rules

Reorder SNAT rules by dragging and dropping the rules or using navigation buttons to move the rules between pages.

TBF organizes rules by interface and displays the rules in groups of 20 rules per page.

Perform the following steps:

1. To move a rule within the current page, drag and drop user-created rules to different positions.
2. To save the new position, select **Save** in the bottom navigation.

The rule appears in its new position.

To move a rule to the next or previous page, or to the first or last page:

1. Select the rule by selecting the first cell of the firewall rule.
2. Use the arrow buttons in the bottom left.
3. To see the button descriptions, hover over the buttons.

Create and update SNAT rules

Create new SNAT rules or modify existing SNAT rules by setting up the options in each section.

When you select **Add**, you create a new SNAT rule on the interface. TBF organizes the rule settings into the following sections:

- General settings
- Sources
- Destinations
- NAT settings
- Advanced

Perform the following steps:

1. To create a new SNAT rule, select **Add**.
2. Set the settings in each section as needed:
    a. Set up *general settings* on page 147.
    b. Define *sources* on page 147.
    c. Specify *destinations* on page 148.
    d. Set *NAT settings* on page 148.
    e. Adjust *advanced* on page 148 options if needed.
3. Save the rule configuration.

The new SNAT rule appears in the rules list.

View rule information such as creation date, modification date, and the user who last modified the rule. For details, see *changes* on page 149.

General

Set basic SNAT rule properties, such as the rule status, assigned interface, and protocol settings.

Settings

Enabled

This setting enables or disables the rule.

Add Rule To

This setting determines where the rule appears in the **Rules** list on the overview page.

Interface

This setting specifies the interface for this rule. When you change the interface, TBF adds the rule to the end of the rule list of that interface.

NAT Type

This setting displays the NAT type.

No NAT

This setting excludes matching traffic from SNAT processing. This setting is useful for creating exceptions.

Address Family

This setting specifies whether the rule applies to IPv4, IPv6, or both. When you select IPv4+IPv6, TBF generates rules for one or both protocols depending on the sources and destinations you define.

Protocol

This setting specifies the layer 2 protocol for the rule.

Description

This field adds a description for the rule.

Sources

Set up source IP addresses and ports for SNAT rules. You can mix IPv4 and IPv6 addresses.

You can add multiple source IP addresses and source ports, and mix IPv4 and IPv6 addresses. TBF automatically sets up the appropriate rules.

Invert IP Match

This setting inverts the matching logic for IP addresses, MAC addresses, and ports.

Source IPs

This setting selects the source IP addresses.

Destinations

Set up destination IP addresses and ports for SNAT rules, and mix IPv4 and IPv6 addresses.

You can add multiple destination IP addresses and destination ports, and mix IPv4 and IPv6 addresses. TBF automatically sets up the appropriate rules.

Invert IP Match

This setting inverts the matching logic for IP addresses and ports.

Destination IPs

This setting selects the destination IP addresses.

NAT

Set up translation IP addresses and ports that the system uses to rewrite SNAT traffic.

You can define the **Translation IP** and, when applicable, the **Translation Port**. This address is the address that the system uses to rewrite the traffic. The firewall changes the source of the network packet to this address or port.

Translation IP

This setting specifies the IP address that the system uses to rewrite the source address of network packets.

Translation Port

This setting specifies the port that the system uses to rewrite the source port of network packets, when applicable.

Static Port(s)

This setting uses a static port instead of a random port. By default, TBF selects a random port for the traffic. This behavior is not suitable for certain situations. Some protocols, such as VoIP, require a static port.

Advanced

Set up advanced SNAT rule options, such as logging, routing tables, input interfaces, and expiry dates.

In the advanced settings, you can set up the following options:

Log this Rule

This setting logs the traffic of the rule and adds a prefix to help you identify the traffic more easily.

WARNING: Firewall logging is a resource-intensive operation that generates many log entries.

Advanced Settings

This setting displays additional advanced settings.

Routing Table

This setting applies the SNAT rule to traffic that uses the specified routing table. A firewall rule must first match the traffic and set the routing table for the connection.

Input Interface

This setting specifies the input interface that packets must match.

Enable Expiry Date

This setting sets an expiry date for the rule.

Expiry Date

This field specifies the expiry date. Enter a date in the format `dd.mm.yyyy`.

Search SNAT rules

Search for SNAT rules using various criteria including protocol, source, destination, IP address, port, and description.

In the upper-right corner of the overview page, search for rules using the following criteria:

- Protocol
- Source
- Destination
- IP address
- Port
- Description

SNAT changes

View rule creation and modification information including dates and user details.

At the bottom of each rule, you can view the following information:

- **Created date**
- **Modified date**
- The user who last modified the rule (**Modified user**)

BiNAT (prerouting and postrouting)

Bidirectional Network Address Translation (BiNAT) modifies both source and destination IP addresses of network packets, combining Destination Network Address Translation (DNAT) and Source Network Address Translation (SNAT) functionality.

Overview

Find the BiNAT rule settings under **Firewall > NAT**.

BiNAT modifies both the source IP address and destination IP address of a network packet. This feature combines DNAT and SNAT functionality into a single rule.

Note: BiNAT only works for a 1:1 mapping of a host to another IP address.

Tabs

The NAT interface has 3 tabs for different translation types:

- **Inbound (DNAT)**
- **Outbound (SNAT)**
- **Both (BiNAT)**

To access and manage BiNAT rules, select the **Both (BiNAT)** tab.

Move BiNAT rules

Change the order of BiNAT rules to control their priority and execution sequence.

TBF processes BiNAT rules in order from top to bottom. To control their priority and help provide proper execution sequence, change the order of rules.

Perform the following steps:

1. Navigate to **Firewall > NAT**.
2. Select the **Both (BiNAT)** tab.
3. Locate the rule you want to move in the rules list.
4. Use the move controls to reposition the rule:

- To move the rule higher in priority, select the up arrow
- To move the rule lower in priority, select the down arrow
- Drag and drop the rule to the desired position

5. To save the new rule order, select **Apply**.

TBF saves the BiNAT rule in its new position and processes the rule according to its new priority order.

Create and update BiNAT rules

To set up bidirectional network address translation, create new BiNAT rules or modify existing ones.

BiNAT rules help provide bidirectional network address translation, combining both DNAT and SNAT functionality in a single rule for 1:1 IP address mapping.

Perform the following steps:

1. Navigate to **Firewall > NAT**.
2. Select the **Both (BiNAT)** tab.
3. To create a new rule, select **Add**. To update an existing rule, select the rule in the list.
4. Set the rule settings in the available tabs:
   a. Set general settings including rule name and description
   b. Set original destination parameters
   c. Set NAT translation settings
   d. Adjust advanced settings if needed
5. To create or update the rule, select **Save**.
6. To apply the changes, select **Apply**.

TBF creates or updates the BiNAT rule. The BiNAT rule becomes active in the firewall configuration.

General

The general settings for BiNAT rules include settings to identify the rule and basic parameters.

Set basic BiNAT rule parameters and settings to identify the rule.

Rule Name

This field specifies a descriptive name to help identify the purpose of the BiNAT rule.

Description

This field specifies an optional detailed description of the function and purpose of the rule.

Enabled

This setting enables or disables the rule. Disabled rules remain in the configuration but the device does not process the rules.

Interface

This setting specifies the network interface where the BiNAT rule applies.

Protocol

This setting specifies the protocol for the rule. Select from:

- **TCP**
- **UDP**
- **Any**

Original destinations

This topic describes settings to define original destination addresses and ports in BiNAT rules.

Define the original destination parameters that trigger BiNAT.

Original Destination IP

This field specifies the original destination IP address or network to translate. This value can be a single IP address, network range, or alias.

Original Destination Port

This field specifies the original destination port or port range. To use any port, leave the field empty.

Source IP

This field optionally restricts the source IP address or network. To match any source, leave the field empty.

Source Port

This field optionally restricts the source port. To match any source port, leave the field empty.

NAT

This topic describes network address translation settings for BiNAT rules, such as destination and source translation.

Set the translation settings for both destination and source addresses in BiNAT rules.

Translated Destination IP

This field specifies the IP address to translate the original destination to. This IP address is the DNAT component of BiNAT.

Translated Destination Port

This field defines the port to translate the original destination port to. To use the original port, leave the field empty.

Translated Source IP

This field specifies the IP address to translate the source to. This IP address is the SNAT component of BiNAT.

Translated Source Port

This field specifies the port range for source port translation. To assign the port automatically, leave the field empty.

Translation Type

This setting defines the type of address translation.

Advanced

This topic describes advanced settings for BiNAT rules, such as logging, routing tables, and expiry dates.

Set advanced settings for BiNAT rules to control logging, routing, and rule lifecycle.

Log this Rule

This setting enables logging for this BiNAT rule. To help identify log entries, add a prefix.

WARNING: Firewall logging is resource-intensive and generates many log entries.

Routing Table

This setting selects the routing table to use for this rule. The default is **Main**.

Rule Priority

This setting sets the priority level to process rules. The device processes higher-priority rules first.

Expiry Date

This field specifies an optional expiry date for the rule. The device automatically disables the rule after this date.

Connection Tracking

This setting enables or disables connection tracking for this rule. This setting affects how the device handles stateful connections.

BiNAT rule changes tracking

This topic describes how to track and monitor changes made to BiNAT rules, such as modification history and audit information.

For audit and troubleshooting purposes, monitor and track changes made to BiNAT rules.

Change History

View a chronological list of modifications made to BiNAT rules, such as timestamps and user information.

Rule Modifications

Track specific changes to rule parameters, such as before and after values for configuration settings.

User Activity

Monitor which users made changes to BiNAT rules and when the modifications occurred.

Configuration Backup

To compare and roll back configurations, access previous versions of BiNAT rule configurations.

Audit Logs

To monitor compliance and security, review detailed audit logs that show BiNAT rule changes.

Search BiNAT rules

To quickly locate rules in large configurations, find specific BiNAT rules with search and filter options.

To quickly locate specific rules based on various criteria when managing multiple BiNAT rules, use search and filter functionality.

Perform the following steps:

1. Navigate to **Firewall > NAT**.
2. Select the **Both (BiNAT)** tab.
3. To filter rules, use the search options:
    - To find rules by name or description, enter text in the search field.
    - To narrow results by specific criteria, use filter options.
    - To sort rules, select column headers.
4. To locate the desired rule, review the filtered results.
5. To view or edit its configuration, select a rule.

The search results display matching BiNAT rules to help you quickly locate and manage specific rules.

Firewall rules advanced

This topic describes advanced firewall features including geo blocking, cloud service integration, and custom rule configuration.

To access the advanced firewall rules settings, navigate to **Firewall > Rules > Advanced**.

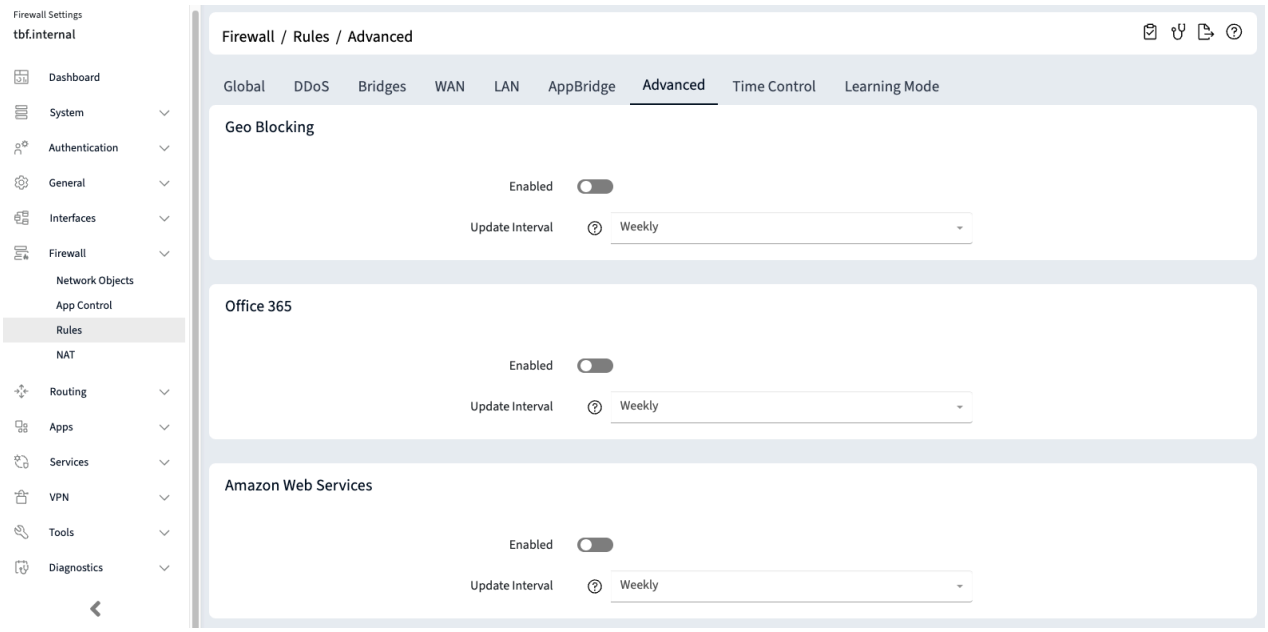


Figure 109: Advanced firewall rules

Geo blocking

This topic describes how to block or allow traffic based on geographic location to help prevent access from specific countries or regions.

Geo blocking allows you to control network access based on the geographic location of IP addresses. This feature helps prevent access from specific countries or regions that may pose security threats.

How geo blocking works

The firewall uses IP geolocation databases to determine the geographic origin of network traffic. To block or allow traffic from specific countries, regions, or continents, create rules.

Settings

Country Selection

This setting selects specific countries to block or allow from a comprehensive list of countries.

Region Blocking

This setting blocks entire geographic regions or continents with a single rule.

Action Type

This setting blocks or allows traffic from selected locations.

Whitelist Mode

This setting enables whitelist mode to allow only traffic from specified countries and block other traffic.

Database Updates

This setting automatically updates the IP geolocation database to maintain accuracy.

Use cases

- Block traffic from countries with high cybersecurity threats
- To comply with data protection regulations, restrict access from certain regions
- Help prevent unauthorized access from specific geographic locations
- For sensitive services, create location-based access policies

Office 365

This topic describes how to set firewall rules specifically for Microsoft Office 365 services and applications.

These settings set specialized firewall rules for Microsoft Office 365 services to help provide proper connectivity while assisting in establishing security. Office 365 requires specific network settings for performance.

Office 365 service categories

Microsoft categorizes Office 365 services based on their network requirements and security considerations:

Optimize Category

These are critical services that require direct internet access with low latency, such as Exchange Online and SharePoint Online.

Allow Category

These services function through proxy servers but benefit from direct access, such as Microsoft Teams and OneDrive.

Default Category

These services operate through standard firewall rules and proxy settings.

Settings

Service Selection

This setting selects specific Office 365 services to set up, such as Exchange, SharePoint, Teams, and OneDrive.

Endpoint Management

This setting automatically manages Microsoft-published IP addresses and URLs for Office 365 services.

Bypass Options

This setting sets up proxy bypass rules for **Optimize Category** services that require direct internet access.

Quality of Service

This setting applies QoS policies to prioritize Office 365 traffic for a good user experience.

Update Frequency

This setting sets how often the firewall updates Office 365 endpoint information from Microsoft.

Best practices

- Allow direct internet access for optimize category services.
- Update Office 365 endpoint information.
- Monitor Office 365 connectivity and performance.
- Set up appropriate QoS policies for real-time services.

Amazon Web Services

This topic describes how to set up firewall integration with Amazon Web Services (AWS) cloud services and infrastructure.

Set up firewall rules and integration for AWS cloud infrastructure. With this configuration, you manage traffic to AWS services, VPC connections, and hybrid cloud deployments.

AWS service integration

The firewall integrates with various AWS services to help provide comprehensive cloud security:

VPC Integration

To assist in establishing security for hybrid connectivity, connect to AWS Virtual Private Clouds (VPCs) through a VPN or Direct Connect.

Security Groups

To assist in establishing a layered security architecture, coordinate firewall rules with AWS security groups.

AWS Services

Set up access rules for AWS services, such as S3, EC2, RDS, and Lambda functions.

CloudWatch Integration

To monitor centrally, send firewall logs and metrics to AWS CloudWatch.

Settings

AWS Region Selection

To optimize routing, specify AWS regions for service access.

Service Endpoints

Set up access to AWS service endpoints, such as public and VPC endpoints.

API Access Control

Manage access to AWS APIs and control which services the network accesses.

Traffic Inspection

To detect threats and anomalies, enable deep packet inspection for AWS-bound traffic.

Load Balancing

To distribute traffic across AWS resources, set load balancing rules.

Security considerations

- To authenticate, use AWS IAM roles and policies.
- Enable encryption for data in transit to AWS services.
- Monitor and log AWS-related network traffic.
- Implement least privilege access principles.
- Review and update AWS service access rules.

Google

This topic describes how to set firewall rules for Google services and applications.

Set up firewall rules specifically for Google services and applications, such as Google Workspace, Google Cloud Platform, and other Google services that require network access.

Google service categories

The device organizes Google services into different categories based on their network requirements:

Google Workspace

This category includes Gmail, Google Drive, Google Meet, Google Calendar, and other productivity applications.

Google Cloud Platform

This category includes Compute Engine, Cloud Storage, BigQuery, and other GCP services for cloud infrastructure.

Consumer Services

This category includes YouTube, Google Search, Google Maps, and other consumer-facing Google services.

Developer Services

This category includes Google APIs, Firebase, Android services, and development tools.

Settings

Service Selection

To allow or restrict access based on organizational policies, select specific Google services.

Domain Filtering

For granular control, set rules based on Google domain names and subdomains.

Application Control

Control access to specific Google applications and allow or block others.

Safe Search

For content filtering, enforce Google Safe Search settings through firewall rules.

YouTube Restrictions

Apply YouTube restricted mode and content filtering policies.

Common use cases

- Allow Google Workspace and block consumer Google services.
- Restrict YouTube access during business hours.
- Enable Google Cloud Platform access for development teams.
- Enforce Safe Search for educational environments.
- Control access to Google APIs and developer services.

Best practices

- Update Google service IP address ranges and domains.
- Monitor Google service usage and performance.
- Implement time-based access controls for productivity.
- Use SSL inspection carefully with Google services.

Miscellaneous

This topic describes additional firewall options for specialized requirements.

This topic describes how to set additional firewall settings that do not fit into standard categories but help provide functionality for specialized network requirements.

General settings

Connection Tracking

Set connection tracking parameters, such as timeout values and maximum connections.

Fragment Handling

Set how the firewall processes fragmented packets and reassembly options.

ICMP Settings

Control how the device handles ICMP packets, such as ping responses and how the device forwards error messages.

TCP Options

Set TCP-specific settings, such as assistance with protection against SYN floods and window scaling.

Advanced settings

Packet Marking

For Quality of Service (QoS) or routing purposes, mark packets with DSCP or TOS values.

Rate Limiting

To help prevent abuse and help provide fair usage, apply rate limiting to specific traffic types.

Custom Chains

For complex firewall logic and to organize rules, create custom rule chains.

Helper Modules

Enable connection tracking helper modules for specific protocols, such as FTP, SIP, or H.323.

Debugging and troubleshooting

Packet Tracing

To diagnose connectivity issues, enable packet tracing for specific rules or traffic flows.

Rule Statistics

View detailed statistics for firewall rules, such as packet counts and byte counters.

Debug Logging

Set detailed debug logging for how the firewall operates and processes rules.

Performance Monitoring

Monitor firewall performance metrics, such as CPU usage and memory consumption.

Custom rules

This topic describes how to create and implement custom nftables rules for advanced firewall setups.

Custom nftables rules allow you to implement advanced firewall configurations that go beyond the standard rule interface. This feature is intended for experienced administrators who need specialized firewall behavior.

WARNING: Custom nftables rules can override existing firewall configurations. Test the custom rules thoroughly before implementation in production environments.

Perform the following steps:

1. Navigate to **Firewall > Rules > Advanced > Custom Rules**.
2. To create a new rule, select **Add Custom Rule**.
3. Set the custom rule parameters:
   a. Enter a descriptive name for the custom rule.
   b. Select the nftables table where the rule applies.
   c. Specify the chain for rule insertion.
   d. Set the rule priority or position.
4. Enter the custom nftables rule syntax in the rule editor.

   Use proper nftables syntax. Examples include:

   - `ip saddr 192.168.1.0/24 tcp dport 80 accept`
   - `meta mark 0x1 counter drop`
   - `limit rate 10/minute accept`
5. To validate the rule syntax, select **Validate**.
6. To store the custom rule, select **Save**.
7. To apply the custom rule in the firewall, select **Apply**.

The custom nftables rule becomes active in the firewall configuration.

To verify that the custom rule functions as expected, monitor firewall logs and performance. To verify proper operation, use the rule statistics.

Firewall time control

This topic describes how to access firewall time control settings to create time-based restrictions for firewall rules.

The firewall time control feature allows you to set specific time periods when firewall rules are active or inactive. To access these settings, navigate to **Firewall > Rules > Time Control**.

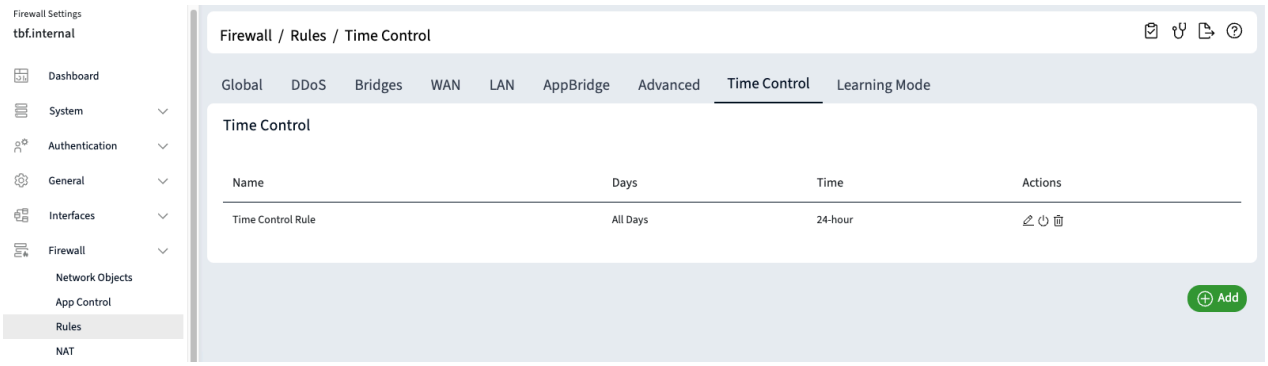


Figure 110: Time control rules

For detailed information on how time control works, see *Time control* on page 158.

Time control

Time control blocks define specific days and hours when firewall rules apply to control traffic based on time schedules.

Create time control blocks to use within firewall rules. Each time control block defines specific days and hours when the rule applies.

To allow or block traffic based on time schedules, use these blocks. The firewall evaluates and applies the entire rule only during the time range you specify.

Important: The firewall does not automatically terminate existing connections when they fall outside the time range you define. Time control only affects new connection attempts.

App control

App control enables you to create advanced firewall rules with application-specific information instead of basic network details.

Find the app control settings under **Firewall > App Control**.

Standard firewall rules work with basic network information like ports, protocols, and IP addresses. However, they cannot identify or control specific applications that run on the network. App control uses application definitions and signatures to create firewall rules that recognize and manage individual applications.

Important: App control relies on the intrusion detection system to function correctly. Therefore, enable intrusion detection first. Both app control and intrusion detection share some configuration options and modify the same underlying system settings.

App control components

App control includes several essential components that work together:

- *Apps* on page 159: These are built-in and custom application definitions that identify specific programs.
- *Flows* on page 162: These are network connection handling and processing methods.
- *Categories* on page 162: These are organized groups of related applications for easier management.
- *Rules* on page 163: These are firewall rules that understand and control applications.
- *Settings* on page 165: These are configuration options for the app control system.

Apps

TBF includes several predefined applications and allows you to create custom application definitions for application-aware firewall rules.

Find the application definitions under **Edit** in the **Actions** column.

Note: If an application is missing or if you have suggestions for new applications, send us an email. We are happy to add new applications to TBF.

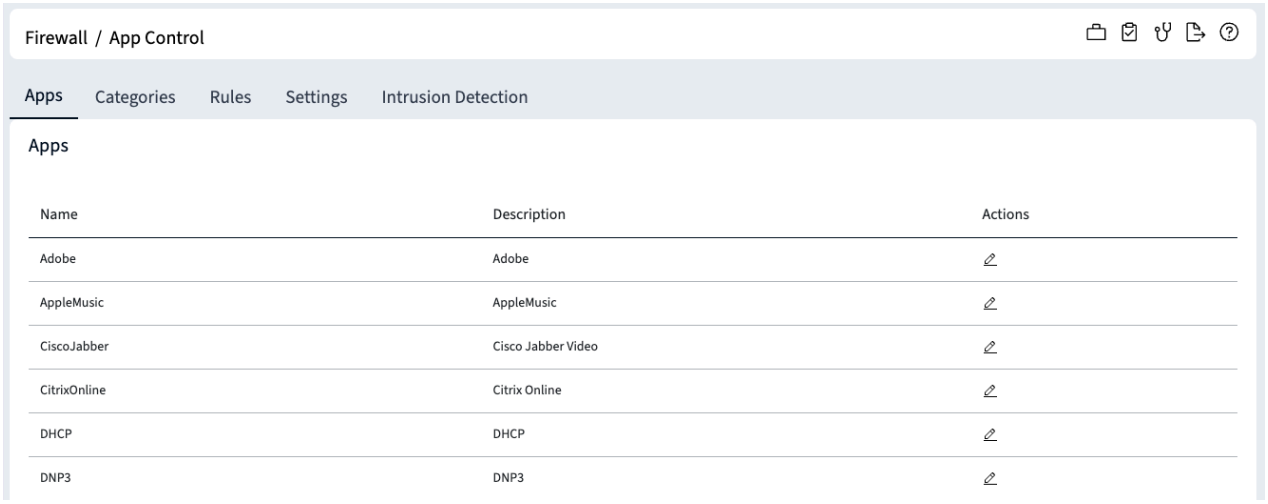


Figure 111: App control

Define new applications by creating custom application definitions. Because most applications use HTTP or HTTPS protocols, the GUI provides 3 predefined options for convenience. To define rules that the standard GUI options do not cover, use the custom option.

For more detailed information, see the *Suricata user guide*.

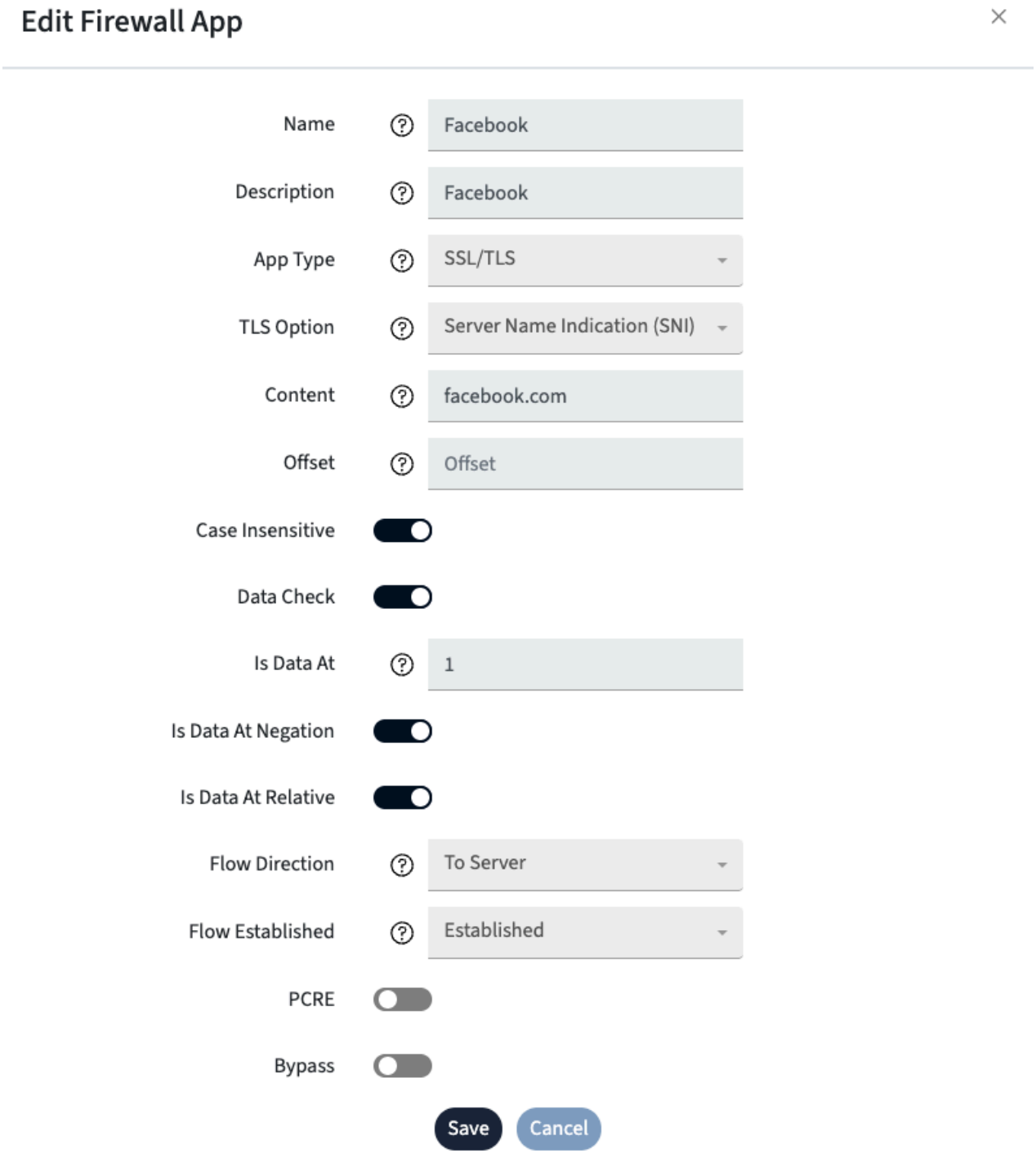


Figure 112: Edit firewall app

Settings

Name

This field specifies a unique identifier for the application. TBF requires a unique name across the applications.

Description

This field accepts a user-defined description that explains the purpose or function of the application.

App Type

This setting determines the application type. Select from:

- **HTTP/SSL/TLS**: This option matches both HTTP and SSL/TLS encrypted connections.
- **SSL/TLS**: This option matches only SSL/TLS encrypted connections.
- **HTTP**: This option matches only HTTP connections.
- **JA3**: This option defines a unique fingerprint based on the encryption parameters of an SSL/TLS connection. JA3 ties to the specific encryption parameters that a server or client uses, which makes JA3 useful for identifying fixed clients where the encryption options remain consistent. A JA3 hash also appears in the log files for each established connection.
- **Custom**: This option allows you to define custom application matching criteria using advanced rules.

App Type Option

For **SSL/TLS**, **HTTP**, or **JA3** application types, this field displays different options that you search for within a connection. For example, with **SSL/TLS**, you match against the Server Name Indication (SNI) or certificate fields of the connection.

Note: After the initial connection handshake, encrypted connections provide no additional readable information.

Content

This setting specifies the content pattern to match against in the selected **Option** field.

Offset

This field sets an offset position within the selected **Option** field. The offset helps improve matching performance. When you specify no offset, TBF searches the entire field content.

Case Insensitive

This setting makes content matches case insensitive, and allows matches regardless of uppercase or lowercase letters.

Data Check

This setting opens additional settings for matching specific data patterns within the connection.

Is Data At

This field checks whether there is additional data at the specified position within the connection.

Is Data At Negation

This setting reverses the logic of the **Is Data At** field, and converts the logic into a NOT **Is Data At** condition.

Is Data At Relative

This setting makes the **Is Data At** match relative to the content matched in the **Content** field.

Flow Direction

This setting determines the flow direction. For HTTP or SSL/TLS connections, the flow direction is typically **To Server**.

Flow Established

This setting checks for an established connection flow. For HTTP or SSL/TLS connections, select **Established**.

PCRE

This setting enables an additional regular expression (PCRE) check for advanced pattern matching. Leave the **Content** field empty and use only the **PCRE** match if necessary.

PCRE Content

This field contains the regular expression pattern for **PCRE** matching, for example, `/belden.com$|belden.de$/i`.

Bypass

This setting bypasses intrusion detection processing when a match occurs, and improves performance for trusted applications.

Flows

Flows represent connections between servers and clients. Attributes such as IP addresses, protocol, and ports identify these connections.

For app control, flows are a fundamental concept. A flow represents a connection between a server and a client. Attributes such as IP addresses, protocol, and ports identify a flow.

The device typically obtains application data only when a client and server establish a connection. For example, with SSL/TLS connections, the TCP connection needs to complete the TCP handshake before certificate and SNI information becomes available.

App control blocks or accepts the connection after the client and server establish the initial connection.

WARNING: To allow the connection to start, first create a *firewall rule* on page 131. TBF executes the app control AFTER it processes the standard firewall rules.

Categories

TBF organizes applications into categories for easier management and policy application.

TBF organizes applications into categories for easier management. Use the default built-in categories or create custom categories.

Categories are logical groups of applications. To apply policies to multiple related applications at once, use categories in the app control rules.

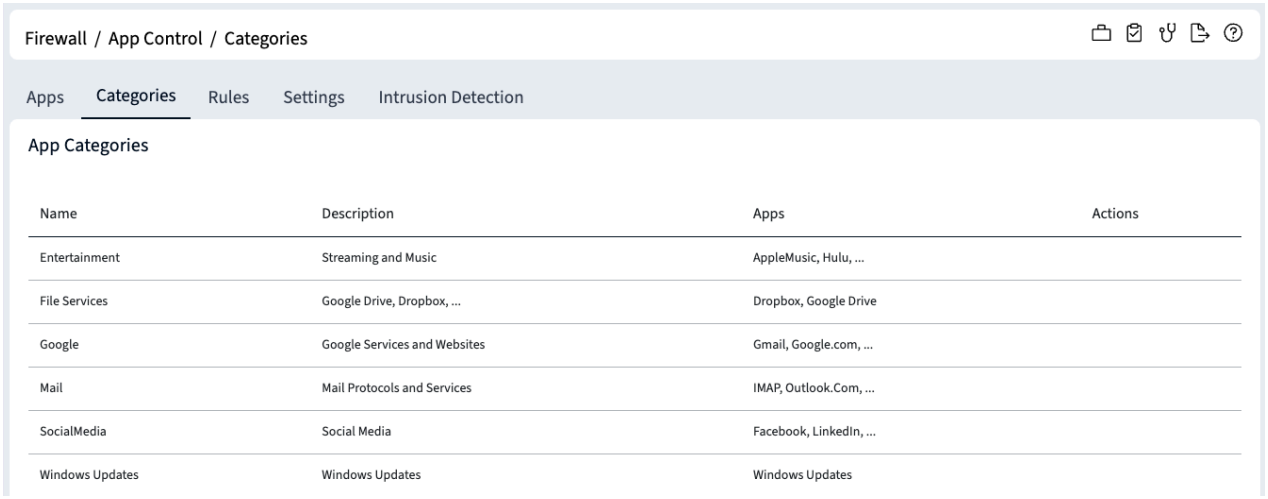


Figure 113: App categories

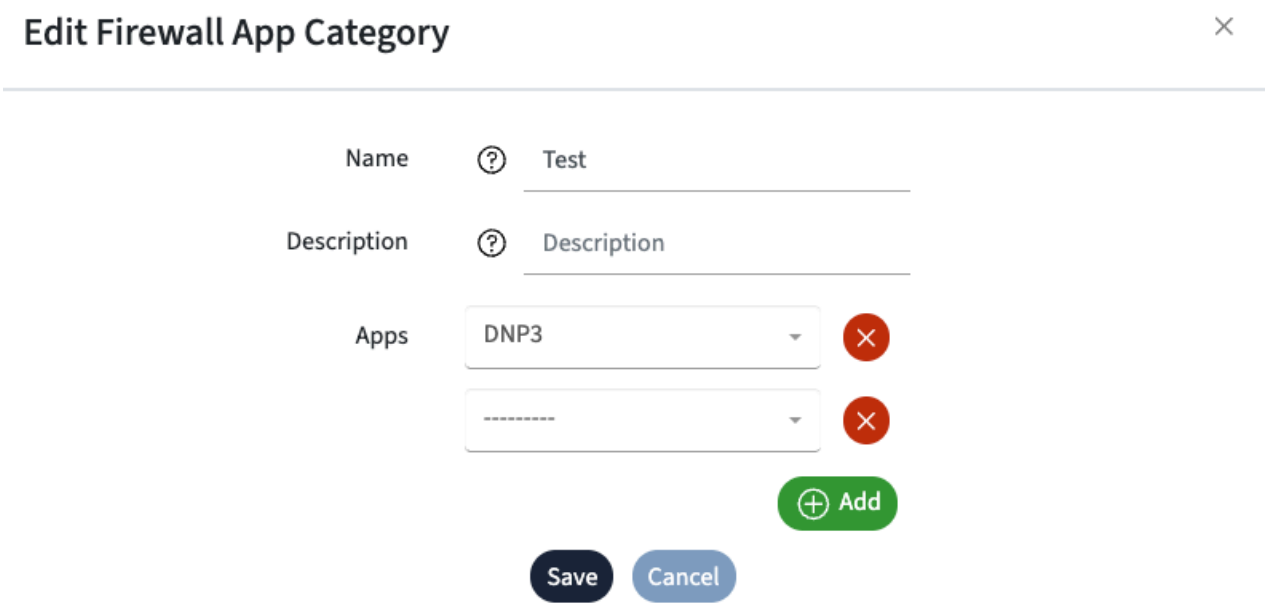


Figure 114: Edit firewall app category

Rules

App control rules work similarly to standard firewall rules but add application-aware filtering capabilities.

App control rules work similarly to standard firewall rules. To narrow down the match criteria, use:

- **IP Version**
- **Protocol**
- **Source IP**
- **Source Port**
- **Destination IP**
- **Destination Port**

The key difference is that you can also add **Apps** and **App Categories** to a rule to enable application-aware filtering.

To prioritize traffic, also assign quality of service (QoS) policies to rules that match.

Note: TBF processes application rules differently than normal firewall rules. TBF processes application rules in the following fixed order: `pass`, `reject`, `drop`, `match`. While you can change the settings to process `pass` rules last, you cannot modify the order of app control rules themselves.

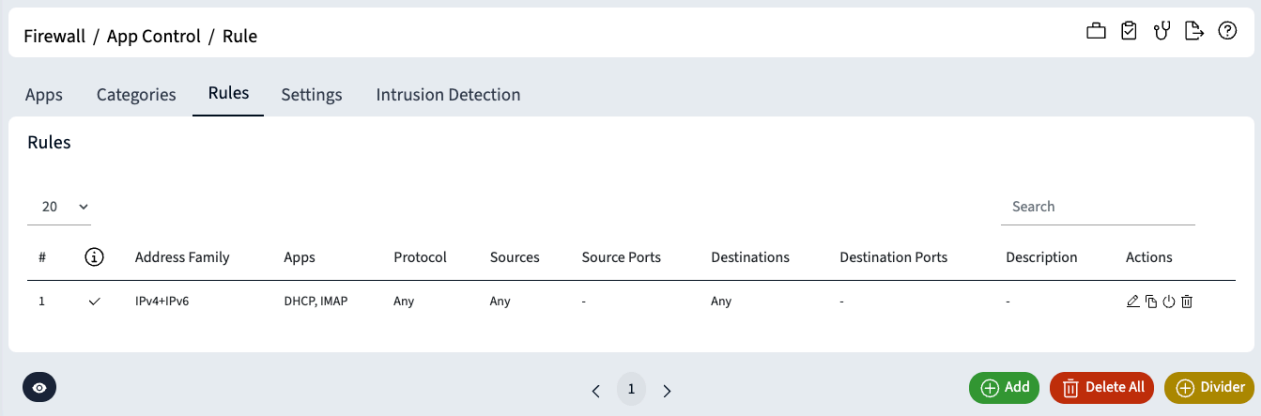


Figure 115: Rules

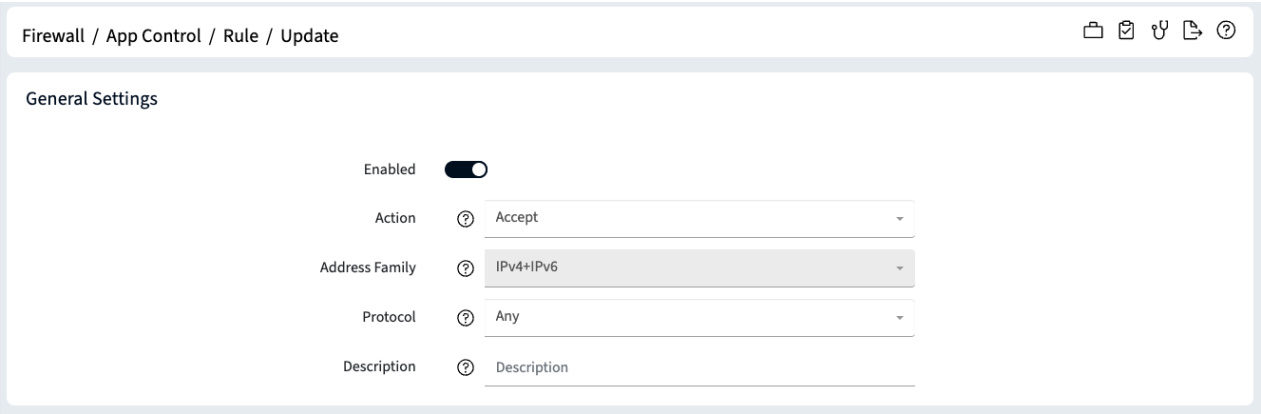


Figure 116: General settings

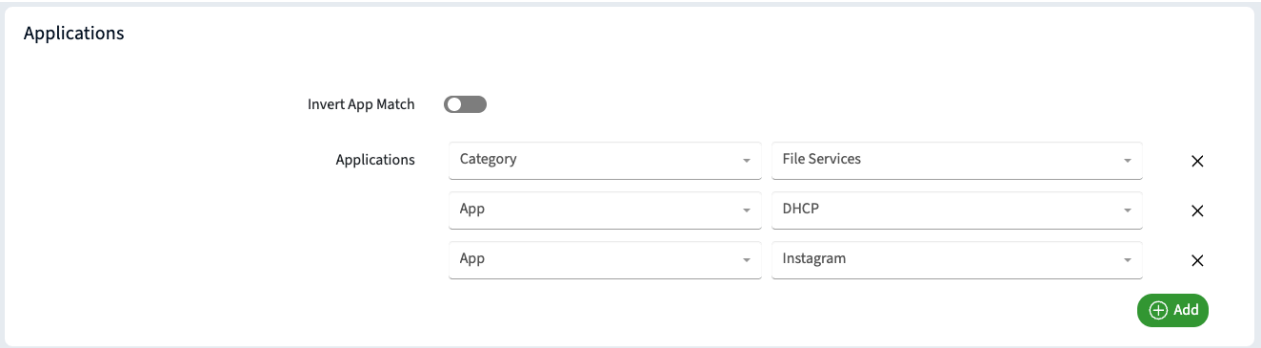


Figure 117: Applications

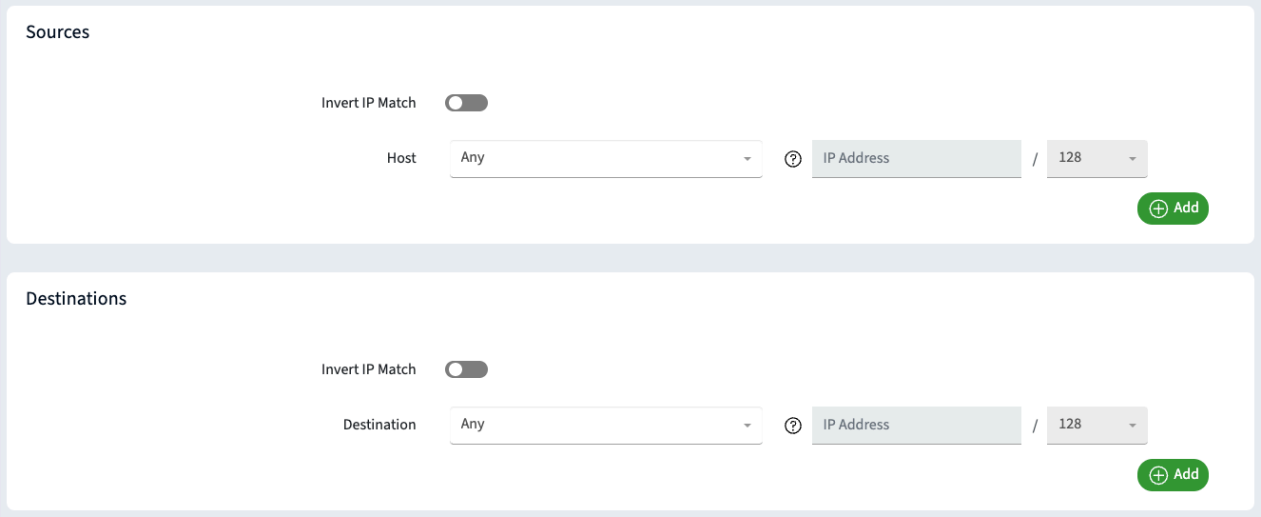


Figure 118: Sources and destinations

Settings

This topic describes how to enable or disable app control functionality and select the network interfaces to analyze.

The settings allow you to enable or disable the app control functionality. Also select which network interfaces the app control system analyzes.

Note: For internet traffic analysis, enable both the input and output interfaces, such as the LAN and WAN interfaces.

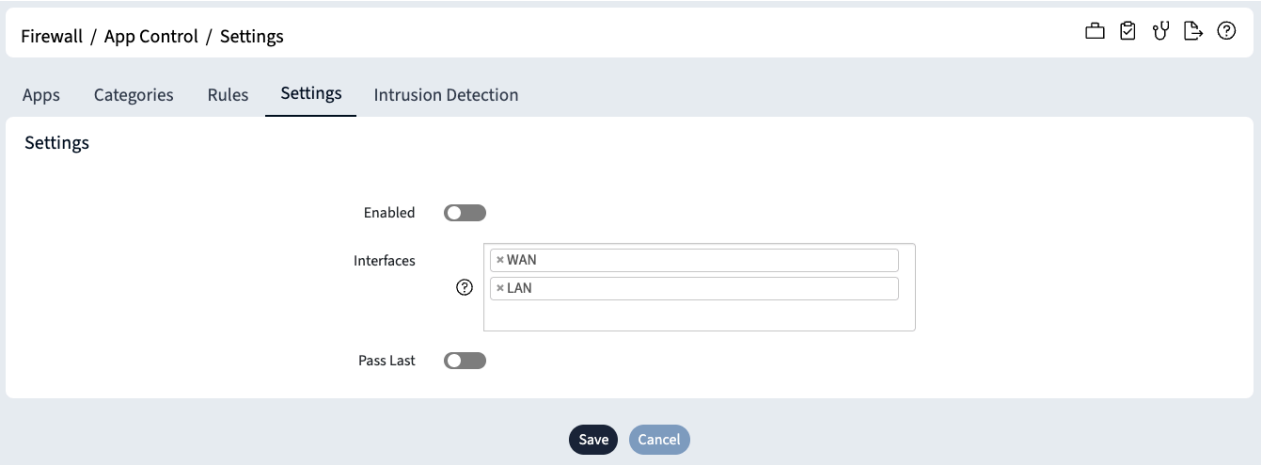


Figure 119: App control settings

DDoS rules

DDoS rules are specialized security filters that identify and match specific IP addresses or packet types to help protect against distributed denial-of-service attacks. The firewall processes these rules at an early stage to maintain performance and help legitimate traffic continue.

Find the DDoS rules under **Firewall > Rules** and select the **DDoS** tab.

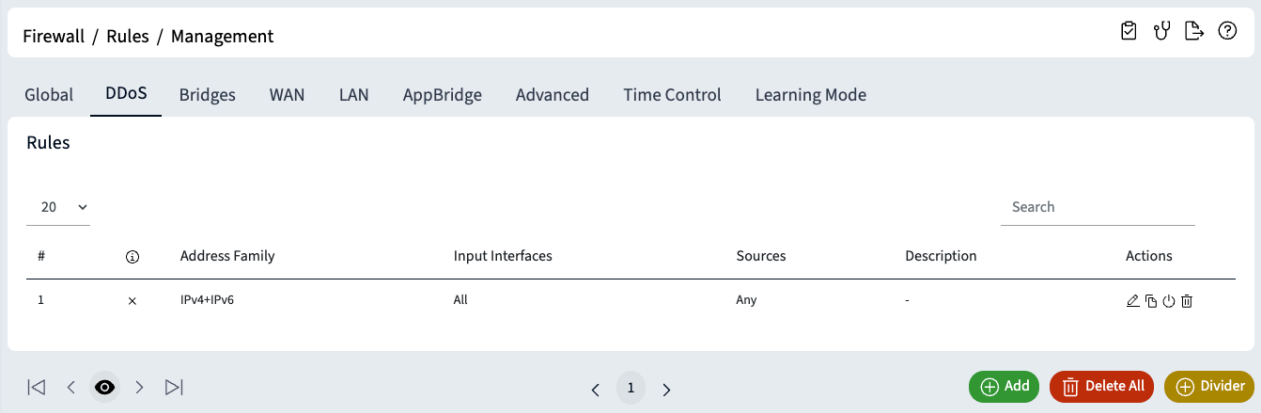


Figure 120: DDOS rules overview

DDoS rules are specialized security filters that identify and match specific IP addresses, both source and destination, or particular packet types. The firewall processes these rules at an early stage when packets arrive, helping provide performance when dropping malicious traffic.

When a DDoS attack occurs, these rules help the system maintain firewall resources and continue processing legitimate traffic.

For information about setting up DDoS rule configurations, see the *general settings* on page 166.

General settings

DDoS rules sharesettings with standard firewall rules. These rules allow you to control traffic processing through settings like status, rule placement, and interfaces. You can choose to drop or accept traffic, specify IP addresses or network ranges, and enable logging for monitoring purposes.

DDoS rules use many of the same settings as standard firewall rules. You can set the following settings for your DDoS protection:

Enabled

This setting enables or disables this rule. When disabled, the rule remains in your configuration but does not process traffic.

Add Rule To

This setting determines where to place this rule in the **Rules** list on the overview page. Rule order affects processing priority.

Input Interfaces

This setting determines which network interface this rule monitors for incoming traffic.

Action

This setting determines the type of action. Select **Drop** to block traffic or **Accept** to allow it. The default action is **Drop**. Use **Accept** to create a whitelist of trusted sources. The system processes `accept` rules first, then `drop` rules.

Description

This field adds a meaningful description to help you identify the purpose of this rule later.

IP Address

This field specifies individual IP addresses or entire network ranges to block. You can enter multiple addresses or networks.

Log this Rule

This setting enables logging to record when this rule blocks or allows traffic. This helps with monitoring and troubleshooting.

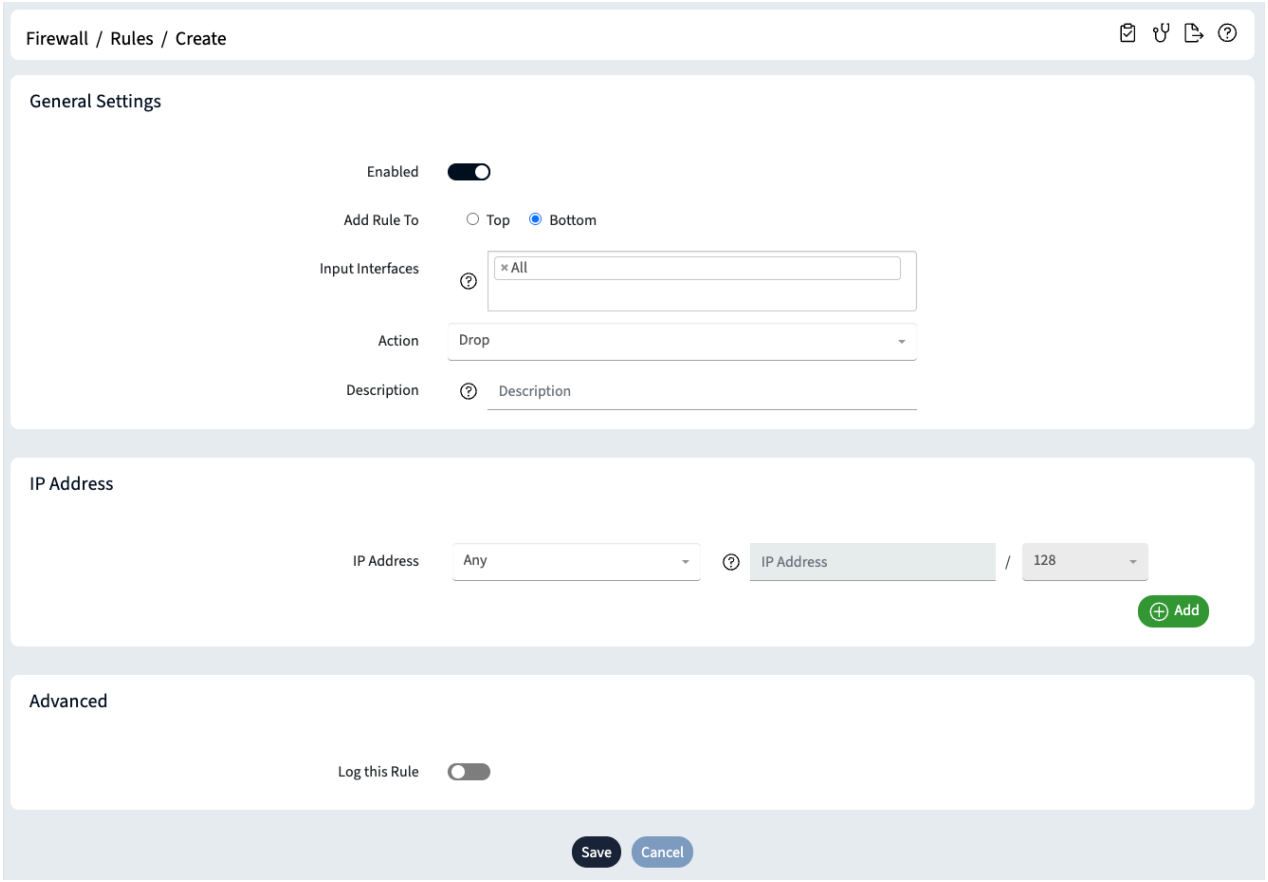


Figure 121: DDOS rule

Create and manage firewall rule dividers

Firewall dividers help you organize and structure firewall rules for better management and clarity.

Dividers act as visual separators that group related firewall rules together. This organization makes it easier to locate specific rules and understand the firewall configuration at a glance.

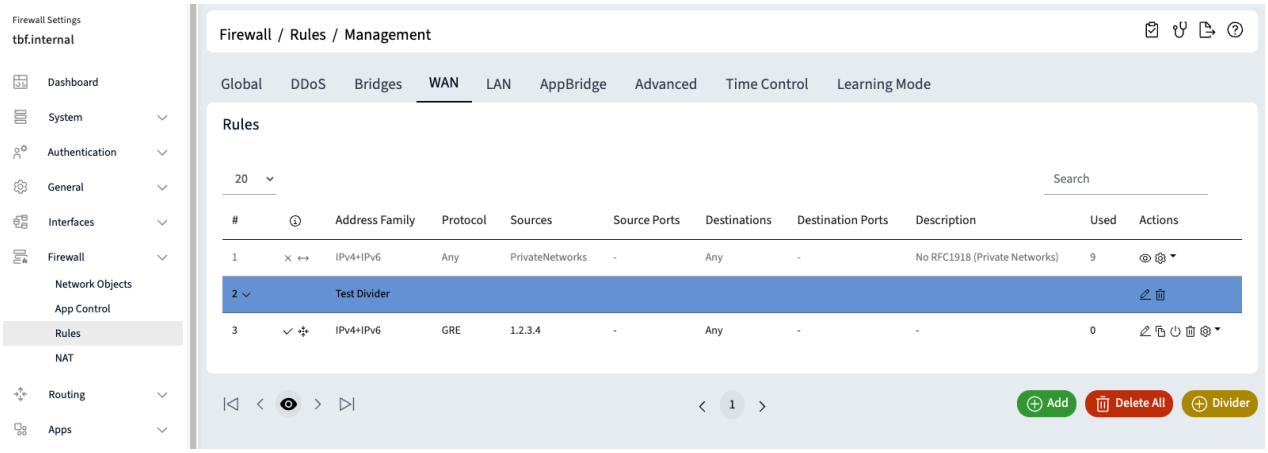


Figure 122: Firewall rules divider

Perform the following steps:

1. Select the **Divider** button in the lower-right corner of a firewall rules overview page.
2. Enter a meaningful **Description** that identifies the rule group.
3. To make the divider visually distinct, select a **Divider Color**.
4. To move dividers to the desired position within the rules list, drag and drop them.
5. To apply the changes, select **Save** at the bottom of the navigation panel.

The divider appears in the firewall rules list and creates a clear visual separation that improves rule organization and management.

In high availability (HA) environments, TBF automatically replicates dividers to the cluster nodes. This helps provide consistent organization across the firewall infrastructure.

Learning mode

Learning mode monitors network traffic to TBF and helps you create firewall rules based on the collected data.

Find the learning mode settings under **Firewall > Rules > Learning Mode**.

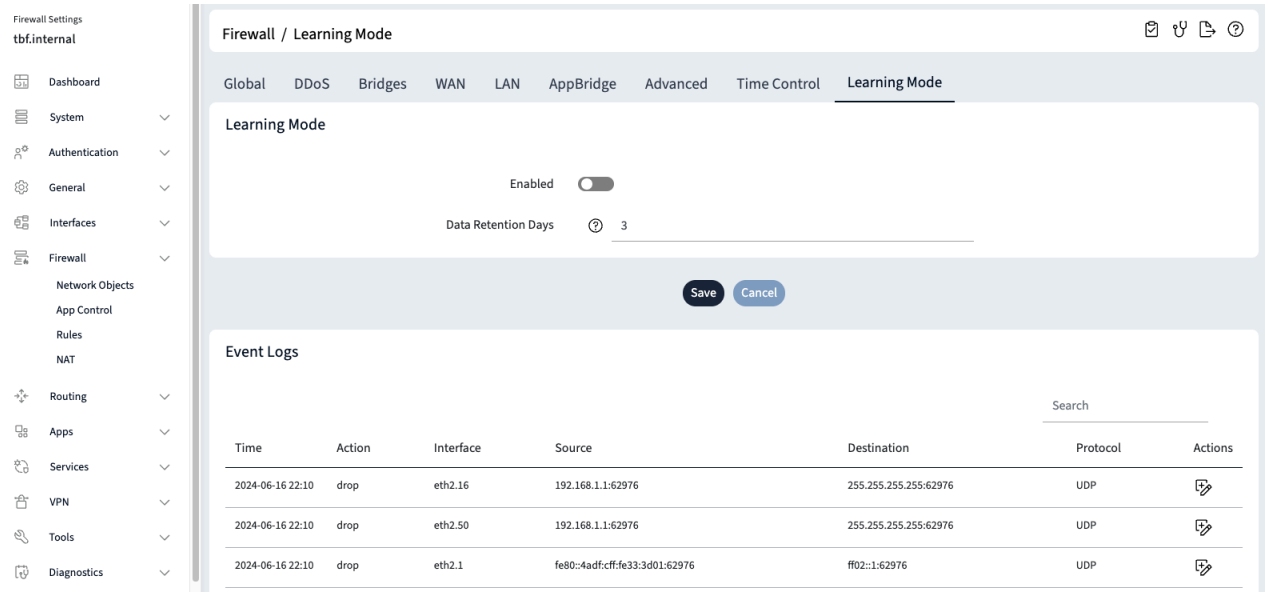


Figure 123: Learning mode

When you enable the learning mode, TBF allows network traffic to pass through while it records detailed information about each connection. TBF logs this traffic data, except for connections that existing firewall rules already control with logging disabled.

Data Retention Days

This setting controls how long TBF keeps logged traffic data for review. A longer retention period allows more time to analyze patterns, but it also uses more disk storage space on the system.

Tip: If the system has limited storage capacity, to help prevent running out of disk space, set this value to a lower number.

To review and search through the collected traffic data, use the **Event Logs** list. If you find traffic that needs a firewall rule, to create a new rule directly from that log entry, use the **Actions** column.

While learning mode runs, a notification icon displays in the upper-right corner of the TBF GUI to show that the system actively monitors traffic.

11. Enforcer

The Enforcer feature has deep packet inspection (DPI) capabilities for operational technology (OT) protocols, which helps protect industrial networks from security threats and protocol violations.

Overview

Enforcer enables DPI of OT protocols. This feature uses Tofino technology for network security for industrial automation systems.

The DPI function examines data packets for suspicious characteristics and protocol violations with inspection of both the header and the payload of each packet. This inspection allows you to monitor and filter data packets, which helps protect the network from unwanted content and potential security threats.

The device automatically blocks any data packets that violate the security profiles you specify. Define security profiles for each supported protocol with the configuration of DPI settings.

Note: When TBF detects an error, set up the device to terminate the data connection automatically.

Supported protocols

Enforcer supports the following OT protocols:

- *AMP* on page 170: ASCII Message Protocol for SCADA and system integration
- *DNP3* on page 176: Distributed Network Protocol for utility communications
- *ENIP* on page 181: EtherNet/IP for industrial automation
- *GOOSE* on page 182: Generic Object Oriented Substation Events
- *IEC104* on page 183: IEC 60870-5-104 telecontrol protocol
- *Modbus* on page 188: Industrial communication protocol
- *OPC Classic* on page 192: OLE for Process Control

Access Enforcer settings

To access the Enforcer settings through the GUI:

1. Navigate to **Firewall > Enforcer**.
2. Select the protocol you want to set up.
3. Set the DPI settings and security profiles.

Related topics

For detailed configuration information, see:

- *Enforcer general* on page 169: Detailed technical information about DPI functionality

Enforcer general

This topic provides detailed technical information about the deep packet inspection (DPI) functionality of the Enforcer feature and supported protocols.

Enforcer has DPI capabilities for operational technology (OT) protocols. The device is built on Tofino technology and supports these industrial communication protocols:

- AMP
- DNP3
- ENIP
- Goose
- IEC104
- Modbus
- OPC Classic

DPI monitors and filters network traffic in real-time and helps protect an industrial network from malicious content and threats. This inspection process helps maintain network integrity and operational reliability.

TBF analyzes each data packet and examines both the packet header and its contents. It looks for suspicious patterns, protocol violations, and anomalies that indicate risks or communication errors.

To define security profiles that match the network requirements, set DPI settings through the GUI. The device automatically blocks packets that violate these security rules and helps prevent potentially harmful traffic from reaching systems.

Note: To help prevent further issues if the system detects a security violation or protocol error, set up the device to automatically terminate the affected connection.

Enforcer rule requirements

When you assign an operational technology (OT) protocol enforcer to a firewall rule, the enforcer restricts which values the rule accepts in the **Source**, **Destination**, and **Destination port** fields.

Overview

You assign an enforcer to a firewall rule with the **Enforcer** setting. For more information, see *firewall rule settings* on page 134.

Each enforcer requires specific firewall rule settings. These requirements verify that deep packet inspection (DPI) targets the correct traffic. The device rejects a rule that does not meet the requirements of the assigned enforcer.

Requirements per protocol

The following table lists the firewall rule requirements for each enforcer. A field marked **Required** must not be set to **Any**.

Enforcer	Source	Destination	Destination port	Other requirements
OPC	Required	Required	Not checked	—
Modbus	Any allowed	Required	Required	The destination port must be `502`.
IEC 60870-5-104	Any allowed	Required	Required	—
DNP3	Any allowed	Required	Required	—
AMP	Any allowed	Required	Required	—
EtherNet/IP (ENIP)	Any allowed	Required	Required	—
GOOSE	Any allowed	Any allowed	Not checked	Do not set a destination MAC address.

Note: The IEC104 and Modbus built-in port objects make the standard ports for these protocols (`2404` and `502`) available. For more information, see *ports* on page 130.

Related topics

- *Enforcer* on page 169: Overview and supported protocols
- *Firewall rule settings* on page 134: The **Enforcer** rule setting

AMP

This page describes the ASCII Message Protocol (AMP) enforcer for industrial automation environments.

The ASCII Message Protocol (AMP) is a communication protocol commonly used in industrial automation for Supervisory Control and Data Acquisition (SCADA) systems and system integration. AMP helps provide communication between industrial equipment to monitor and control automation devices such as programmable logic controllers (PLCs), sensors, and meters.

The device uses deep packet inspection (DPI) to examine and filter data packets. When packets violate the rules in the defined profiles, the device discards them. The AMP enforcer function supports both the Common ASCII Message Protocol (CAMP) and the Non-Intelligent Terminal Protocol (NITP) over TCP connections. The device applies DPI to CAMP and NITP data streams based on the program and mode protection settings and the profiles you create.

When you enable an AMP enforcer profile, the device applies the rules of that profile to the incoming data stream.

The device allows only data packets that contain values you specify in these fields. The available fields depend on the program and mode protection settings:

- **Protocol**
- **Message Types**
- **Address Classes**
- **Device Classes**
- **Memory Addresses**
- **Data Words**
- **Task Codes**
- **Task Code Data**
- **Error check characters**
- **Block check characters**
- **Sanity Check**

The AMP enforcer interface includes these tabs:

- **AMP**
- **Task Codes**
- **Global Settings**

Find the AMP enforcer under **Firewall > Enforcer > AMP**.

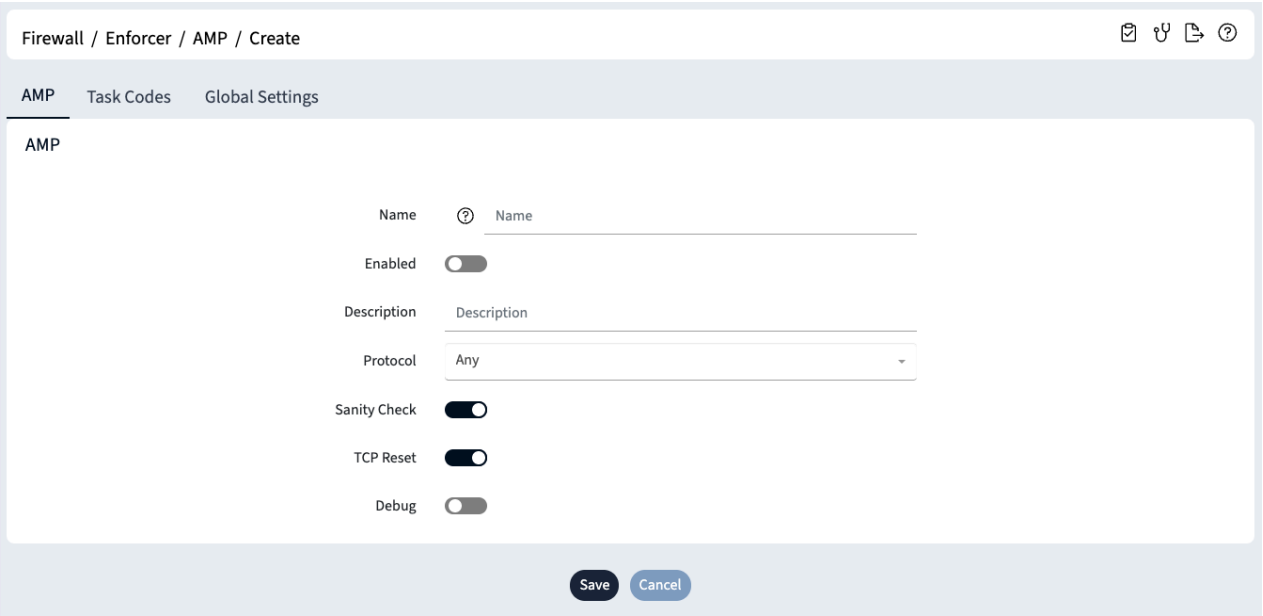


Figure 124: AMP

Related topics

- *Enforcer rule requirements* on page 170: Firewall rule requirements for each enforcer

AMP settings

This page describes the AMP enforcer rule configuration settings.

Name

This field specifies a name for the AMP enforcer. Use 0..100 characters.

Description

This field specifies a description for the AMP enforcer. Use 0..250 characters.

Enabled

This setting enables or disables the AMP enforcer.

Protocol

This setting specifies the TCP payload protocol type for data packets that use this profile. The device applies the profile only to packets that match the protocol you select.

- **Any**: This option applies the profile to data packets regardless of protocol. This is the default.
- **CAMP (Common ASCII Message Protocol)**
- **NITP (Non-Intelligent Terminal Protocol)**

Task Codes

This field becomes available when you select one of the following protocols:

- **Any**: Also select **Any** in the **Message Types** field.
- **CAMP (Common ASCII Message Protocol)**: Also enter a hexadecimal value from `00..03` or `FF` in the **Message Types** field.
- **NITP (Non-Intelligent Terminal Protocol)**

The following values are available:

- `01..9A`

Task Code Data

This field specifies the task code data for the selected task code. This field becomes available when you select one of the following protocols:

- **CAMP (Common ASCII Message Protocol)**: Also enter a hexadecimal value from `00..03` or `FF` in the **Message Type** field and a single hexadecimal value in the **Task Codes** field.
- **NITP (Non-Intelligent Terminal Protocol)**: Also enter a single hexadecimal value in the **Task Codes** field.

The following values are available:

- `0..F`: The device applies the profile only to data packets that contain the specified task code data. The maximum length is 72 bytes.

Message Types

This setting determines whether the message is a command type or a response type. This setting becomes available when you select **CAMP (Common ASCII Message Protocol)** as **Protocol**.

The following values are available:

- `00..FF`

Address Classes

This setting specifies the type of memory to access on the equipment. This setting becomes available when you:

- Select **CAMP (Common ASCII Message Protocol)** as **Protocol**.
- Enter a hexadecimal value from `00..03` or `FF` in the **Message Types** field.

The following values are available:

- `any`: The device applies the profile to data packets regardless of address class. This is the default.
- `0000..FFFF`: The device applies the profile only to data packets that contain the specified address class.

Device Classes

This setting specifies the type of device class (vendor-specific device) to access. This setting becomes available when you:

- Select **CAMP (Common ASCII Message Protocol)** as **Protocol**.
- Enter a hexadecimal value from `00..03` or `FF` in the **Message Types** field.

The following values are available:

- `any`: The device applies the profile to data packets regardless of device class. This is the default.
- `0000..FFFF`: The device applies the profile only to data packets that contain the specified device class.

Memory Addresses

The setting specifies the starting address of the memory to read or write. This setting becomes available when you:

- Select **CAMP (Common ASCII Message Protocol)** as **Protocol**.
- Enter a hexadecimal value from `00..01` or `04..09` or `FF` in the **Message Types** field.

The following values are available:

- `any`: The device applies the profile to data packets regardless of memory address. This is the default.
- `0000..FFFF`: The device applies the profile only to data packets that contain the specified memory address.

Data Words

This setting specifies the starting address that the device uses to read data from the packet. This setting becomes available when you:

- Select **CAMP (Common ASCII Message Protocol)** as **Protocol**.
- Enter a hexadecimal value from `00..01` or `08..09` or `FF` in the **Message Types** field.

The following values are available:

- `any`: The device applies the profile to data packets regardless of data word. This is the default.
- `0000..FFFF`: The device applies the profile only to data packets that contain the specified data word.

Sanity Check

This setting enables or disables plausibility checking for data packets.

- **Enabled**: This option enables plausibility checking. The device verifies that data packets meet format and specification requirements. The device blocks packets that violate the specified profiles. This is the default.
- **Disabled**: This option disables plausibility checking.

TCP Reset

This setting specifies whether to reset the TCP connection when a protocol violation occurs or when the plausibility check detects an error.

- **Enabled**: This option enables the TCP connection reset. When the device identifies a protocol violation or detects a plausibility check error, it terminates the TCP connection. The device re-establishes the connection when it receives a new connection request. This is the default.
- **Disabled**: This option disables the TCP connection reset.

Debug

This setting enables or disables debugging for profiles.

- **Enabled**: This option enables debugging. The device sends the reset packet along with information on the TCP connection termination. To make this option work, also enable **TCP Reset**.
- **Disabled**: This option disables debugging. This is the default.

Task codes settings

This page describes the task code configuration settings.

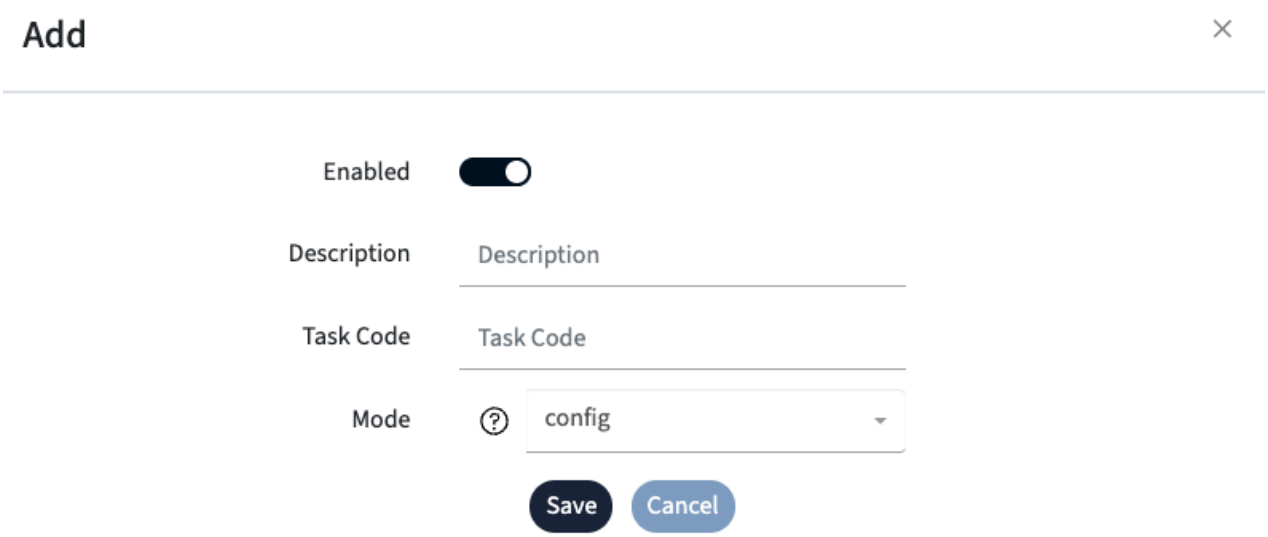


Figure 125: AMP task codes

Enabled

This setting controls whether the AMP enforcer is active. When you enable the AMP enforcer, it monitors and controls task code operations.

Description

This setting provides a text description for the AMP enforcer. To identify or explain the purpose of this enforcer instance, you can enter up to 250 characters.

Task Code

This setting specifies the task code value. Valid values range from `01..9A`.

Mode

This setting defines the operational mode for the task code:

- **config**: This mode applies to commands that modify controller settings, application programs, or operational modes. When the task code involves system configuration changes, use this mode.
- **non-config**: This mode applies to standard `read` and `write` commands that do not modify controller settings, application programs, or operational modes. For routine data operations, use this mode.

AMP global settings

This page describes the global configuration settings for the AMP enforcer.

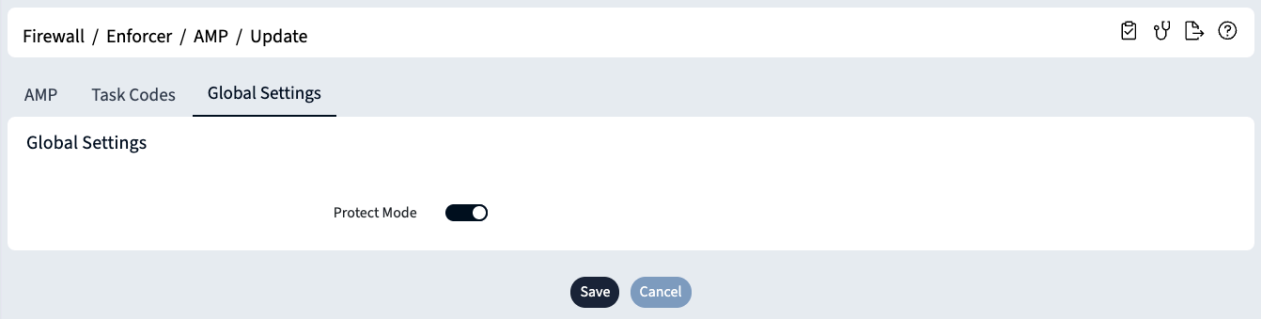


Figure 126: AMP global settings

Protect Mode

This setting controls whether the device inspects data packets that contain task codes with **config** values in the **Mode** field.

Enabled

This is the default setting that enables packet inspection. The device forwards only data packets that match the profile parameters. It blocks data packets that contain **config** values in the **Mode** field for task codes specified in the profiles.

Disabled

This setting disables packet inspection. The device forwards data packets that match the profile parameters, including those with task codes that contain **config** values in the **Mode** field.

Task code reference

This page lists the available task codes and their functions.

#	Meaning
01	Read Word Memory Random
02	Write Word Memory Area Random
30	Read Operational Status
32	Program to Run Mode
33	Go to Program Mode
34	Execute Power-up
35	Execute Complete (Warm) Start
36	Execute Partial (Hot) Start
50	Read User Word Area Block
51	Write User Word Area Starting at Address
58	Set Controller Time of Day Clock
59	Write Discrete I/O Status or Force via Data Element Type
5A	Write Block
6B	Read Discrete I/O Status or Force via Data Element Type
71	Read Controller Time of Day Clock
7D	Read SF/Loop Processor Mode
7E	Read Random
7F	Read Block
88	Select Number of SF Module Task Codes Per Scan
89	Read Number of SF Module Task Codes Per Scan
99	Write VME Memory Area Block/Random
9A	Read VME Memory Area Block/Random

AMP message types reference

This page lists the AMP message types and their identifiers.

#	Meaning
00	Module General Query Command
01	Module General Response Command
02	Packet T/C Command
03	Packed T/C Response
04	Read data Command
05	Read data Response
06	Write data Command
07	Write data Response
08	Mem Exch Command
09	Mem Exch Response
FF	Protocol Error

DNP3

The DNP3 protocol helps provide communication between components in process automation systems and offers multiplexing, error checking, link control, prioritization, and layer 2 addressing services for user data.

The DNP3 enforcer function activates the deep packet inspection (DPI) firewall capabilities for the DNP3 data stream. The device blocks data packets that violate the specified settings. If the user requests it, the device verifies data packets for their plausibility and fragment characteristics. The device verifies and monitors DNP3 data connections and helps protect against invalid or falsified data packets.

When you enable the DNP3 enforcer settings, the device applies the settings to the data stream:

- The device permits data packets that contain only the function codes that you specify in **Function Codes**.
- The device rejects data packets that contain any other function code that you do not specify in **Function Codes**.

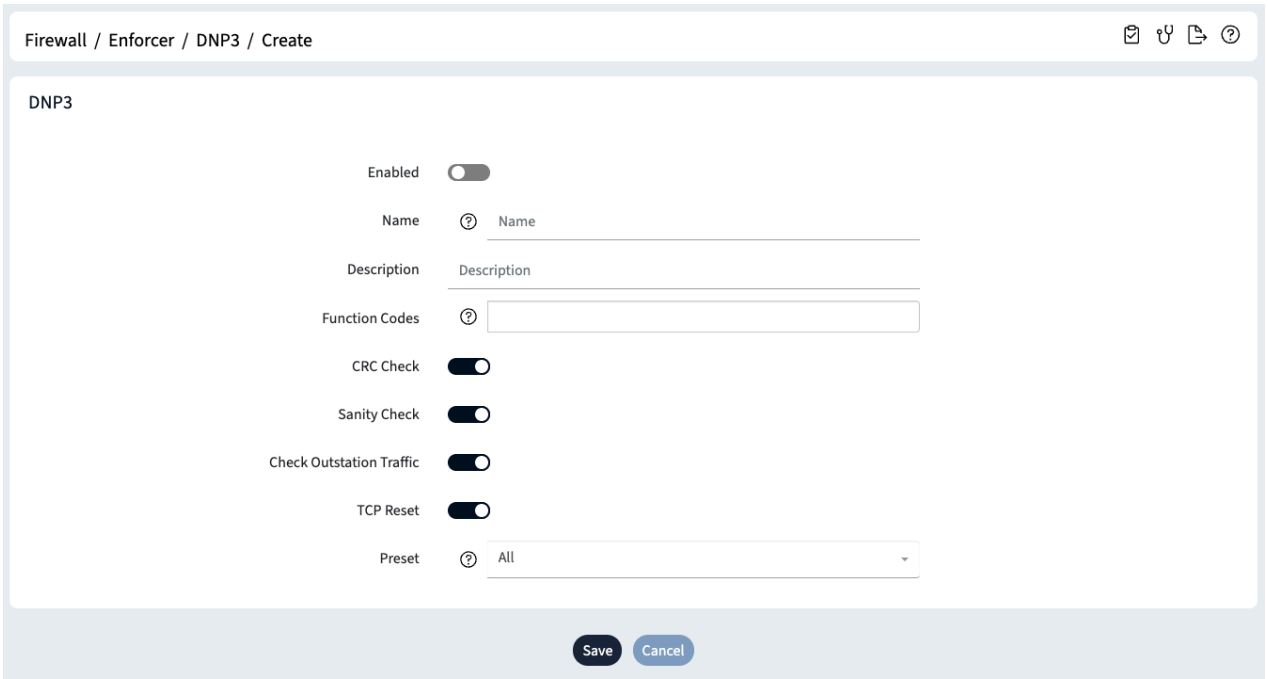


Figure 127: DNP3

Related topics

To view detailed configuration information, see the following topics:

- *DNP3 settings* on page 177: These are configuration parameters for the DNP3 enforcer function.
- *DNP3 objects* on page 178: These are object configuration parameters and definitions.
- *DNP3 function codes* on page 179: This is a reference table of supported function codes.
- *DNP3 default object list* on page 180: This is a reference table of default DNP3 objects.
- *Enforcer rule requirements* on page 170: These are the firewall rule requirements for each enforcer.

DNP3 settings

This topic describes configuration settings for the DNP3 enforcer function.

Enabled

This setting determines whether the DNP3 enforcer is active.

- **Enabled**
- **Disabled** (default)

Name

This field specifies the name of the DNP3 enforcer. Enter a string with 0..100 characters.

Description

This field provides the description of the DNP3 enforcer. Enter a string with 0..250 characters.

Function Codes

This setting displays the function codes for the DNP3 enforcer. The device permits data packets with the specified properties. You can specify multiple function codes.

- `0..255`

CRC Check

This setting enables or disables the CRC check for data packets to validate the checksum that the DNP3 data packets contain.

- **Enabled** (default): This option enables the CRC check. The device calculates the checksum and compares it with the checksum field in the DNP3 data packets.
- **Disabled**: This option disables the CRC check.

Sanity Check

This setting enables or disables the plausibility check for data packets.

- **Enabled** (default): This option enables the plausibility check. The device checks the plausibility of data packets regarding format and specification. The device blocks data packets that violate the specified settings.
- **Disabled**: This option disables the plausibility check.

Check Outstation Traffic

This setting enables or disables the checking of data packets that an outstation originates.

- **Enabled** (default): This option enables the checking of data packets from an outstation.
- **Disabled**: This option disables the checking of data packets from an outstation.

TCP Reset

This setting enables or disables the resetting of the TCP connection in the case of a protocol violation or if the plausibility check has a detected error.

- **Enabled** (default): This option enables the resetting of the TCP connection. If the device identifies a protocol violation or has a detected plausibility check error, the device terminates the TCP connection. The device establishes the TCP connection again upon receiving a new request.
- **Disabled**: This option disables the resetting of the TCP connection.

Preset

This setting determines the preset of default DNP3 objects.

- **All** (default): This option assigns default objects.
- **Custom**: This option lets you specify user-defined objects.

DNP3 objects

This topic describes settings for DNP3 object definitions.

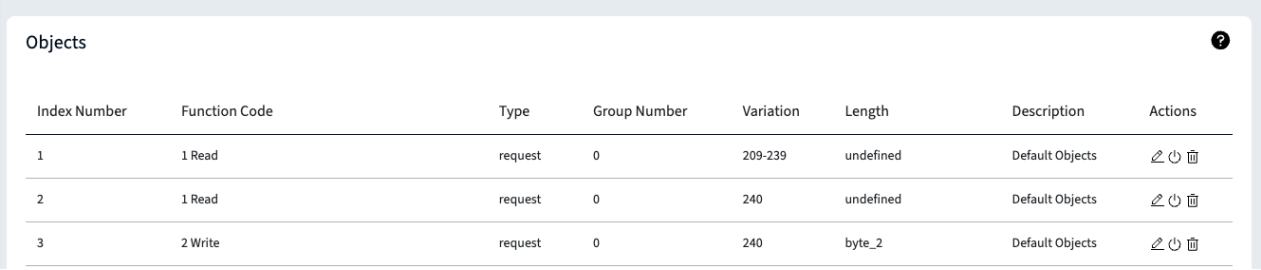


Figure 128: DNP3 objects

Description

This setting contains the description of the DNP3 object. Enter a string with 0..250 characters.

Type

This setting specifies the type of the message.

- **Request**: This option creates a request message object in the object list.
- **Response**: This option creates a response message object in the object list.

Group Number

This setting specifies a means of classifying the type or types of data packets in a message. Specify a valid value in the **Type** field.

- `0..255`: Each group number shares a common point type and method of data packet creation. The point type defines the machine in an outstation.

Variation

This setting specifies the variation number. Specify a valid value in the **Group Number** field. The device applies the DNP3 enforcer profile only to data packets that contain the specified value.

The DNP3 function provides the choice of encoding formats for the type of data packets known as *variation number*. Every value in the **Group Number** field has a set of variation numbers.

- `0..255`: This setting allows you to specify the following options:
  - You can specify a single variation number with a single numerical value, for example `1`.
  - You can specify a range with numerical values connected by a dash, for example `0-55`.

Function Code

This setting identifies the purpose of the message. Specify a valid value in the **Variation** field. The device applies the DNP3 enforcer profile only to data packets that contain the specified value.

- **0..128**: These values represent request messages from masters. Specify a single numerical value, for example **1**.
- **129..255**: These values represent response messages from outstations. Specify a single numerical value, for example **254**.

Qualifier

This setting specifies the qualifier code for a pair of each **Group Number**, **Variation**, and **Function** field. The qualifier code is an 8-bit value that defines the prefix code and the range specifier code for the object in a DNP3 message.

Specify a valid value in the **Function** field. The device applies the DNP3 enforcer profile only to data packets that contain the specified value.

- `0x00..0xff`: This setting specifies multiple individual qualifier codes using hexadecimal values separated by a comma for a set of each **Group Number**, **Variation**, and **Function** field.

Length

This setting specifies the optional length for the object. Specify a valid value in the **Function** field. The device applies the DNP3 enforcer profile only to data packets that contain the specified value.

- **0..255**: This setting specifies a single numerical value, for example 1.
- **byte_2**: The second byte of the object data contains the length of the remaining portion of the data.
- **single_bit_packed**: When the count of bit values is not a multiple of 8, the device pads the packed single-bit values up to the next byte boundary.
- **double_bit_packed**: When the count of double bit values is not a multiple of 4, the device pads the packed double bit values up to the next byte boundary.
- **variation**: This option encodes the length of the object.

DNP3 function codes

This topic provides a reference table of DNP3 function codes with descriptions and operational details.

The following table lists the DNP3 function codes that the Enforcer supports:

Code	Name	Description
0	CONFIRM	Confirms receipt of a message that requested confirmation
1	READ	Requests data from the outstation
2	WRITE	Writes data to the outstation
3	SELECT	Selects control points in preparation for a subsequent operate command
4	OPERATE	Operates previously selected control points
5	DIRECT_OPERATE	Directly operates control points without prior selection
6	DIRECT_OPERATE_NR	Directly operates control points without prior selection and without response
7	IMMED_FREEZE	Immediately freezes counter values
8	IMMED_FREEZE_NR	Immediately freezes counter values without response
9	FREEZE_CLEAR	Freezes and clears counter values
10	FREEZE_CLEAR_NR	Freezes and clears counter values without response
11	FREEZE_AT_TIME	Freezes counter values at a specified time
12	FREEZE_AT_TIME_NR	Freezes counter values at a specified time without response
13	COLD_RESTART	Performs a cold restart of the outstation
14	WARM_RESTART	Performs a warm restart of the outstation
15	INITIALIZE_DATA	Initializes data to default values
16	INITIALIZE_APPL	Initializes application-specific data
17	START_APPL	Starts application processing
18	STOP_APPL	Stops application processing
19	SAVE_CONFIG	Saves current configuration to non-volatile memory
20	ENABLE_UNSOLICITED	Enables unsolicited responses

Code	Name	Description
21	DISABLE_UNSOLICITED	Disables unsolicited responses
22	ASSIGN_CLASS	Assigns data points to reporting classes
23	DELAY_MEASURE	Measures communication delay
24	RECORD_CURRENT_TIME	Records the current time
25	OPEN_FILE	Opens a file for reading or writing
26	CLOSE_FILE	Closes an open file
27	DELETE_FILE	Deletes a file
28	GET_FILE_INFO	Gets information about a file
29	AUTHENTICATE_FILE	Authenticates a file
30	ABORT_FILE	Aborts a file operation
129	RESPONSE	Response to a request
130	UNSOLICITED_RESPONSE	Unsolicited response from outstation
131	AUTHENTICATE_RESP	Response to authentication challenge

DNP3 default object list

This topic provides a reference table of the default DNP3 objects with request and response message specifications.

Request messages

The following table shows the default DNP3 request message objects. Each entry includes the message index, group number, variation range, function code, function name, data length, and qualifier values.

Index	Group number	Variation	Function	Function name	Length	Qualifier
1	0	209-239	1	Read	undefined	`0x00`
2	0	240	1	Read	undefined	`0x00`
3	0	240	2	Write	`byte_2`	`0x00`
4	0	241-243	1	Read	undefined	`0x00`
5	0	245-247	1	Read	undefined	`0x00`
6	0	245-247	2	Write	`byte_2`	`0x00`
7	0	248-250	1	Read	undefined	`0x00`
8	0	252	1	Read	undefined	`0x00`
9	0	254	1	Read	undefined	`0x00,0x06`
10	0	255	1	Read	undefined	`0x00,0x06`

Note: This table shows the first 10 entries from a total of 138 request message entries. For the complete table, refer to the full DNP3 specification documentation.

Response messages

The following table shows the default DNP3 response message objects. Response messages use the function code `129` and include specific qualifier values for data handling.

Index	Group number	Variation	Function	Function name	Length	Qualifier
139	0	209-239	129	Response	`byte_2`	`0x00,0x17`
140	0	240	129	Response	`byte_2`	`0x00,0x17`
141	0	241-243	129	Response	`byte_2`	`0x00,0x17`
142	0	245-247	129	Response	`byte_2`	`0x00,0x17`
143	0	248-250	129	Response	`byte_2`	`0x00,0x17`

Note: This table shows the first 5 entries from a total of 179 response message entries. For the complete table, see the full DNP3 specification documentation.

ENIP

This topic describes EtherNet/IP (ENIP) protocol support and setup for industrial automation applications.

ENIP is a communication protocol that industrial automation systems use. ENIP combines standard Ethernet and Internet Protocol (IP) technologies to deliver Common Industrial Protocol (CIP) data packets across networks.

Overview

ENIP delivers real-time control and information solutions for manufacturing automation environments. ENIP handles 2 main types of communication:

- Real-time I/O messaging for immediate control responses
- Explicit messaging for device setup and monitoring tasks

Key features also include:

- Compatibility with various industrial device types
- Integration with existing Ethernet network infrastructure

Security considerations

To help protect the system when you deploy ENIP in industrial networks, implement these security measures:

- To separate ENIP traffic from other network communications, use network segmentation.
- To limit which devices communicate with each other, apply access control lists.
- Keep ENIP-enabled devices updated with the current security patches.
- To track ENIP communication patterns, set up monitoring and logging systems.

Related information
ENIP settings on page 181
ENIP class IDs on page 182
Enforcer rule requirements on page 170

ENIP settings

This topic describes settings that control how the ENIP protocol handles communication and applies security policies.

Settings

To adjust communication behavior and security policies for the network environment, use these settings.

Port

This field specifies the TCP port number that the system uses for ENIP communication. The default is `44818`.

Timeout

This field specifies the time in seconds before a connection attempt times out. The default is `30`.

Max connections

This field specifies the maximum number of simultaneous connections the system allows. The default is `100`.

Buffer size

This field specifies the size in bytes that the system allocates for storing incoming and outgoing messages. The default is `1024`.

Enable logging

This field records detailed information on protocol activities for troubleshooting. The default is `false`.

ENIP class IDs

This topic describes class IDs that identify different types of objects and services within the ENIP Common Industrial Protocol (CIP) protocol framework.

ENIP uses class IDs as unique identifiers for different types of objects and services within the CIP framework. Each class ID represents a specific function or data type that devices use to communicate effectively.

Class ID	Object Type	Primary Function
0x01	Identity Object	Provides device identification and basic information
0x02	Message Router	Routes messages between different objects and services
0x04	Assembly Object	Groups I/O data for efficient transfer between devices
0x06	Connection Manager	Establishes and maintains network connections
0x0F	Parameter Object	Stores and manages device configuration parameters
0x43	Analog Input Point	Handles continuous analog input signals, such as voltage, current, and temperature
0x44	Analog Output Point	Controls continuous analog output signals
0x45	Digital Input Point	Processes discrete digital input signals, such as on/off and true/false
0x46	Digital Output Point	Controls discrete digital output signals

Important: Class IDs are hexadecimal values that must exactly match the device specifications to help provide proper communication. Incorrect class IDs cause communication to be unsuccessful.

GOOSE

The Generic Object Oriented Substation Event (GOOSE) Loadable Security Module (LSM) has deep packet inspection (DPI) firewall capabilities specifically designed for GOOSE traffic within IEC 61850 protocol communications.

The GOOSE LSM has DPI firewall capabilities specifically designed for GOOSE traffic. This module acts as a specialized security layer that examines and filters GOOSE communications to help protect the substation network.

GOOSE traffic is a communication standard within the IEC 61850 protocol suite. This protocol enables intelligent electronic devices (IEDs) in electrical substation automation systems to exchange specific information quickly. GOOSE messages typically carry status updates, alarms, and control commands between devices.

Find the GOOSE settings under **Firewall > Enforcer > GOOSE**.

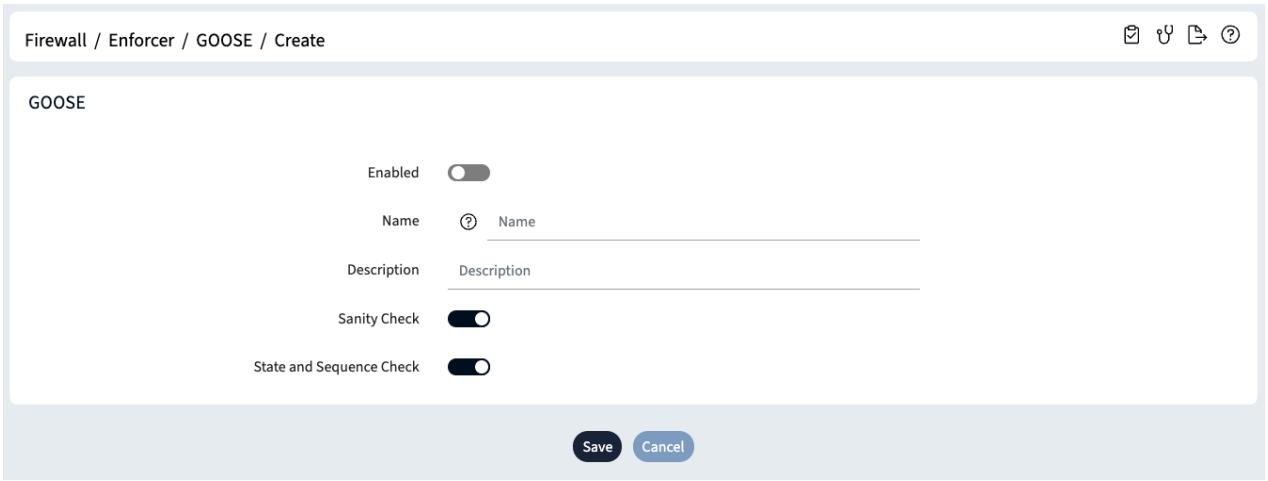


Figure 129: GOOSE

For detailed configuration options, see *GOOSE settings* on page 183.

Related topics

- *Enforcer rule requirements* on page 170: Firewall rule requirements for each enforcer

GOOSE settings

The GOOSE settings control the GOOSE enforcer functionality, including activation status, identification fields, and security validation features. Key settings include enabling the enforcer, defining name and description fields, and controlling sanity checks and state and sequence verification for data packet validation.

Enabled

This setting controls whether the GOOSE enforcer is active.

- **Enabled**
- **Disabled** (default)

Name

This field specifies the name of the GOOSE enforcer. Enter a string with 0..100 characters.

Description

This field provides the description of the GOOSE enforcer. Enter a string with 0..250 characters.

Sanity Check

This setting enables or disables the plausibility check for data packets.

- **Enabled** (default): Enables the plausibility check. The device verifies that data packets conform to the correct format and specifications. The device blocks data packets that violate the specified profiles.
- **Disabled**: Disables the plausibility check.

State and Sequence Check

This setting enables or disables the state and sequence check functionality.

- **Enabled** (default)
- **Disabled**

IEC104

The IEC104 protocol is a communication standard used in industrial automation systems to exchange data between control centers and remote substations over TCP/IP networks.

IEC104 transfers data packets between a control station (client) and a substation (server) over TCP/IP networks. The IEC104 enforcer function helps provide deep packet inspection (DPI) firewall protection for IEC104 data streams. Type IDs within the protocol identify the specific purpose of each data transfer. The device blocks any data packets that do not match the security profiles.

When you activate an IEC104 enforcer profile, the device automatically applies that profile to monitor and filter the data stream.

The device allows only data packets that contain values you specify in these settings:

- Advanced **Type IDs** list
- **Originator Address List**
- **Common Address List**

Find the IEC104 enforcer settings under **Firewall > Enforcer > IEC104**.

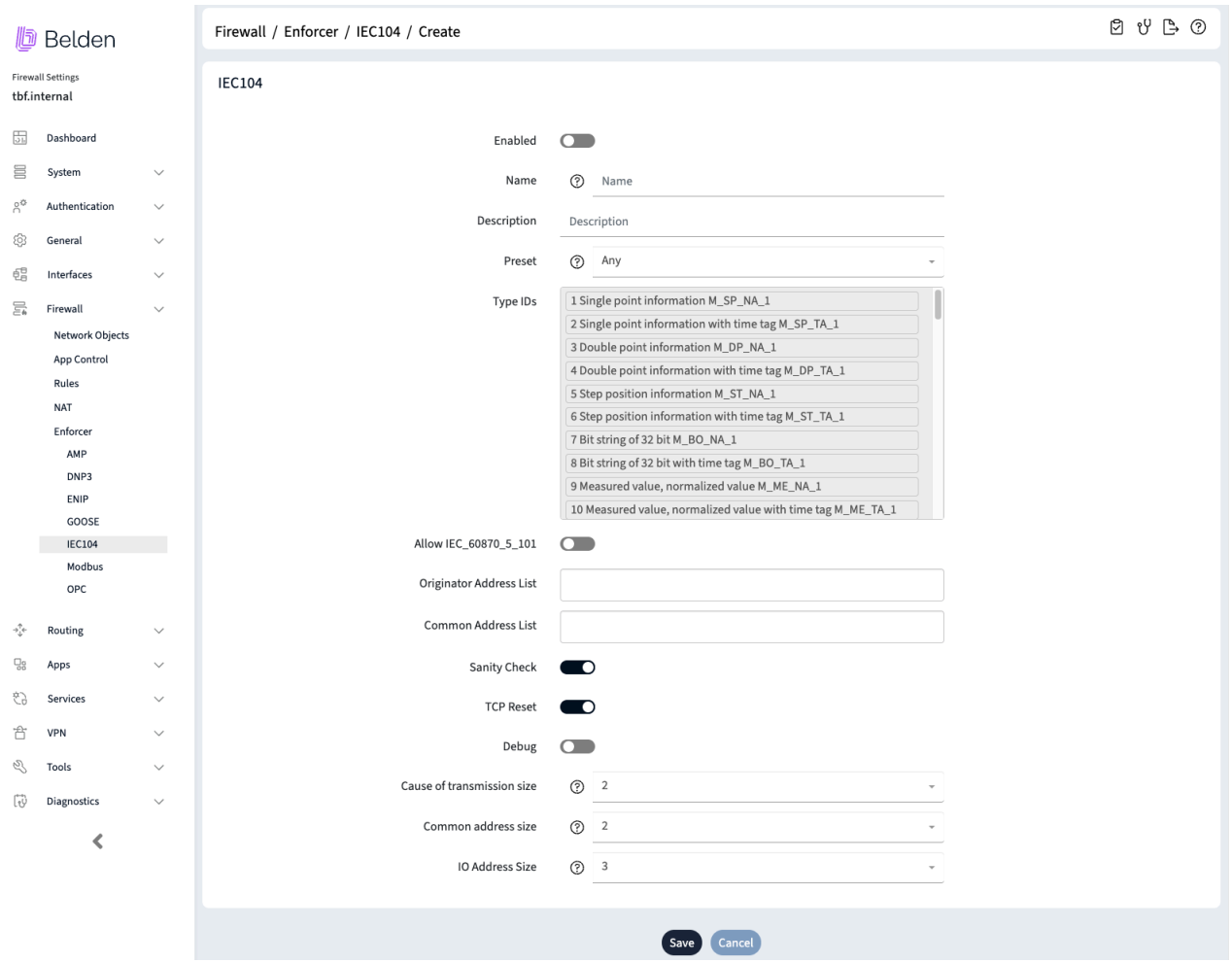


Figure 130: IEC104

Related topics

- *IEC104 settings* on page 184: Configuration parameters for the IEC104 enforcer
- *IEC104 type ID meanings* on page 186: Reference table of type ID values and their meanings
- *Enforcer rule requirements* on page 170: Firewall rule requirements for each enforcer

IEC104 settings

This topic describes the settings for the IEC104 enforcer function.

Enabled

This setting controls whether the IEC104 enforcer is active.

- **Enabled**
- **Disabled** (default)

Name

This field specifies the name of the IEC104 enforcer. Enter a string with 0..100 characters.

Description

This field specifies the description of the IEC104 enforcer. Enter a string with 0..250 characters.

Preset

This setting determines the preset configuration for type IDs.

- **Read Only**: This preset assigns type IDs for `read` functions only.

  `1,3,5,7,9,11,13,15,20,21,30-40,70,100-102`

- **Read Write**: This preset assigns type IDs for both `read` and `write` functions.

  `1,3,5,7,9,11,13,15,20,21,30-40,45-51,58-64,70,100-102`

- **Common**: This preset assigns type IDs for standard functions.

  `1,3,5,7,9,11,13,15,20,21,30-40,45-51,58-64,70,100-102,110-113,120-127`

- **Any** (default): This preset assigns type IDs for all functions.

  `1,2,..,254,255`

  The device helps prevent any subsequent changes to the advanced **Type IDs** list.

- **Advanced**: This preset allows you to specify custom values in the **Type IDs** list.

Type IDs

The system automatically populates this field based on the preset selection. If the preset is **Advanced**, select custom type IDs.

Allow IEC_60870_5_101

This setting controls whether type IDs defined by the IEC101 specification are allowed.

- **Enabled**: This option allows type IDs defined by the IEC101 specification. The device permits type ID values `2,4,6,8,10,12,14,16,17,18,19,103,104,105,106` along with the type IDs based on the function type field or advanced **Type IDs** list settings.
- **Disabled** (default): This option blocks type IDs defined by the IEC101 specification. The device permits only the type ID values based on the advanced **Type IDs** list settings.

Originator Address List

This list specifies the addresses from which data packets originate. To use this feature, set the **Cause of transmission size** field to **2**.

- `<empty>` (default): The device accepts data packets from any originator address.
- `0..255`: The device accepts data packets only from the originator address you specify.

Common Address List

This list specifies the addresses to which the device forwards IEC104 data packets.

- `0..255`: The device accepts data packets with the common address you specify. To use this range, set the **Common address size** to **1**.
- `0..65535` (2^{16}-1): The device accepts data packets with the common address you specify. To use this range, set the **Common address size** to **2**.

Sanity Check

This setting controls whether the device performs plausibility checks on data packets.

- **Enabled** (default): This option enables plausibility checking. The device verifies that data packets comply with format and specification requirements. The device blocks data packets that violate the profiles you specify.
- **Disabled**: This option disables plausibility checking.

TCP Reset

This setting controls whether the device resets the TCP connection when a protocol violation occurs or when the plausibility check detects an error.

- **Enabled** (default): This option enables TCP connection reset. If the device identifies a protocol violation or detects a plausibility check error, it terminates the TCP connection. The device re-establishes the TCP connection when it receives a new request.
- **Disabled**: This option disables TCP connection reset.

Debug

This setting controls whether the device provides debugging information for profiles.

- **Enabled**: This option enables debugging. The device sends the reset packet along with information related to the TCP connection termination. To use this feature, enable **TCP Reset**.
- **Disabled** (default): This option disables debugging.

Cause of transmission size

This setting specifies the size in octets that defines the variation of the respective fields in data packets. The device performs the DPI function based on these settings.

- **1**: The data packets do not contain an originator address.
- **2** (default): The data packets contain an originator address.

Common address size

This setting specifies the size in octets of the common address to which the device forwards IEC104 data packets. This setting affects the **Common Address List** field.

- **1**
- **2** (default)

IO Address Size

This setting specifies the size in octets of the information object address.

- **1**
- **2**
- **3** (default)

IEC104 type ID meanings

This topic provides a reference table that shows the meaning of IEC104 type ID values and their protocol identifiers.

#	Meaning
1	Single point information `M_SP_NA_1`
2	Single point information with time tag `M_SP_TA_1`
3	Double point information `M_DP_NA_1`
4	Double point information with time tag `M_DP_TA_1`
5	Step position information `M_ST_NA_1`
6	Step position information with time tag `M_ST_TA_1`
7	Bit string of 32 bit `M_BO_NA_1`
8	Bit string of 32 bit with time tag `M_BO_TA_1`
9	Measured value, normalized value `M_ME_NA_1`
10	Measured value, normalized value with time tag `M_ME_TA_1`
11	Measured value, scaled value `M_ME_NB_1`
12	Measured value, scaled value with time tag `M_ME_TB_1`
13	Measured value, short floating point value `M_ME_NC_1`

#	Meaning
14	Measured value, short floating point value with time tag `M_ME_TC_1`
15	Integrated totals `M_IT_NA_1`
16	Integrated totals with time tag `M_IT_TA_1`
17	Event of protection equipment with time tag `M_EP_TA_1`
18	Packed start events of protection equipment with time tag `M_EP_TB_1`
19	Packed output circuit information of protection equipment with time tag `M_EP_TC_1`
20	Packed single-point information with status change detection `M_PS_NA_1`
21	Measured value, normalized value without quality descriptor `M_ME_ND_1`
30	Single point information with time tag `CP56Time2a M_SP_TB_1`
31	Double point information with time tag `CP56Time2a M_DP_TB_1`
32	Step position information with time tag `CP56Time2a M_ST_TB_1`
33	Bit string of 32 bit with time tag `CP56Time2a M_BO_TB_1`
34	Measured value, normalized value with time tag `CP56Time2a M_ME_TD_1`
35	Measured value, scaled value with time tag `CP56Time2a M_ME_TE_1`
36	Measured value, short floating point value with time tag `CP56Time2a M_ME_TF_1`
37	Integrated totals with time tag `CP56Time2a M_IT_TB_1`
38	Event of protection equipment with time tag `CP56Time2a M_EP_TD_1`
39	Packed start events of protection equipment with time tag `CP56Time2a M_EP_TE_1`
40	Packed output circuit information of protection equipment with time tag `CP56Time2a M_EP_TF_1`
45	Single command `C_SC_NA_1`
46	Double command `C_DC_NA_1`
47	Regulating step command `C_RC_NA_1`
48	Setpoint command, normalized value `C_SE_NA_1`
49	Setpoint command, scaled value `C_SE_NB_1`
50	Setpoint command, short floating point value `C_SE_NC_1e`
51	Bit string 32-bit `C_BO_NA_1`
58	Single command with time tag `CP56Time2a C_SC_TA_1`
59	Double command with time tag `CP56Time2a C_DC_TA_1`
60	Regulating step command with time tag `CP56Time2a C_RC_TA_1`
61	Setpoint command, normalized value with time tag `CP56Time2a C_SE_TA_1`
62	Setpoint command, scaled value with time tag `CP56Time2a C_SE_TB_1`
63	Setpoint command, short floating point value with time tag `CP56Time2a C_SE_TC_1`
64	Bit string 32 bit with time tag `CP56Time2a C_BO_TA_1`
70	End of initialization `M_EI_NA_1`
100	General interrogation command `C_IC_NA_1`
101	Counter interrogation command `C_CI_NA_1`
102	Read command `C_RD_NA_1`
103	Clock synchronization command `C_CS_NA_1`
104	IEC 101 test command `C_TS_NB_1`
105	Reset process command `C_RP_NC_1`
106	IEC 101 delay acquisition command `C_CD_NA_1`
107	Test command with time tag `CP56Time2a C_TS_TA_1`
110	Parameter of measured value, normalized value `P_ME_NA_1`
111	Parameter of measured value, scaled value `P_ME_NB_1`
112	Parameter of measured value, short floating point value `P_ME_NC_1`
113	Parameter activation `P_AC_NA_1`
120	File ready `F_FR_NA_1`
121	Section ready `F_SR_NA_1`
122	Call directory, select file, call file, call section `F_SC_NA_1`
123	Last section, last segment `F_LS_NA_1`
124	Ack file, ack section `F_AF_NA_1`

#	Meaning
125	Segment `F_SG_NA_1`
126	`F_DR_TA_1`
127	QueryLog - request archive file `F_SC_NB_1`
128 .. 255	Vendor-specific Type ID `128` .. `255`

Modbus

This page describes the Modbus protocol enforcer for industrial communication security.

Modbus is a communication protocol that Modicon Incorporated originally developed for use with their programmable logic controllers (PLCs). This protocol enables devices to communicate and exchange data in industrial automation systems.

Modbus profiles define specific function codes and register addresses or coil addresses that control how data transfers occur. The function code in the Modbus TCP protocol tells the system what type of data operation to perform. The device automatically blocks data packets that do not match the profiles you set up. If the device detects an error in the communication, it ends the data connection based on how you set up the system. You can use the predefined function code lists or the function code generator to select the necessary function codes.

When you enable the Modbus enforcer profile, TBF applies these security rules to incoming data.

- The device accepts data packets that contain only the function codes you list in the **Function Code** field.
- The device rejects data packets that contain any function codes not listed in the **Function Code** field.

Find the Modbus protocol settings under **Firewall > Enforcer > Modbus**.

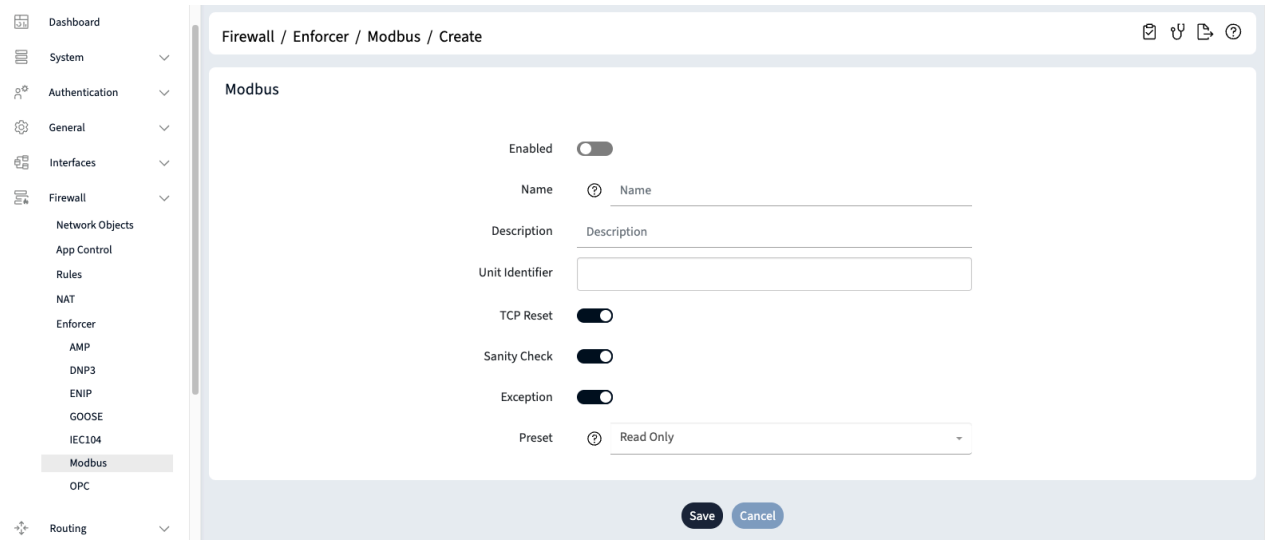


Figure 131: Modbus

Related topics

- *Enforcer rule requirements* on page 170: Firewall rule requirements for each enforcer. The Modbus destination port must be `502`.

Modbus settings

This page describes the Modbus enforcer configuration settings.

Enabled

This setting controls whether the Modbus enforcer is active on the device.

- **Enabled**: This option enables the Modbus enforcer to monitor and control Modbus traffic.
- **Disabled** (default): This option disables the Modbus enforcer, which allows Modbus traffic to pass through without inspection.

Name

This field assigns a descriptive name to identify this Modbus enforcer configuration. Enter up to 100 characters to help you distinguish between multiple enforcer profiles.

Description

This field provides additional details about the purpose or configuration of this Modbus enforcer. Enter up to 250 characters to document the role of the enforcer in the network.

Unit identifier

This field specifies which Modbus TCP unit identifiers the enforcer accepts. This helps control which devices communicate through the enforcer.

- `<empty>` (default): This option accepts data packets from any unit identifier, and does not filter based on device identity.
- `0..255`: This option accepts data packets only from specified unit identifiers. Set this up in two ways:
  - Single unit identifier: Enter one number, such as `1`, to allow only that specific device.
  - Multiple unit identifiers: Enter comma-separated numbers, such as `1,2,3`, to allow several specific devices.

TCP Reset

This setting determines whether the device closes the TCP connection when it detects protocol violations or data validation errors.

- **Enabled** (default): This option closes the TCP connection immediately when violations occur. The connection automatically reopens when a new valid request arrives, to help provide a clean restart.
- **Disabled**: This option keeps the TCP connection open even when violations occur, which allows communication to continue.

Sanity Check

This setting controls whether the device validates incoming data packets against Modbus protocol standards before the device processes them.

- **Enabled** (default): This option validates data packets to verify they meet Modbus format requirements. This option blocks any packets that do not comply with the security profiles.
- **Disabled**: This option skips validation and processes incoming packets without checking their format or compliance.

Exception

This setting controls whether the device sends error responses back to clients when it detects protocol violations or validation failures.

- **Enabled** (default): This option sends exception responses to inform clients about detected errors, then closes the Modbus TCP connection to help prevent further issues.
- **Disabled**: This option silently handles detected errors without sending responses to clients, and keeps the Modbus TCP connection active.

Preset

This setting selects a predefined set of allowed Modbus function codes based on common use cases. Each preset includes different levels of access permissions.

- **Read Only** (default): This preset allows only data `read` operations, which helps provide a restrictive security level.

  `1,2,3,4,7,11,12,17,20,24`

- **Read Write**: This preset allows both `read` and `write` operations, suitable for devices that need to modify data.

  `1,2,3,4,5,6,7,11,12,15,16,17,20,21,22,23,24`

- **Programming**: This preset includes advanced programming functions in addition to `read` and `write` operations, typically used for device configuration.

  `1,2,3,4,5,6,7,11,12,15,16,17,20,21,22,23,24,40,42,90,125,126`

- **All**: This preset permits possible function codes, which helps provide high flexibility but low security.

  `1,2,..,254,255`

- **Custom**: This preset enables you to specify custom function codes in the **Function Code** field for tailored security requirements.

Modbus rules

This page describes how to configure Modbus enforcer rules.

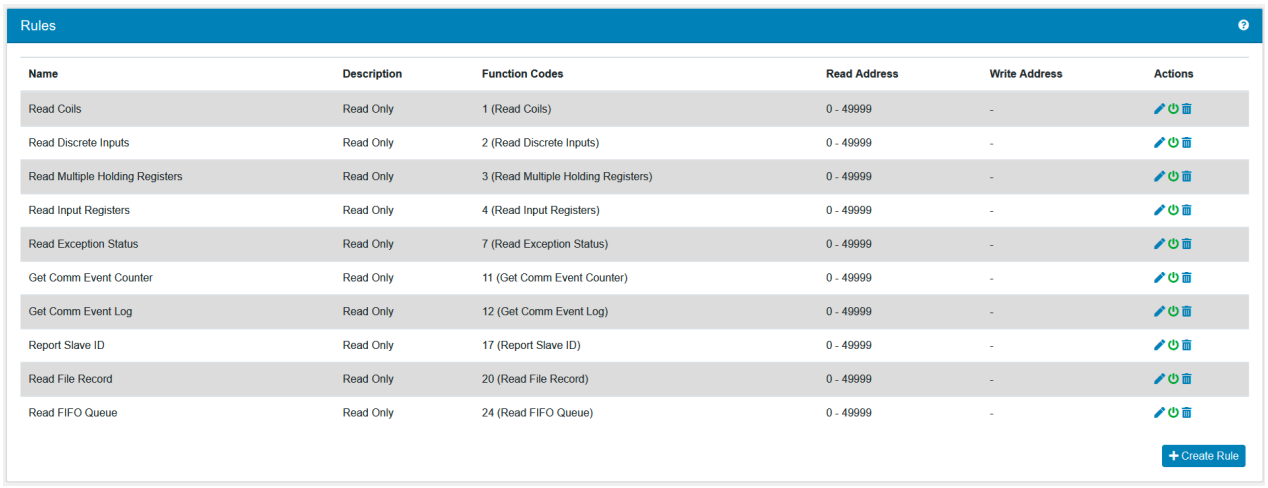


Figure 132: Modbus rules

Name

Specify a unique name for the Modbus rule. Enter up to 100 characters.

Description

Provide an optional description that explains the purpose of the Modbus rule. Enter up to 250 characters.

Function Code

Select the Modbus function code that this rule applies to. The valid range is:

- `0..255`

Read Address Start

Set the first address in the range for `read` operations. The default is `0`. The valid range is:

- `0..65535` (2^{16}-1)

Read Address Length

Specify how many consecutive addresses to include in the `read` operation range. The default is `1`. The valid range is:

- `0..65535` (2^{16}-1)

Write Address Start

Set the first address in the range for `write` operations. The default is `0`. The valid range is:

- `0..65535` (2^{16}-1)

Write Address Length

Specify how many consecutive addresses to include in the `write` operation range. The default is `1`. The valid range is:

- `0..65535` (2^{16}-1)

Modbus function codes

This page lists the Modbus function codes and their descriptions.

Function code	Description	Read address range	Write address range
1	Read coils (discrete outputs)	`<0..65535>`	-
2	Read discrete inputs	`<0..65535>`	-
3	Read holding registers	`<0..65535>`	-
4	Read input registers	`<0..65535>`	-
5	Write single coil	-	`<0..65535>`
6	Write single register	-	`<0..65535>`
7	Read exception status	-	-
8	Diagnostics	-	-
11	Get communication event counter	-	-
12	Get communication event log	-	-
13	Program controller (584/984 series)	-	-
14	Poll controller (584/984 series)	-	-
15	Write multiple coils	-	`<0..65535>`
16	Write multiple registers	-	`<0..65535>`
17	Report slave ID	-	-
20	Read file record	-	-
21	Write file record	-	-
22	Mask write register	-	`<0..65535>`
23	Read/write multiple registers	`<0..65535>`	`<0..65535>`
24	Read FIFO queue	`<0..65535>`	-
40	Program controller (Concept series)	-	-
42	Concept symbol table	-	-
43	Encapsulated interface transport	-	-
48	Advantech Co. Ltd. management functions	-	-
66	Scan Data Inc. expanded read holding registers	-	-
67	Scan Data Inc. expanded write holding registers	-	-
90	Unity programming/OFS	-	-
100	Scattered register read	-	-
125	Schneider Electric firmware functions	-	-

OPC

OLE for Process Control (OPC) is a communication standard that provides access to process control information on Microsoft Windows systems, with deep packet inspection (DPI) capabilities for network security.

Overview

OPC is a communication standard that developers built on OLE, COM, and DCOM technologies. This standard provides access to process control information on Microsoft Windows systems.

OPC serves as an integration protocol for industrial environments. The OPC enforcer acts as a security function that helps protect network communications by blocking data packets that violate specified security profiles. When you request it, the device validates data packets for both accuracy and fragment characteristics. The device monitors and verifies OPC data connections, helping protect against invalid or malicious data packets. This function dynamically opens TCP ports for each data connection. When an OPC server makes a request, the device establishes the data connection exclusively between the OPC server and its corresponding OPC client.

Deep packet inspection requirements

To enable DPI, set up an authentication level of 5 or lower on the end device. The end device can be a computer or any other equipment that can send OPC data packets. The authentication level determines what type of authentication an OPC client requires to connect with an OPC server.

State information clearing

The device clears state information from the packet filter when the following events occur:

- When the device applies profiles that it saves to the data stream.
- When you enable or disable the routing function on a router interface.

This process includes removing potential DCE RPC information from the OPC enforcer. During this operation, the device interrupts any open communication connections.

Access OPC settings

Find the OPC settings, under **Firewall > Enforcer > OPC**.

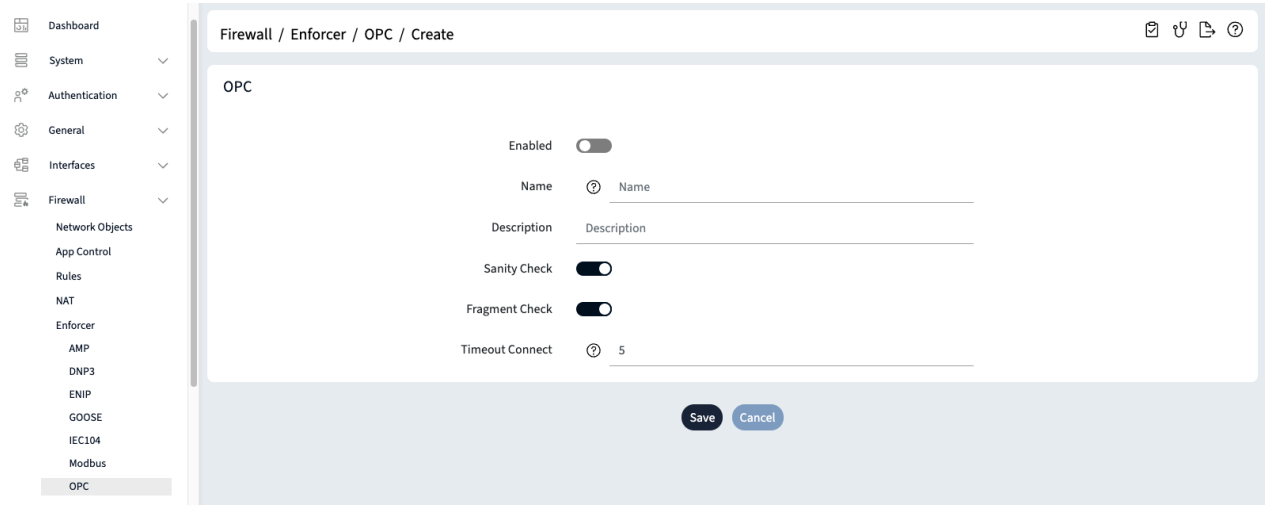


Figure 133: OPC

Related topics

For configuration details, see:

- *OPC settings* on page 193: Configuration parameters and options
- *Enforcer rule requirements* on page 170: Firewall rule requirements for each enforcer

OPC settings

This topic describes the settings for the OPC enforcer deep packet inspection (DPI) functionality.

Overview

The OPC enforcer offers multiple settings that allow you to control DPI behavior and security validation. These settings help you customize how the system monitors and validates OPC communications.

Settings

Enabled

This setting controls whether the OPC enforcer is active and monitors network traffic.

- **Enabled**: This option enables OPC monitoring and enforcement.
- **Disabled** (default): This option disables OPC monitoring.

Name

This field assigns a unique identifier to the OPC enforcer instance. Enter a descriptive name using 0..100 characters to help you identify this enforcer configuration.

Description

This field provides additional details about the purpose or configuration of the OPC enforcer. Enter a description using 0..250 characters to document the role of the enforcer in the network.

Sanity Check

This setting controls whether the system validates data packets against OPC protocol standards.

- **Enabled** (default): This option validates data packets to help ensure they meet OPC format and specification requirements. TBF blocks packets that do not comply with the profiles.
- **Disabled**: This option skips data packet validation and allows packets to pass through.

Fragment Check

This setting controls whether the system examines data packets for fragmentation issues that could indicate security threats.

- **Enabled** (default): This option analyzes packets for suspicious fragmentation patterns that attackers might use.
- **Disabled**: This option skips fragment analysis and improves performance but reduces security.

Timeout Connect

This setting specifies how long in seconds the system keeps dynamic TCP ports open after OPC data connections become inactive. This helps manage system resources and security.

- `1..300`: This value specifies the timeout period in seconds. The default is `5` seconds.
- `0`: This setting disables the timeout and keeps OPC connections open indefinitely.

Related topics

For conceptual information, see:

- *OPC* on page 192: Overview and functionality

12. Certificates

A certificate serves as digital proof of identity within a network and establishes trust between devices and users.

Find the certificate settings under **General > Certificates > Certificates**.

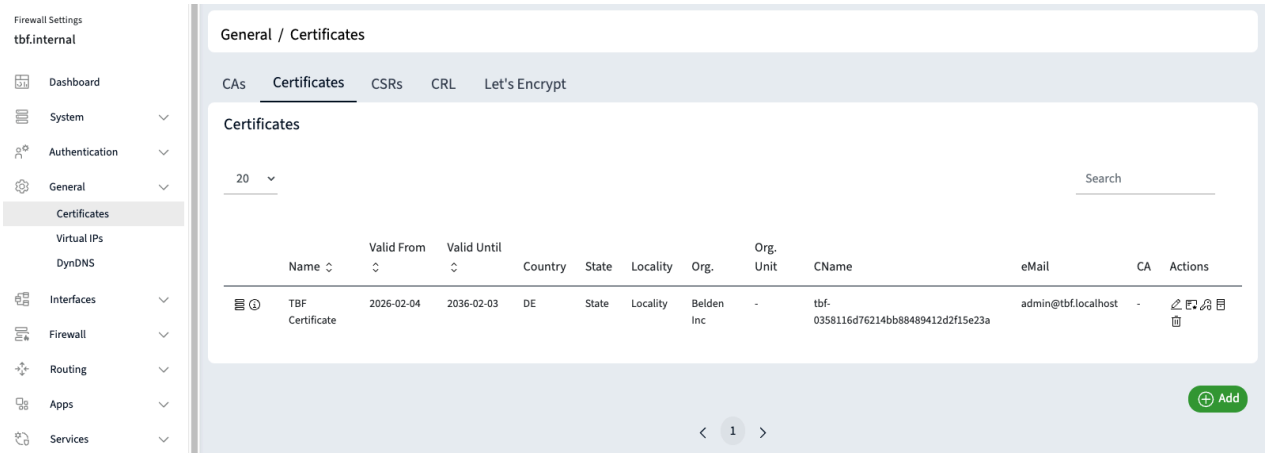


Figure 134: Certificates

A certificate acts as a digital identification card for the network. It proves the identity of devices and users, and allows them to trust each other during secure communications.

TBF supports exporting both the certificate and its private key in X.509 PEM format. This feature helps you create backups or transfer certificates to other systems.

WARNING: Protect the private key at all times. If someone gains unauthorized access to the private key, revoke the certificate immediately to prevent security breaches.

Each certificate requires a unique name within the system. TBF helps prevent you from creating two certificates with identical names.

When you create a certificate, assign it a type: either **Server** or **User**.

Note: TBF locks this setting after creation, so assign carefully.

The certificate type determines which security attributes TBF applies automatically. These attributes control where and how you can use the certificate.

TBF offers three methods to add certificates:

- **Create certificate**: Generate a new certificate directly in the system.
- **Import certificate**: Upload an existing certificate file.
- **Import P12**: Upload a PKCS#12 file that contains both the certificate and the private key.

Certificate settings

To create certificates, fill in standard identification details that verify the identity of the certificate holder:

See *Certificate settings* on page 194.

Certificate settings

This topic describes the standard certificate settings used across certificate creation and management.

Settings

Country

This setting specifies the country where the company is registered.

State or Province

This setting specifies the state or province where the company operates.

City

This setting specifies the city where the company operates.

Organization

This setting specifies the company name.

Organizational Unit

This setting specifies the specific department within the company.

Common Name

This setting specifies the domain name, host name, or URL associated with the company.

Email Address

This setting specifies a contact email address for the company.

CAs

Certificate authorities (CAs) are trusted entities that issue and manage digital certificates for secure network communication.

Find the CA settings under **General > Certificates > CAs**.

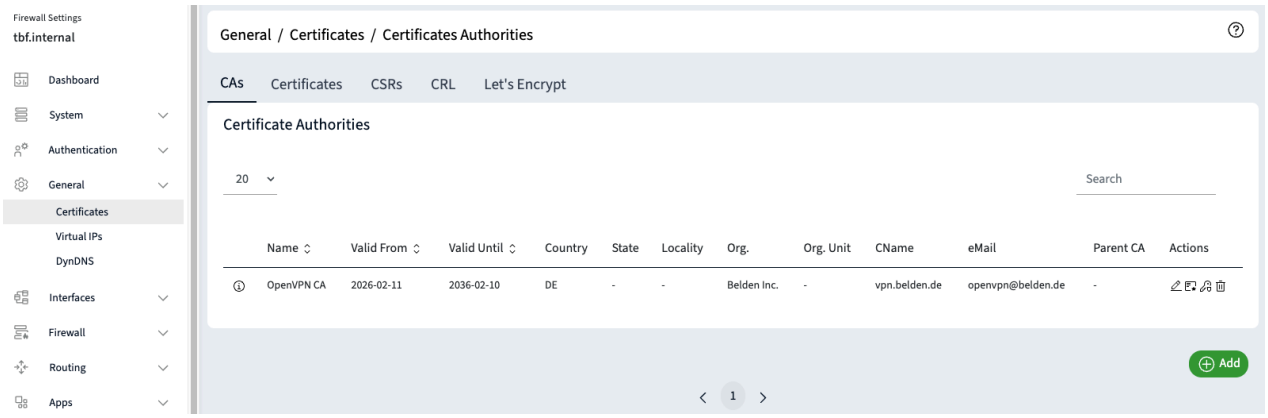


Figure 135: Certificate authorities

A CA acts as a trusted third party that issues digital certificates to verify the identity of entities in the network. Think of a CA as a digital passport office that confirms who is who in the network environment. CAs are essential to establish secure communication channels between devices, services, and users.

When you create a CA, you establish a foundation of trust for the entire network infrastructure. The CA then issues certificates to other devices and services, which builds a chain of trust that enables secure communication protocols such as HTTPS, SSL/TLS, and VPN connections. This chain works like a family tree of trust, where each certificate traces back to the original CA.

WARNING: Protect the private key of the CA with a high level of security. If someone gains access to this key, then they compromise the entire certificate chain, which forces you to revoke the issued certificates and create a new CA from scratch.

On the CAs overview, edit existing CAs and export their certificates and private keys in `X.509 PEM` format.

To create a new CA, provide a unique name that no other CA in the system uses. Create a CA with one of the following methods:

- Create CA
- Create intermediate CA
- Import CA

When you edit a CA, set the **Next Serial Number**. TBF applies this value when you create new certificates using this CA.

CA settings

These are setting descriptions for certificate authority creation and configuration forms.

Settings

Country

This field specifies the country where the organization is legally registered. Use the 2-letter ISO country code. For example, `US` for United States.

State or Province

This field specifies the state or province where the organization is located. Enter the full name without abbreviations.

City

This field specifies the city where the organization is located. Enter the full city name.

Organization

This field specifies the complete legal name of the organization as it appears in official documents.

Organizational Unit

This field specifies the specific department, division, or unit within the organization that manages the certificate. This field is optional.

Common Name

This field specifies the fully qualified domain name (FQDN), hostname, or URL that uniquely identifies the certificate authority. This is the server name that users access.

Email Address

This field specifies a valid contact email address for certificate-related communications. This field is optional but recommended for administrative purposes.

Manage CAs

Create and manage certificate authorities (CAs) to establish communication that assists in establishing security in the network infrastructure.

CAs act as trusted entities that issue digital certificates to verify the identity of devices and services in the network. Use the CA management to create new CAs or to modify existing CAs. Create CAs with different methods that depend on security requirements and network setup.

Perform the following steps:

1. Choose from the following creation methods:
   - **Create new**: Generate a completely new CA with a fresh private key and certificate. This option helps provide high security because the system creates the components newly.
   - **Import existing**: Upload an existing CA certificate and private key from another system. Use this option when you migrate from another certificate management system or when you already have established CAs.
   - **Generate from CSR**: Create a CA certificate from a certificate signing request (CSR). This method allows you to use a CSR that someone generated elsewhere while you create the CA certificate on this system.
2. Fill in the certificate fields with the information of the organization.

   These fields include details like organization name, country, and common name that identify the CA. Accurate information helps establish trust with devices and users. See the *CA settings* on page 196 for detailed descriptions of each field.
3. Set the starting serial number for certificates that this CA issues.

   Each certificate that a CA issues must have a unique serial number. The system automatically increments this number for each new certificate. Set a custom starting value, which is useful for tracking or organizational purposes, or accept the default value.
4. To complete the setup, save the CA configuration.

The new CA appears in the CA list and is ready to issue certificates for network devices and services. Use this CA to create and sign certificates that establish trusted connections that assist in establishing security throughout the network.

CSRs

A certificate signing request (CSR) is an unsigned certificate that you submit to a certificate authority to obtain a digital identity certificate.

Find the CSR settings under **General > Certificates > CSR**.

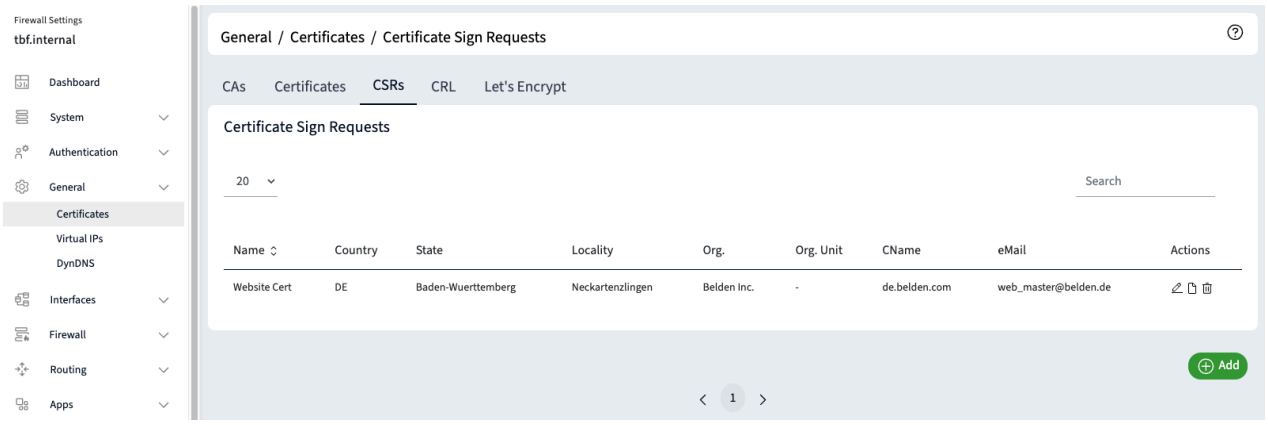


Figure 136: Certificate signing requests

Create a CSR using two methods:

- **Create CSR**: Generate a new certificate signing request from scratch
- **Import CSR**: Upload an existing certificate signing request file

Related topics:

- *CSR settings* on page 198
- *Import CSR* on page 198
- *Sign with certificate authority* on page 198

CSR settings

This topic provides reference information for certificate signing request settings and their descriptions.

Settings

Country

This field specifies the two-letter country code where the organization has legal registration, for example, *US* for United States or *CA* for Canada.

State or Province

This field specifies the full name of the state or province where the organization operates. Avoid abbreviations.

City

This field specifies the city where the main office of the organization operates.

Organization

This field specifies the complete legal name of the organization as it appears in official business documents.

Organizational Unit

This field specifies the specific department, division, or team within the organization that uses this certificate. This field is optional.

Common Name

This field specifies the fully qualified domain name (FQDN) or IP address that users access. Confirm that this matches exactly what appears in the browser address bar

Email Address

This field specifies a valid email address where the certificate authority sends notifications about the certificate

Import CSR

Import an existing certificate signing request (CSR) from another source.

Use this procedure when you have a CSR from an external source that you want to add to the system. This proves useful when you have generated a CSR elsewhere or received one from a certificate authority.

Perform the following steps:

1. Navigate to **General > Certificates > CSR**.
2. Select **Import CSR**.
3. Copy the complete CSR text from the source file or application.
   The CSR text typically begins with `-----BEGIN CERTIFICATE REQUEST-----` and ends with `-----END CERTIFICATE REQUEST-----`.
4. Paste the CSR text into the text field.
5. To store the imported CSR in the system, select **Save**.

Sign with certificate authority

Sign a certificate signing request (CSR) using a certificate authority (CA).

Prerequisites:

Before you begin, verify that you have created both a CA and a CSR, and saved them to the system.

Perform the following steps:

1. Open the CSR you want to sign for editing.
2. For the **Sign With** setting, select an available CA from the drop-down list.
3. Choose the certificate type in the **Type** field:
   - **Server**: Use this option for server authentication certificates.
   - **User**: Use this option for client authentication certificates.
4. In the **Valid Days** field, enter the number of days the certificate should remain valid.
   TBF sets this to `3650` days by default. Adjust this value based on the security requirements.
5. To apply the configuration, save the changes.
6. To start the signing process, select **Sign via Certificate Authority** in the **Actions** column.



Figure 137: Sign with CA

Note: TBF creates a new certificate object and automatically copies the signed certificate into the CSR certificate field. The certificate is ready for use.

CRL

This page describes Certificate Revocation List (CRL) management.

Find the CRL settings under **General > Certificates > CRL**.

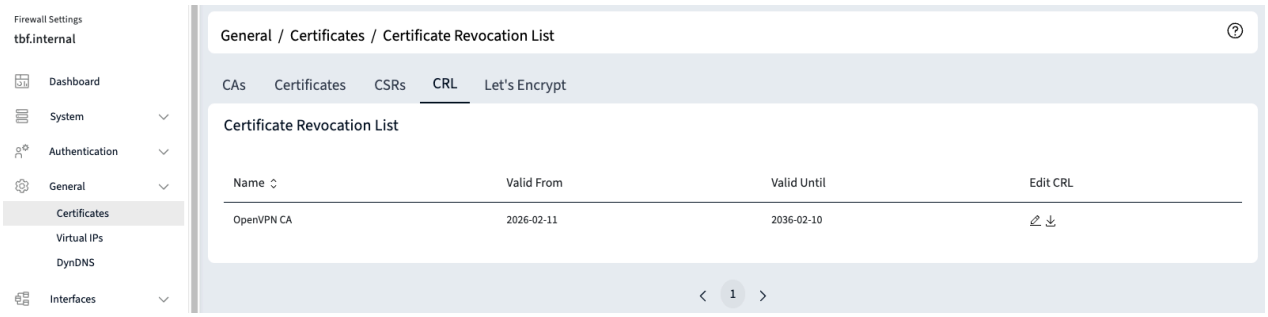


Figure 138: Certificate revocation list

A certificate revocation list (CRL) is a database that contains certificates that a certificate authority (CA) issues and that are no longer trustworthy or valid.

When a CA revokes a certificate, TBF records the date and time of the revocation.

There are several revocation reasons:

- Unspecified
- Affiliation changed
- CA compromised
- Cessation of operation
- Key compromised
- On hold
- Superseded

Let's Encrypt

Let's Encrypt is a nonprofit certificate authority (CA) that offers free SSL/TLS certificates for domain owners.

Find the Let's Encrypt settings under **General > Certificates > Let's Encrypt**.

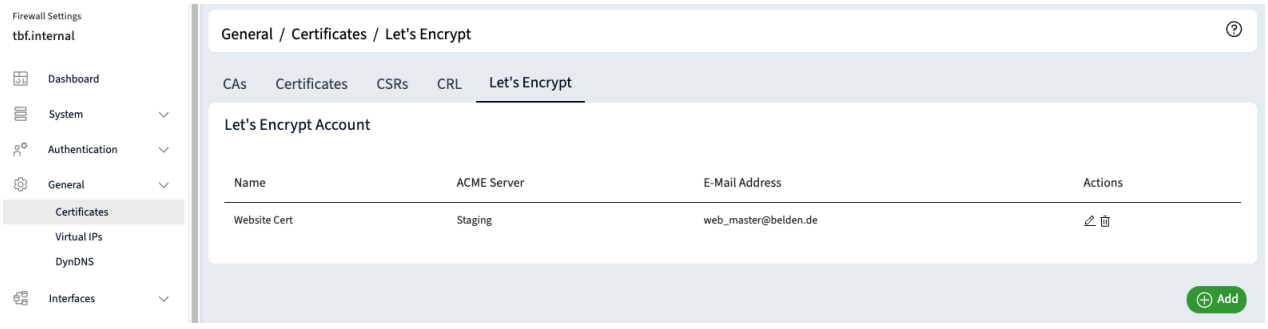


Figure 139: Let's encrypt

Let's Encrypt is a CA that operates as a nonprofit organization. It offers SSL/TLS certificates at no cost to website owners who need to secure their domains with encryption.

The Let's Encrypt service offers several authentication methods and includes automated certificate management features. Through the GUI, you can set up accounts, request certificate signatures, and handle certificate renewals automatically.

Related tasks

- *Create a Let's Encrypt account* on page 202
- *Sign a Let's Encrypt certificate* on page 202
- *Renew a Let's Encrypt certificate* on page 203
- *Revoke a Let's Encrypt certificate* on page 203

Related concepts

- *DNS ACME handle* on page 203

Create a Let's Encrypt account

To organize and manage certificates with Let's Encrypt, create a new Let's Encrypt account.

Before you can request certificates, create a Let's Encrypt account. This account serves as your identity with Let's Encrypt and helps organize certificates in one place. Creating an account is free and only takes a few minutes.

Important: You need this account to revoke any certificates you issue later, so keep the account information secure.

Perform the following steps:

1. Navigate to **General > Certificates > Let's Encrypt**.
2. Fill in the account settings as follows:

   Account Name

   To help you identify this ACME configuration later, enter a descriptive name.

   ACME Server

   Select the server environment:
   - **Staging (ACME v2)**: For testing certificates
   - **Production (ACME v2)**: For live certificates
   - **Custom**: For a custom ACME server

   Custom Server

   If you select **Custom** for the **ACME Server**, then enter the server URL. This option is useful when you run a local certificate server that supports the ACME protocol.

   Private key

   TBF generates this key automatically when you leave this field empty.

   E-Mail Address

   Enter an email address for account notifications and contact purposes.

   Auto Firewall Rule

   To automatically create a temporary firewall rule that allows certificate validation, enable this option. If you disable this option, then manually open port 80 on the **WAN** interface.
3. Save the account configuration.

Sign a Let's Encrypt certificate

Sign a certificate using Let's Encrypt with webserver, DNS, or custom script authentication.

Prerequisites:

Before you begin, verify that you have a Let's Encrypt account set up and a certificate signing request (CSR) ready.

To obtain a signed certificate, create a CSR and save it. During the CSR editing process, select the **Let's Encrypt** account and choose the preferred authentication method.

Perform the following steps:

1. Create and save a CSR.
2. Edit the CSR and select the **Let's Encrypt** account.
3. Choose one of the following authentication methods:
   - **Webserver**: Let's Encrypt verifies domain ownership by looking for a specific file on the web server of the domain. TBF automatically handles this verification process, as long as the DNS of the domain points to TBF.
   - **DNS**: Let's Encrypt verifies domain ownership by checking for a specific DNS record in the DNS settings of the domain. Select an ACME handle and create a CNAME entry for validation on the target domain. TBF manages the certificate authentication process.
   - **Custom Script**: Provide a custom authentication script.

   WARNING: Web authentication requires the web interface of TBF to be accessible on port 80 or 443. If this is not set up, then use DNS authentication instead.
4. To sign the certificate, select the **Sign** button on the right.

   The system generates the new certificate, which appears on the certificate overview page.

Important: To allow the signing process to work properly, verify that port 80 is open on the **WAN** interface. The DNS entry of the certificate must also point to TBF so it remains accessible during the signing process. Let's Encrypt needs to contact TBF to verify that the DNS entry is valid.

Renew a Let's Encrypt certificate

Manually renew a Let's Encrypt certificate that is approaching expiration.

Manually renew a signed Let's Encrypt certificate when it approaches expiration. TBF displays a **Renew** button on the right side of the certificate overview page when a certificate has 30 days or fewer remaining before expiration.

TBF also includes an automatic renewal process. A cron job runs weekly to automatically renew Let's Encrypt certificates that are approaching expiration.

Perform the following steps:

1. Go to the certificate overview page.
2. Find the certificate that has 30 days or fewer remaining before expiration.
3. Select the **Renew** button on the right side of the certificate entry.

Revoke a Let's Encrypt certificate

Revoke a signed Let's Encrypt certificate and optionally create a new certificate for the same domain.

Perform the following steps:

1. Go to the certificate overview page.
2. Select the **Revoke** button next to the signed certificate you want to revoke.

   TBF revokes the certificate with Let's Encrypt.
3. If you need a replacement, then create a new certificate for the same domain.

DNS ACME handle

If you plan to use DNS authentication instead of web authentication, then create a DNS ACME handle.

This handle creates a DNS entry that TBF can dynamically update and use as the CNAME entry for the actual domain that the system validates.

For example, the DNS handle looks like this:

```
177c0dc6-4d2e-486b-932e-db248b2dd123.auth.acme-dns.io
```

Then add it to the actual domain like this:

```
_acme-challenge.your-domain CNAME 177c0dc6-4d2e-486b-932e-db248b2dd123.auth.acme-dns.io.
```

Note: TBF cannot update the ACME handle during certificate signing requests.

13. Virtual IPs

Virtual IP addresses offer flexible network addressing solutions that include VRRP failover and IP aliases.

Virtual IP addresses enhance the network setup and let you assign multiple IP addresses to interfaces and create automatic backup systems when primary connections are unsuccessful.

TBF offers the following virtual IP address setups:

- *VRRP* on page 205: Creates automatic backup systems between multiple machines that share the same IP address.
- *IP alias* on page 206: Adds extra IP addresses to network interfaces.

For more complex VRRP setups, create *custom VRRP scripts* on page 206 that trigger backup switching according to specific requirements.

VRRP

Virtual Router Redundancy Protocol (VRRP) helps provide automatic failover for IP addresses across multiple machines.

Find the VRRP settings under **General > Virtual IPs**.

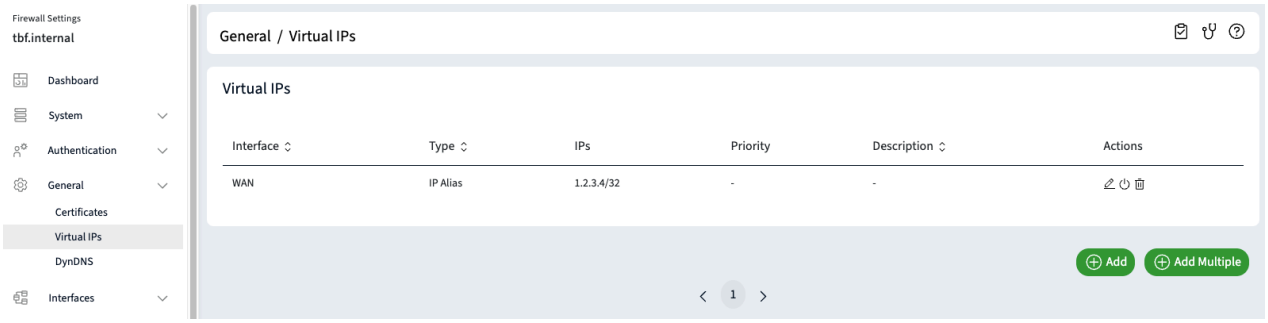


Figure 140: Virtual IPS

A VRRP IP address is a shared IP address that multiple machines use. The primary machine, called the *master*, holds the address, while one or more backup machines remain on standby. If the master goes offline, then a backup machine automatically takes over the address.

VRRP-enabled machines communicate through multicast messages on their network interface. When a machine stops receiving these multicast messages from its neighbors, it assumes the master role according to the priority settings you set.

Add a VRRP IP address only on top of an existing IP address on the same interface.

Note: Use VRRP only with static IP addresses. While DHCP IP addresses work, being unsuccessful in receiving a DHCP IP address triggers an automatic failover.

VRRP uses the following multicast destination IP addresses:

- IPv4: `224.0.0.18`
- IPv6: `fe02::12`

Set the priority level for each machine. The machine with the highest available priority becomes the master.

Set a password and verify that it matches on all machines that share the address.

Default Active

This setting is either primary or secondary. This setting enables an active/active cluster setup where some VRRP IP addresses are assigned to the primary firewall and others to the secondary firewall. Network clients need to use the different VRRP addresses as gateways. During a failover event, one firewall takes control of all IP addresses.

Track Interface

This setting overrides the trackable interface for this VRRP IP address. This is particularly useful when you have a VRRP set up on top of a bridge interface. Because the bridge only goes offline when all interfaces are offline, this setting ties the failover event to a specific interface. By default, it tracks the underlying interface of the current settings.

Important: A disk detected inoperability results in a VRRP service failover.

IP alias

An IP alias assigns additional IP addresses to network interfaces for multiple address configurations. A single interface handles multiple IP addresses simultaneously.

Find the virtual IP settings under **General > Virtual IPs**.

Define any IP address for the alias. However, verify that you use the correct netmask that matches the network configuration and specific requirements.

Note: When you use a netmask other than `/32`, TBF automatically creates an interface route for the entire network range.

Custom VRRP scripts

Create custom scripts to trigger VRRP failover according to specific conditions.

Add custom scripts to trigger a failover for VRRP IP addresses. Verify that scripts are executable and are located in the `/etc/keepalived/scripts.d` directory.

Perform the following steps:

1. Create the script in the `/etc/keepalived/scripts.d` directory.
2. Make the script executable.
3. Verify that the script returns the correct exit code:
   - `0` indicates success
   - Any non-zero value indicates detected failure

Important: If any script in the folder is unsuccessful, then the entire VRRP daemon changes its status to unsuccessful.

BGP default route check script

To trigger a failover when the BGP default route is not present, use the following script:

```
#!/bin/bash

ip r s | grep default | grep bgp
```

14. DynDNS

Dynamic Domain Name System (DynDNS) allows you to assign a DNS name to a dynamic IP address. The service helps provide consistent domain name access when an IP address changes frequently.

Find the DynDNS settings under **General > DynDNS**.

DynDNS lets you link a fixed domain name to an IP address that changes over time. This service proves particularly valuable when the internet service provider assigns you a different IP address each time you connect. You still need people to reach a device with the same web address.

For example, when you host a website or remote access service from a home network, DynDNS helps visitors find a device. This applies even when an IP address changes.

TBF works with several popular DynDNS service providers and includes customizable settings to match specific network setup and requirements.

Set DynDNS settings

Set up DynDNS to automatically update the domain name when an IP address changes.

DynDNS links the domain name to the correct IP address even when the internet service provider changes it. To enable automatic domain name updates, follow these steps.

1. Navigate to **General > DynDNS**.
2. Select a **Service Type** from the drop-down list.

   Different providers offer various features and update methods. For details about each service type, see the *supported service types* on page 208.
3. Select the **IP Type** that matches the network setup.

   If the network supports the newer protocol, then select IPv6. For standard internet connections, select IPv4.
4. Set up the **Connection** method based on how you want TBF to detect IP address changes.
   - To monitor the IP address of that interface, select a specific interface.
   - To monitor the IP address of the current default gateway, select a routing table.
   - To automatically verify and update the IP address every 2 minutes, select the periodic check.
5. Enter the complete **Hostname**.

   Include the full domain name, for example, `myserver.example.com`.
6. Enter the DynDNS service credentials.
   a. Enter the **Username** in the corresponding field.
   b. Enter the **Password** in the corresponding field.

   These credentials authenticate the device with the DynDNS service provider.
7. To apply the settings, select **Save**.

TBF now monitors the IP address and automatically updates the domain name whenever it detects a change. This helps point the domain to the current IP address.

DynDNS settings

This topic contains reference information for DynDNS settings that include supported service types and setting descriptions.

Supported service types

Select from the following **Service Type** options:

- **DynDNS**
- **Gandi**
- **dnsHome**
- **Strato**
- **Custom Script**

Settings

IP Type

This setting specifies the IP address type. Select from:

- **IPv4**
- **IPv6**

Connection

This field uses a specific network interface with its IP address, a routing table, or a periodic check. The periodic check runs every 2 minutes to detect IP address changes and automatically updates the DynDNS record.

When you select the routing table option, TBF uses the IP address of the current default gateway. When multiple gateways have the same IP address in the routing table, TBF uses the first active gateway it finds.

Hostname

This field specifies the complete hostname that includes the domain name that you want to assign the IP address to.

Username

This field specifies the username to authenticate with the DynDNS service. If the service requires authentication, this field is required.

Password

This field specifies the password to authenticate with the DynDNS service. If the service requires authentication, this field is required.

Custom Script

This setting runs a custom script with limited user privileges and restricted access to system resources to assist in establishing security.

The script accesses environment variables that match the GUI field values:

- `USERNAME`
- `PASSWORD`
- `HOSTNAME`
- `IPADDRESS`
- `IP_TYPE`

15. Routing

To control how network traffic flows through the system, set up routing settings.

The routing system determines how network packets travel between different networks and destinations. To optimize network performance and help provide connectivity, set up static routes, dynamic routing protocols, and routing tables.

Components

The routing system includes these key components:

***Routes* on page 209**

Static routes direct traffic to specific destinations through defined gateways or interfaces.

***Routing tables* on page 211**

Custom routing tables supplement the main routing table for policy-based routing.

***Load balancing* on page 213**

This component distributes traffic across multiple paths to improve performance and reliability.

***Gateways* on page 214**

Network gateways serve as next-hop destinations for routing traffic.

***Dynamic routing* on page 222**

Dynamic routing uses protocols like BGP, OSPF, and ISIS for automatic route discovery and management.

Advanced features

Additional routing capabilities include:

- *MPLS* on page 218: Multiprotocol label switching for traffic engineering
- *BFD* on page 223: Bidirectional forwarding detection for fast detected inoperability detection
- *Access lists* on page 236: Control route advertisements and filtering
- *Prefix lists* on page 236: Filter routes according to network prefixes
- *Route maps* on page 237: Advanced route manipulation and policy control

Getting started

To begin setting up routing:

1. For essential connectivity, set up basic *static routes* on page 209.
2. For the network infrastructure, set up *gateways* on page 214.
3. If needed for policy routing, then create custom *routing tables* on page 211.
4. For automatic route management, enable *dynamic routing protocols* on page 222.

Routes

Routes are static routes that direct network traffic to specific destinations in the system.

Find the routes settings under **Routing > Routes**

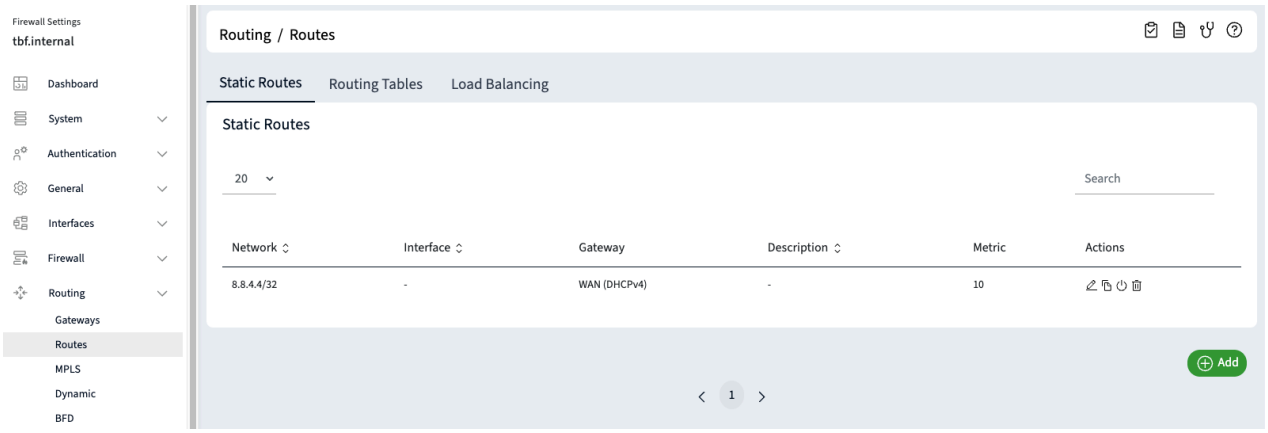


Figure 141: Static routes

Routes are predefined paths that tell the network where to send data packets. When you create a static route, you establish a fixed path that remains consistent until you manually change it. This offers precise control over how traffic flows through the network.

The following types of routes are available:

- *Gateway route* on page 210: Routes traffic through a specific gateway device
- *Interface route* on page 210: Routes traffic through a particular network interface
- *Multipath route* on page 210: Distributes traffic across multiple paths for load balancing

For detailed information about route settings, see *route settings* on page 210.

To modify the routing tables, see the *routing tables* on page 211.

Gateway route

Gateway routes direct traffic through a specific gateway device to reach destinations.

To specify a gateway, go to *gateway* on page 214, or to create a gateway, select the **Add** button. The route uses the IP address in the gateway as the next hop destination. The next hop destination is typically another router in the network.

Interface route

Interface routes send traffic directly through a network interface and do not use a gateway.

Select an interface for the route. TBF defines the route directly on that interface.

Note: To reach a client in the destination network, verify that the client is in the same L2 network.

Multipath route

Multipath routes distribute traffic across multiple gateways to help provide load balancing and redundancy.

Select multiple gateways and their metrics for the route to use. TBF considers the gateway status check to determine whether each gateway is active or inactive.

To change the BGP setup, navigate to **Routing > BGP**. See also the*BGP settings* on page 224.

Route settings

Use these settings to control how TBF manages static routes and their behavior and set route settings.

Settings

Enabled

This setting enables or disables a route. When you change this setting, TBF automatically adds or removes the route during the next route check.

Destination Network

This setting specifies the target network that the route directs traffic to. Set up one of the following route types:

- *Gateway* on page 210
- *Interface* on page 210
- *Multipath* on page 210

Metric

Assign a metric value to each route. When they have different metric values, multiple routes can share the same destination. The metric acts as a priority level, where the route with the lowest metric becomes the active route in the system.

Source

In the advanced settings, you can specify a route source. This setting helps when the firewall needs to use the route and has multiple IP addresses on the interface where you define the route.

Routing Table

In the advanced settings, you can select one or more routing tables. TBF automatically places each new route in the **Main** routing table, ID 254, by default.

Gateway Status

This setting automatically removes the route when the gateway becomes inoperable and restores the route when the gateway becomes operable again. TBF enables this setting by default. When you disable this setting, the route stays active even if monitoring checks show the gateway is inoperable.

Routing tables

Routing tables supplement the **Main** routing table by providing custom routing paths for specific clients and enabling multipath routing with gateway priorities.

Find the routing tables under **Routing > Routes**, on the **Routing Tables** tab.

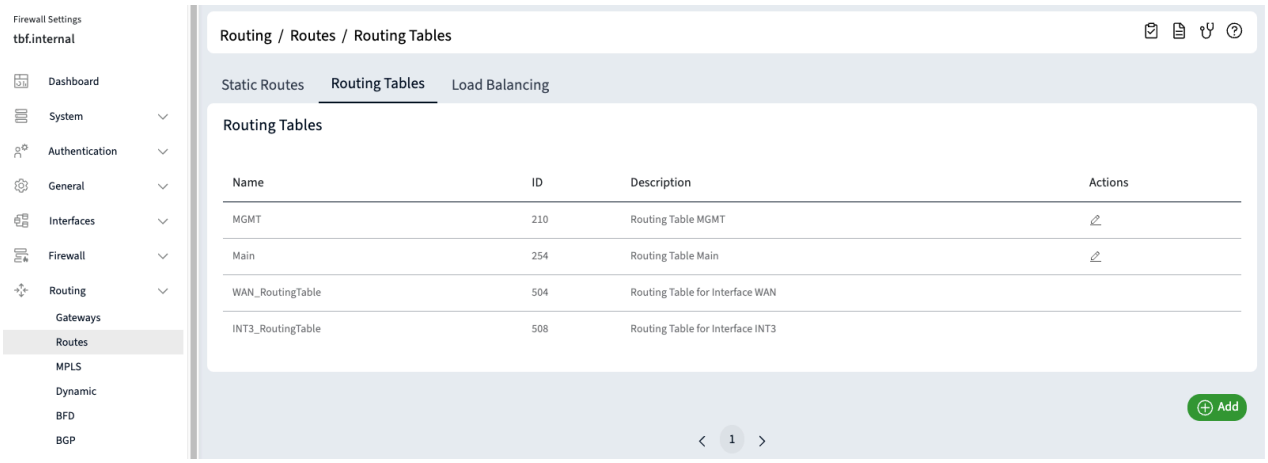


Figure 142: Routing tables

Note: The **Main** routing table with ID 254 is a built-in route that you cannot delete. Each new routing table requires a unique name and ID.

When you create or edit static routes, assign them to one or multiple routing tables. For more information, see *routes* on page 209.

Routing tables work as supplements to the **Main** routing table. TBF checks these custom tables before it consults the **Main** routing table. When TBF finds no match in the custom tables, TBF queries the **Main** routing table.

You can assign clients to a specific routing table through firewall rule options:

1. Open *firewall rules (forward and input)* on page 131.
2. Access the advanced settings.
3. Select the routing table for matched clients to use.

When you create or edit a routing table, you can link both IPv4 and IPv6 gateways to it. Each linked gateway has a priority value between 1 and 100, where lower numbers indicate higher priority.

Gateway priority determines routing behavior:

- **Single highest-priority gateway**: TBF designates it as the default gateway.
- **Multiple gateways with equal highest priority**: TBF treats them as a multipath route and creates a load-balanced internet connection.

Note: TBF only makes routing decisions when a connection starts. Once established, an open connection does not switch to a different gateway. The *DNS troubleshooting* on page 263 section contains information about multipath gateway scenarios.

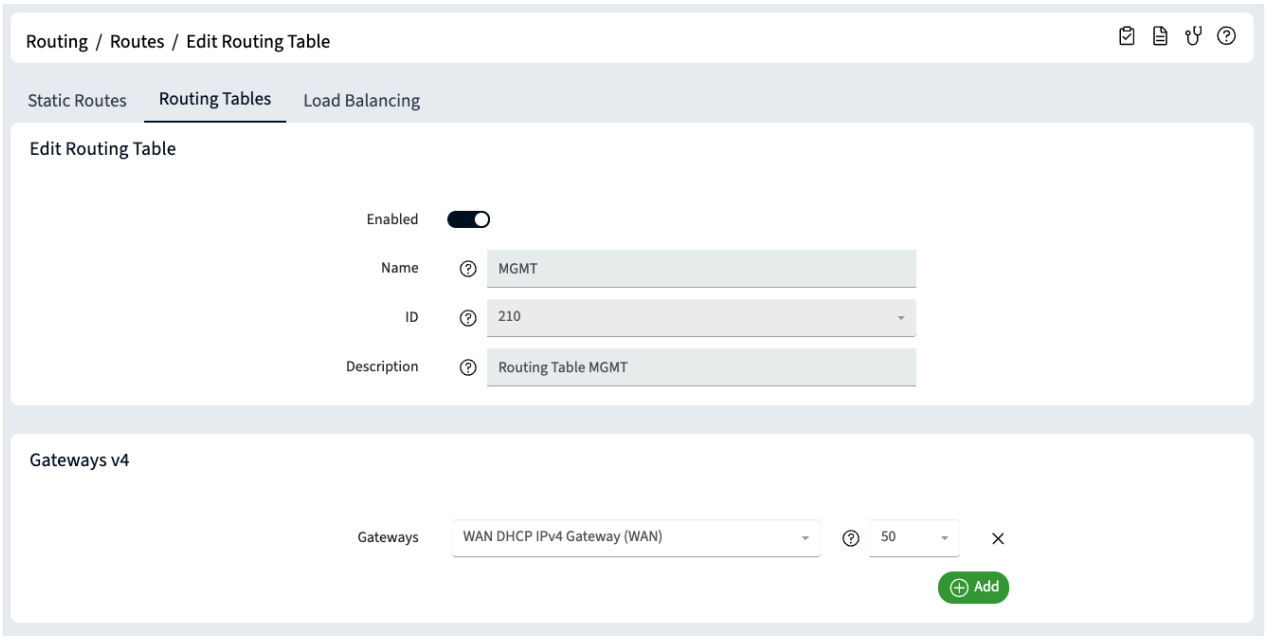


Figure 143: Edit routing table IPV4

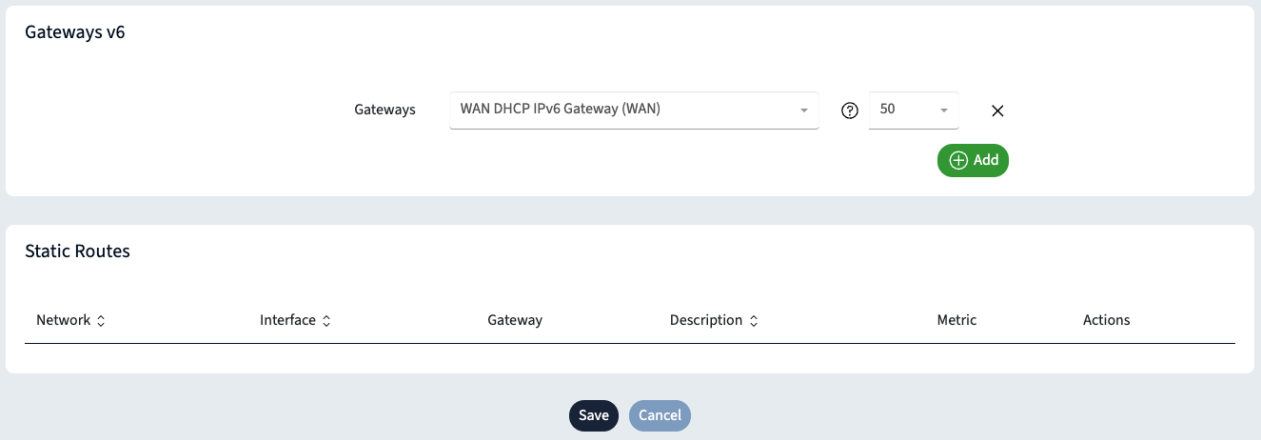


Figure 144: Edit routing table IPV6

You can adjust the weight of multipath routes directly in the gateway configuration under *gateway* on page 214. For Equal Cost Multipath Routing (ECMP), set equal weights for the gateways.

At the bottom of the edit screen, you can view the static routes linked to the selected routing table.

Load balancing

Load balancing allows you to use multiple routing tables simultaneously to distribute traffic across multiple gateways for improved performance and redundancy.

Find the load balancing settings under **Routing > Routes**, on the **Load Balancing** tab.

The policy determines how the system assigns each connection state to a specific routing table. TBF makes this decision when a connection starts, and the selected routing table remains consistent throughout the lifetime of the connection.

Note: This approach causes issues with servers that require multiple open connections, because the source IP address varies across connections when you use load balancing.

You can add multiple routing tables and assign each a priority value from 1 to 99.

Load balancing settings

The following settings are available when you set up load balancing.

Settings

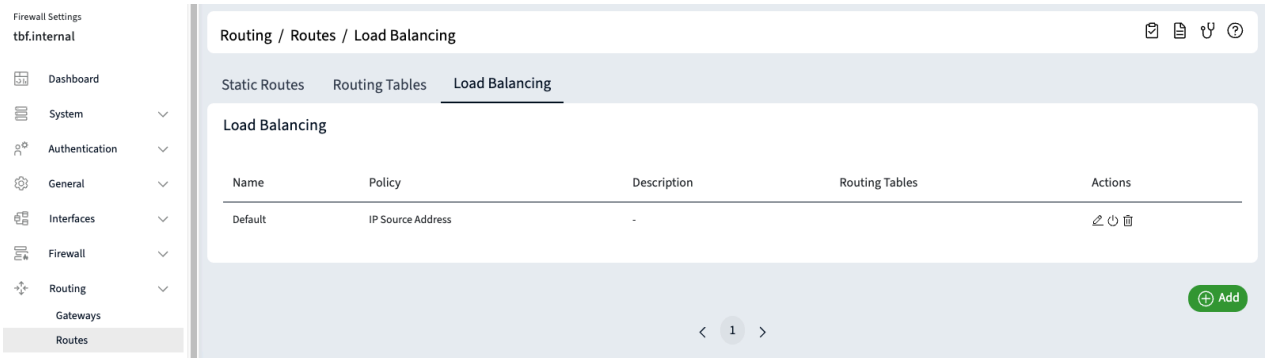


Figure 145: Load balancing

Name

Enter a name to identify the load-balancing configuration.

Description

Enter a description of the load-balancing configuration.

Policy

Select how traffic distributes:

- **Random**: This option is the default setting.
- **IP Source Address**
- **IP Source Address + IP Destination Address**

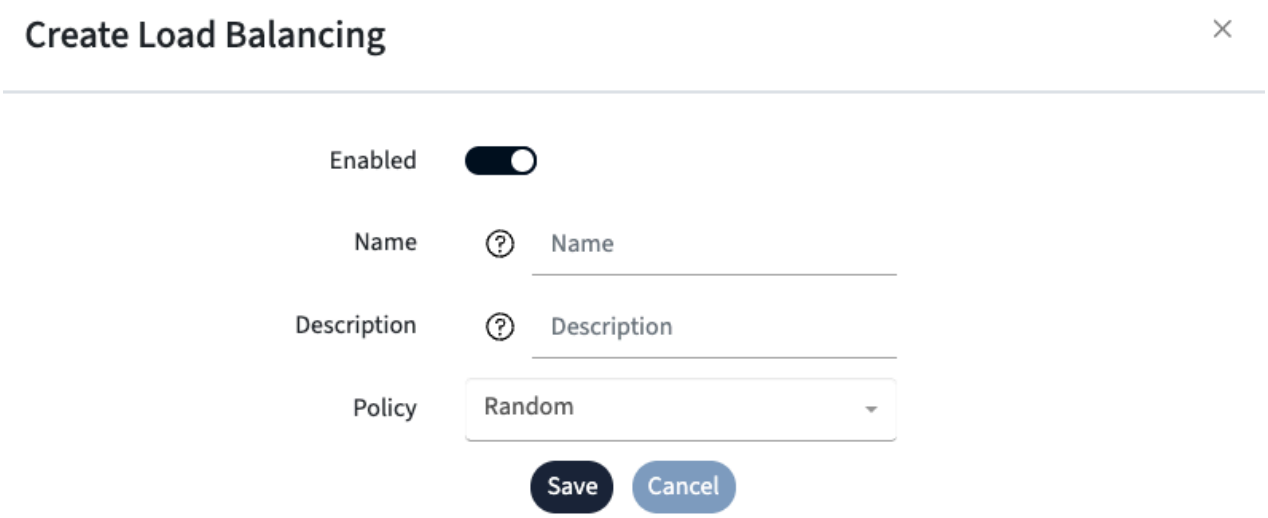


Figure 146: Create load balancing

Gateways

Gateways provide network routing functionality for both IPv4 and IPv6 networks, with automatic assignment for DHCP, PPP, and OpenVPN interfaces.

Find the gateway settings under **Routing > Gateways**.

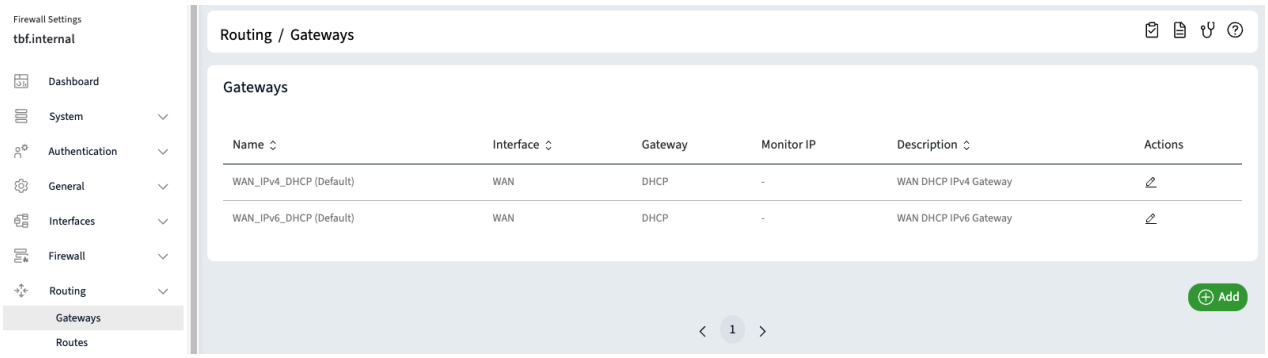


Figure 147: Gateways

TBF supports gateway creation for both IPv4 and IPv6 networks. TBF automatically assigns gateways to interfaces that use the DHCP, PPP, and OpenVPN protocols.

Each gateway requires 3 components:

- A unique **Name**
- The **Interface** where it operates
- The **Gateway IP** address

If you designate a gateway as the **Default Gateway**, verify that the assigned interface can reach it.

Before you set up the gateway, assign an IP address to the interface.

TBF allows only one **Default Gateway** for IPv4 and one for IPv6. When you save a new **Default Gateway**, TBF automatically removes the default status from any existing default gateway.

Create a gateway

Create and set up gateways with monitoring options for network routing functionality.

To help provide network reliability, enable gateway monitoring. By default, TBF monitors the gateway and sends ping requests to the **Gateway IP** address. To monitor a different IP address instead, enter the IP address in the **Monitoring IP Address**. When TBF detects that a gateway is unavailable, TBF automatically removes the gateway from the routing table to help prevent traffic routing issues.

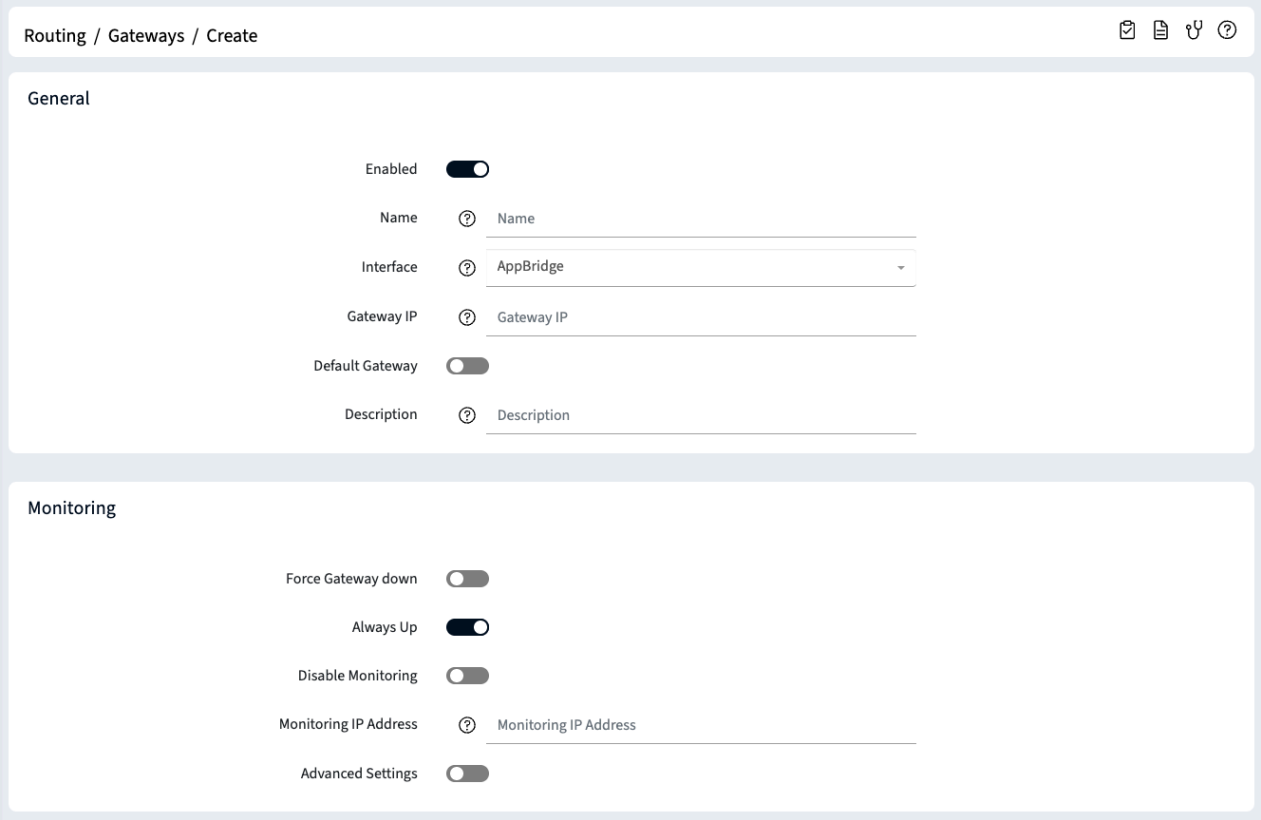


Figure 148: Create gateway

1. Navigate to **Routing > Gateways**.
2. To create a new gateway, select **Add**.
3. Enter a descriptive name in the **Name** field. Use a unique name within the gateway configuration.
4. Select the network interface from the **Interface** drop-down list. Select the interface that connects to the network where the gateway resides.
5. Enter the IP address of the gateway in the **Gateway IP** field. This address is the IP address of the router or device that handles traffic routing.
6. Optional: If this gateway handles traffic that does not match specific routes in the routing table, enable **Default Gateway**.
7. Set monitoring options based on the network requirements:
    - To manually disable the gateway, select **Force Gateway down**. Use this option for maintenance or troubleshooting.
    - To keep the gateway active in the routing table while TBF still monitors the gateway status, select **Always Up**. TBF continues to check the connectivity but does not remove the gateway automatically.
    - To keep the gateway permanently active without any health checks, select **Disable Monitoring**. Use this option only when you are certain the gateway is reliable.
8. Optional: To monitor connectivity to a specific server or device beyond the gateway itself, enter a different IP address in the **Monitoring IP Address** field.
9. To create the gateway, select **Save**.

TBF creates the gateway and adds the gateway to the active routing table. When you enable monitoring, TBF immediately begins health checks and uses ping requests to the specified IP address. You can view the gateway status and monitoring results in the gateway list.

Gateway monitoring

Gateway monitoring helps provide network reliability by tracking gateway availability and automatically managing routing table entries.

When a gateway is unsuccessful, TBF automatically removes the gateway from the routing table to help prevent sending network traffic to unreachable destinations.

You can view gateway status as follows:

- Through the dashboard widget for a quick overview
- Under **Diagnostics > Gateways** for detailed information

The monitoring system sends ping requests to test gateway connectivity. This method quickly identifies when a gateway stops responding.

Select from the following monitoring options based on the network requirements:

- **Standard monitoring**: This option sends ping requests to the gateway IP address at specified intervals. This option is a common option for typical network setups.
- **Custom monitoring**: This option sends ping requests to a different IP address that you specify. To test connectivity to a specific server or service beyond the gateway, use this option.
- **Always Up**: This option keeps the gateway active in the routing table while TBF still performs health checks. To get monitoring data but avoid automatic failover, use this option.
- **Disable Monitoring**: This option keeps the gateway permanently active without any health checks. If you are certain the gateway is reliable or if monitoring interferes with network operations, use this option.

Monitoring settings

Monitoring settings control how TBF tests gateway connectivity and determines gateway availability.

Settings

This section contains monitoring options that help you set up and fine-tune how TBF monitors the network gateways.

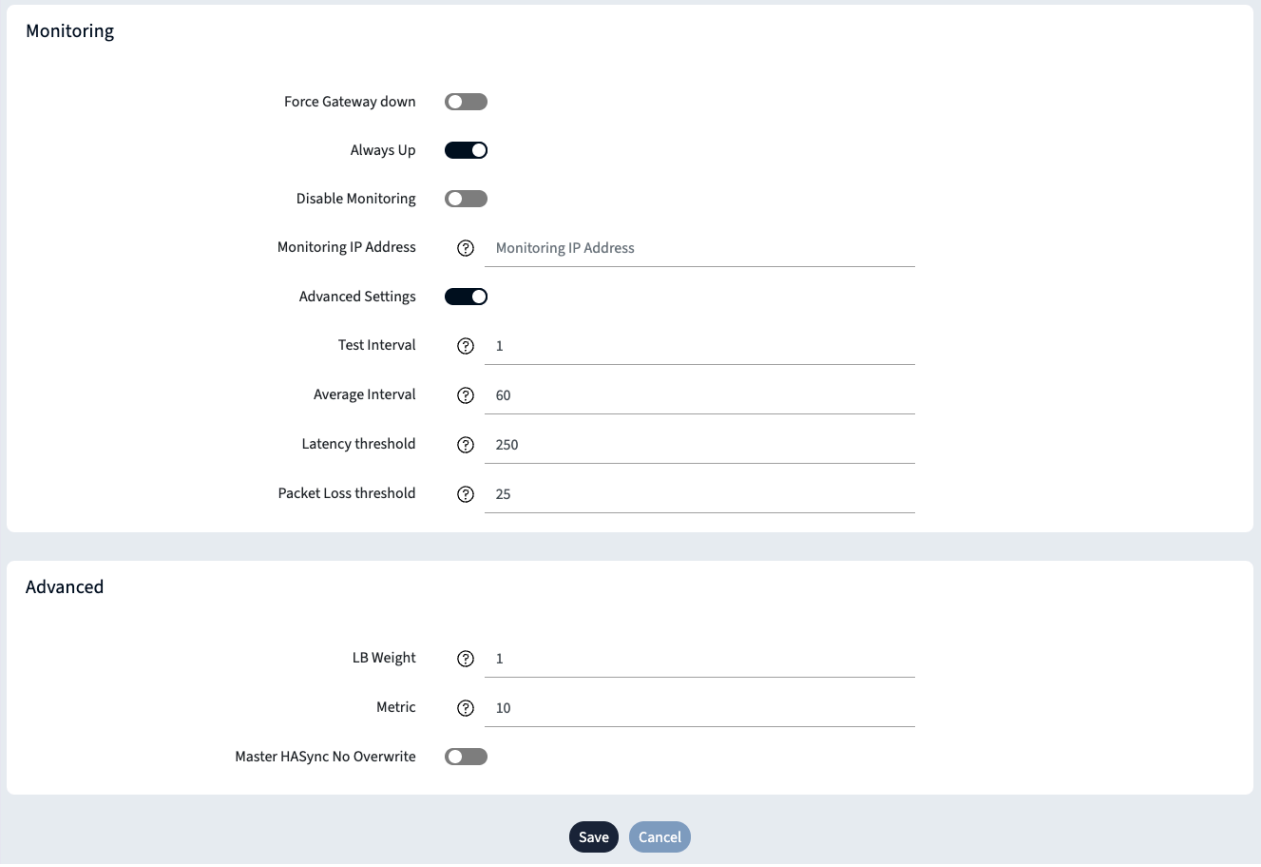


Figure 149: Gateway settings

Monitoring

Test Interval

This setting sets how often TBF sends ping requests to the monitoring IP address. TBF measures the value in seconds. For example, a setting of `30` means TBF pings the gateway every 30 seconds.

Average Interval

This setting defines the time window that TBF uses to calculate average test results. TBF uses these averaged results to determine if a gateway is valid. A longer interval helps provide more stable results but takes longer to detect errors.

Latency threshold

This setting sets the maximum acceptable response time from the monitoring IP address. When responses consistently take longer than this threshold, TBF considers the gateway too slow and marks the gateway as unavailable.

Packet Loss threshold

This setting defines the maximum percentage of lost packets that TBF allows within the average interval. When packet loss exceeds this threshold, TBF marks the gateway as unavailable and stops routing traffic through the gateway.

Advanced settings

Advanced gateway settings control load balancing, routing metrics, and high availability synchronization.

Load balancing

The **LB Weight** setting controls how TBF distributes network traffic between multiple gateways in the routing tables. TBF uses this weight value to determine the ratio of packets that it sends to each gateway.

For example, *gateway 1* has an LB weight of 1 and *gateway 2* has an LB weight of 2. TBF sends 2 packets to *gateway 2* for every packet that it sends to *gateway 1*. This creates a 1:2 distribution ratio.

Note: TBF applies this distribution ratio only when it establishes new connections. Existing connection flows continue to use their current gateway path.

Routing metrics

The **Metric** setting specifies the metric value for default routes when this gateway acts as the default gateway for the network. To maintain a static kernel backup default route with dynamic routing protocols like BGP or OSPF, modify this value.

Set the metric to `4278198272` for compatibility. The dynamic routing daemon (FRR) interprets kernel metrics as a combination of administrative distance and priority. The administrative distance is the upper byte and the priority is the lower 3 bytes. This specific metric value translates to `[255/8192]` in FRR notation.

High availability

The **Master HASync No Overwrite** setting helps prevent automatic data synchronization from the master to the slave TBF. When you enable this option, TBF creates the initial gateway configuration on the slave. TBF does not apply subsequent configuration updates from the master.

To maintain the gateway on both systems with different IP addresses or specific settings on each device, use this setting.

Additional resources

For a visual walkthrough of gateway configuration, watch this video tutorial: *Gateway Configuration Tutorial*

MPLS

Multiprotocol label switching (MPLS) uses labels to route data packets through the network.

Find the MPLS settings under **Routing > MPLS**.

MPLS routes are static routes that you define in TBF to control how data packets move through the network with labels instead of traditional IP address routing.

TBF supports the following types of MPLS routes:

- *Encapsulation* on page 219: This type adds MPLS labels to incoming packets.
- *Label switch* on page 220: This type forwards packets based on their MPLS labels.
- *Decapsulation* on page 220: This type removes MPLS labels from outgoing packets.

Enable or disable each MPLS route as needed. When you make changes, TBF automatically updates the routing table during the next route check cycle.

For automatic label distribution with protocols, see *LDP* on page 221.

Encapsulation

Encapsulation adds an MPLS label to a network route and forwards the route to a specified gateway.

Settings

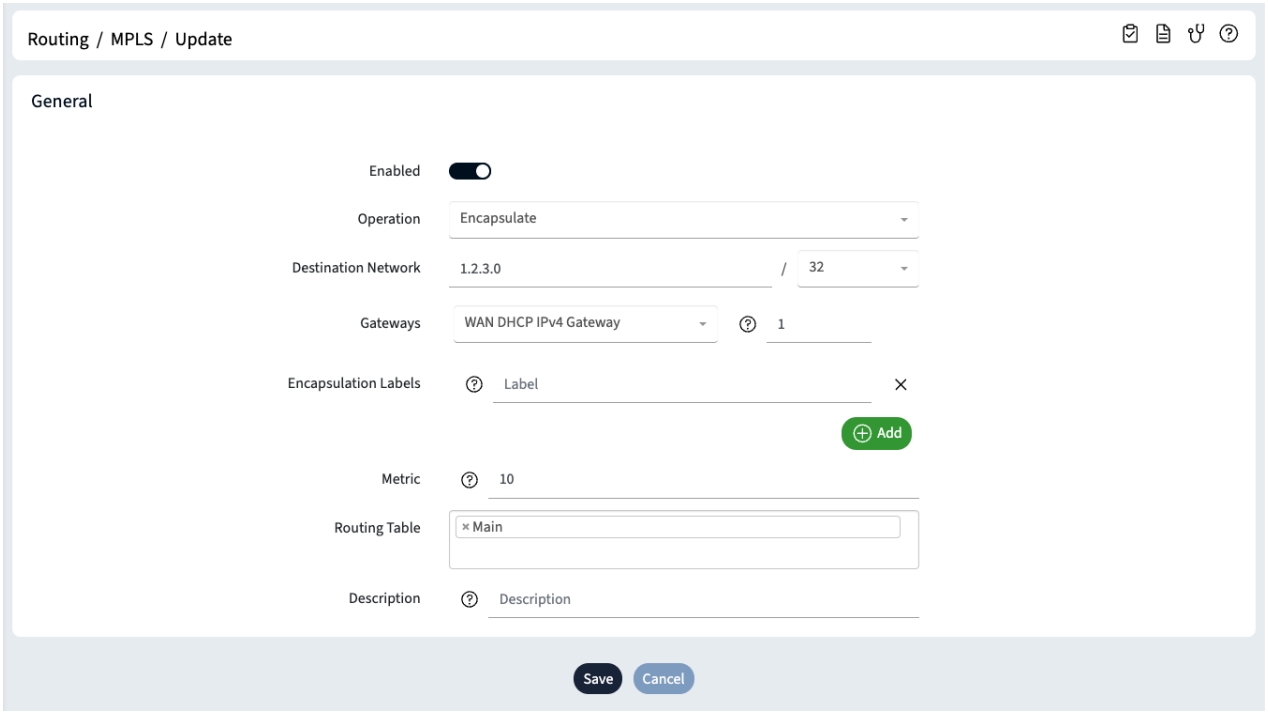


Figure 150: MPLS encapsulation

Destination Network

This field defines the target network for the route. This field is required.

Gateways

Select a gateway for the route.

Note: Set up only one gateway per route.

Encapsulation Labels

These labels are the MPLS labels that TBF applies to packets before TBF forwards the packets. You can specify multiple labels, and TBF applies the labels in the order you define.

Metric

This value determines route priority. You can create multiple routes to the same destination with different metric values. The route with the lowest metric value takes precedence and becomes the active route.

Routing Table

This setting is the routing table or tables where TBF stores the route. By default, TBF assigns new routes to the **Main** routing table with ID 254. To modify routing table assignments, see *routing tables* on page 211.

Note: TBF stores encapsulated routes in the standard routing tables alongside other routes.

Label switch

Label switch replaces one label with one or more other labels and forwards the packets to a gateway.

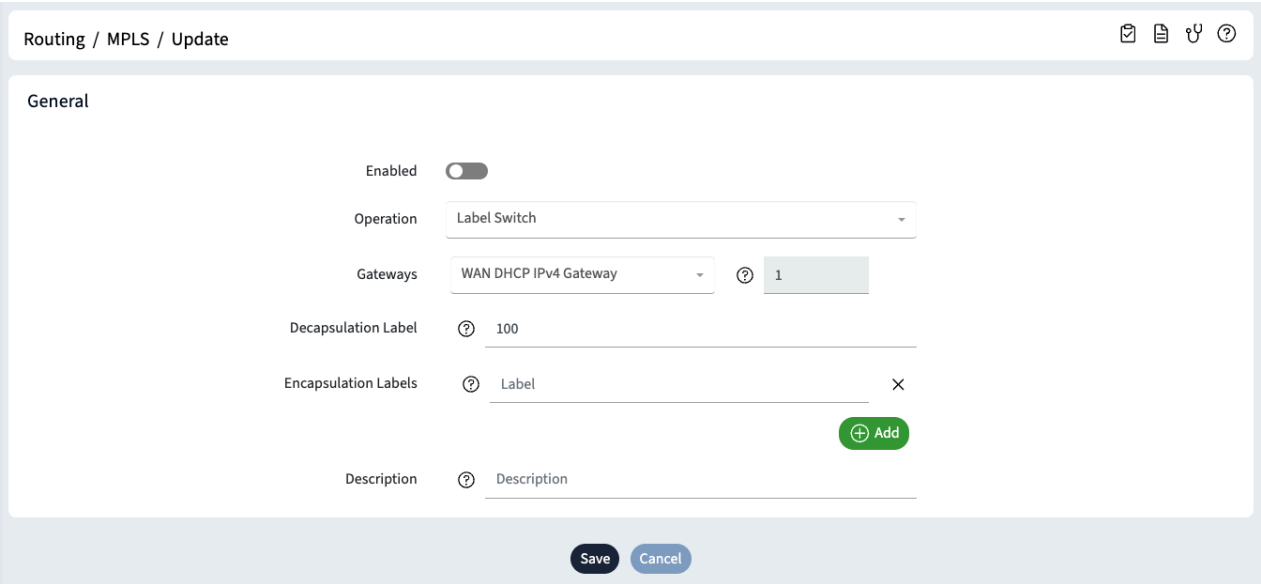


Figure 151: MPLS label switch

Settings

Gateways

This setting specifies the gateway where the system forwards packets. Set up only one gateway per label switch.

Decapsulation Label

This setting specifies the label that TBF removes from incoming packets before it applies new labels.

Encapsulation Label

This setting specifies the label that TBF applies to packets before it forwards them. You can define multiple labels, and TBF applies them in the order you specify.

Note: TBF automatically adds label switch routes to the MPLS routing tables.

Decapsulation

Decapsulation removes a label from a packet and forwards it to an interface or gateway.

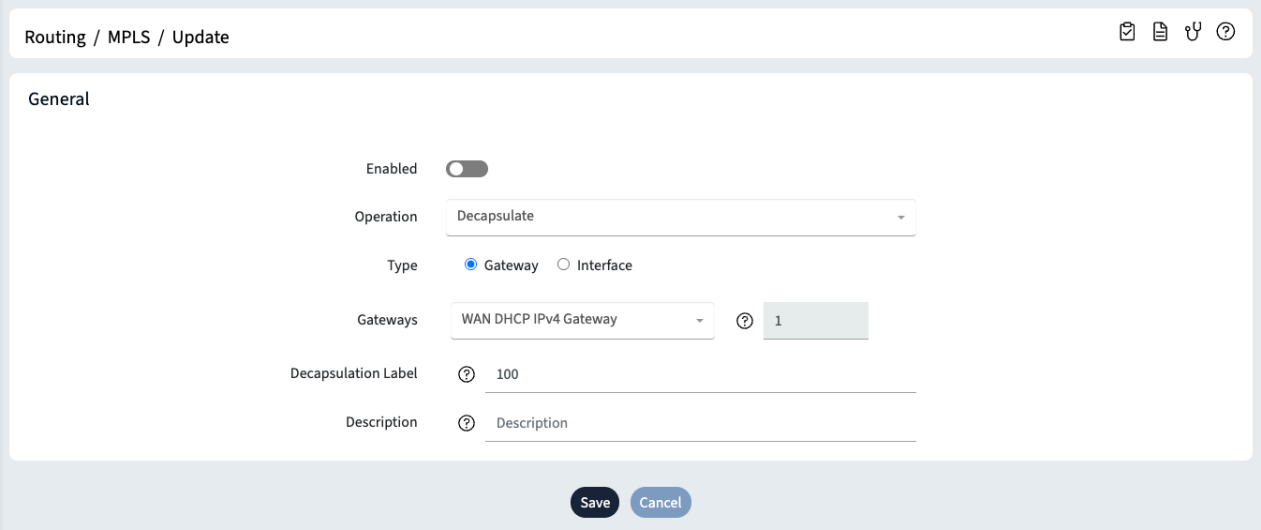


Figure 152: MPLS decapsulation

Settings

Gateways

This setting sets up a gateway for decapsulation. Set up only one gateway for each decapsulation rule.

Decapsulation Label

This setting specifies the MPLS label that TBF removes from incoming packets before it forwards them to their destination.

Note: TBF automatically adds decapsulation routes to the MPLS routing tables when you create a decapsulation rule.

LDP

The Label Distribution Protocol (LDP) enables MPLS routers to automatically exchange label mapping information with each other.

Fields

Enabled

Turns LDP on or off for this router.

Router ID

The system assigns this automatically when you leave the field empty. (Optional)

Interfaces

Select which network interfaces will participate in LDP.

Discovery Transport Address

The IP address that other routers use to establish LDP sessions with this router. Select this address from your available network objects.

Address Family

Choose between **IPv4** or **IPv6** addressing. The system uses **IPv4** by default.

Neighbors

Add specific LDP neighbors that this router should connect to. Each neighbor requires an IP address and can optionally use a password for authentication.

Dynamic

Dynamic routing in TBF automatically manages network paths with free range routing (FRR) software. TBF supports BGP, OSPF, OSPFv6, ISIS, and BFD protocols for automated routing decisions.

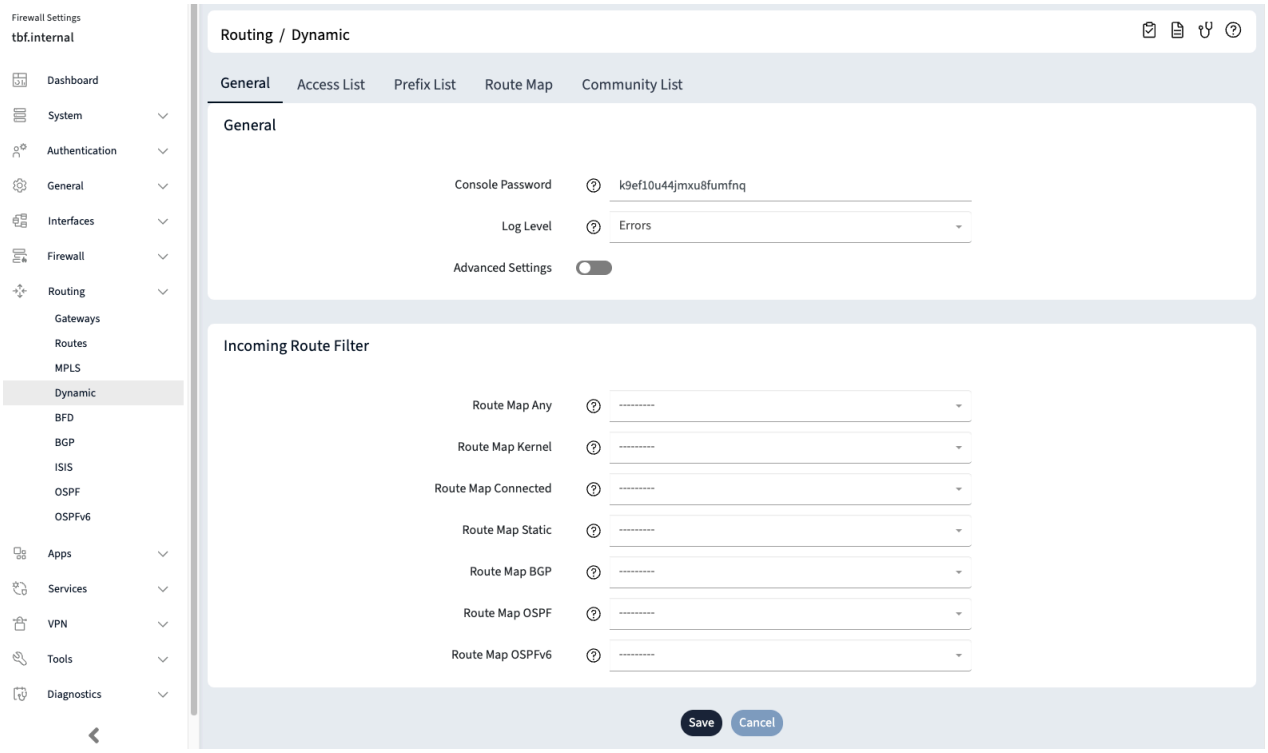


Figure 153: Dynamic

Set up dynamic routing in TBF to automatically manage network paths. TBF uses free range routing (FRR) software to handle these routing decisions behind the scenes.

For complete technical details, see the official FRR documentation: *FRR User Guide*.

TBF supports these dynamic routing protocols:

- *BGP* on page 224
- *OSPF* on page 229
- *OSPFv6* on page 233
- *ISIS* on page 227
- *BFD* on page 223

To set up dynamic routing, see these configuration topics:

- *General settings* on page 222
- *Administrative distance* on page 223

General settings

General settings control console access and custom configuration options for the routing system.

For security purposes, set a console password. While you do not need this password for regular operations, leaving the console unprotected creates potential security vulnerabilities.

To access FRR from the console or bash shell, use the `vtysh` command. This connection requires no password.

You can define **Custom Options** in the **Advanced Settings** section. TBF automatically adds these options to the configuration file at the top level.

Administrative distance

This page describes how administrative distance determines route selection priority.

Administrative distance enables the dynamic routing daemon (FRR) to determine which routes to install in the Routing Information Base (RIB) based on the source protocol. TBF prioritizes routes with the lowest administrative distance value. These distance values follow industry-standard conventions that other routing platforms use.

Protocol	Distance
System	0
Kernel	0
Static	1
EBGP	20
OSPF	110
ISIS	115
IBGP	200

TBF adds routes based on their administrative distance. When multiple routes have the same administrative distance,TBF uses the route metric to determine the preferred path.

TBF adds all routes to the **Main** routing table with a fixed metric value of 20. TBF does not use the internal metric that the dynamic routing daemon maintains for route installation. This internal metric serves only for internal calculations and route redistribution to other routing peers.

BFD

Bidirectional forwarding detection (BFD) is a network protocol that quickly detects communication failures between two routers or network devices.

BFD offers rapid failure detection between network devices, helps maintain network stability and enables faster convergence when link failures occur. This protocol sends periodic control packets between devices to monitor connectivity status.

Find the BFD settings under **Routing > BFD**.

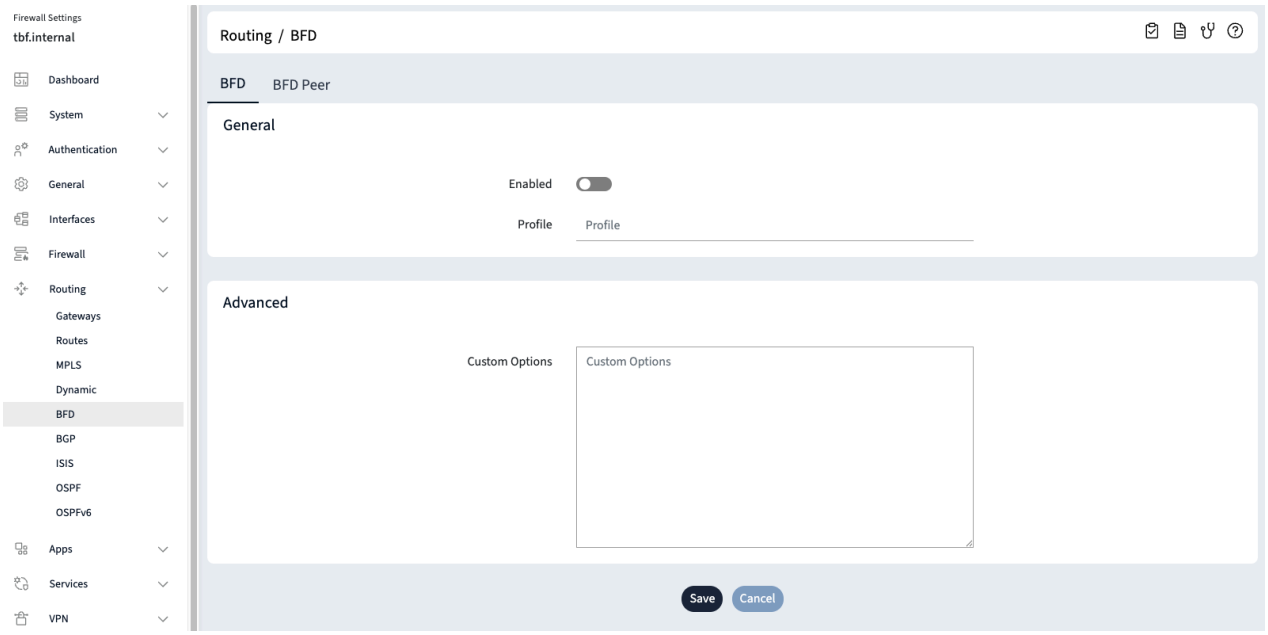


Figure 154: BFD

Use the **Profile** setting to define specific BFD parameters that match the network requirements. These profiles contain timing intervals, multipliers, and other detection parameters that determine how quickly BFD identifies link detected failures.

Use **Custom Options** to add additional configuration parameters that the standard profile settings do not cover. This helps provide flexibility to fine-tune BFD behavior for specific network conditions.

To set up connections with other network devices, see how to *set up a BFD peer* on page 224.

Set up a BFD peer

To monitor communication with a remote device and detect connection failures quickly, set up a bidirectional forwarding detection (BFD) peer.

To monitor the connection between the device and a remote device, set up a BFD peer. BFD helps detect network failures faster than traditional routing protocols, and improves network reliability and response times.

Perform the following steps:

1. Navigate to **Routing > BFD > BFD Peer**.

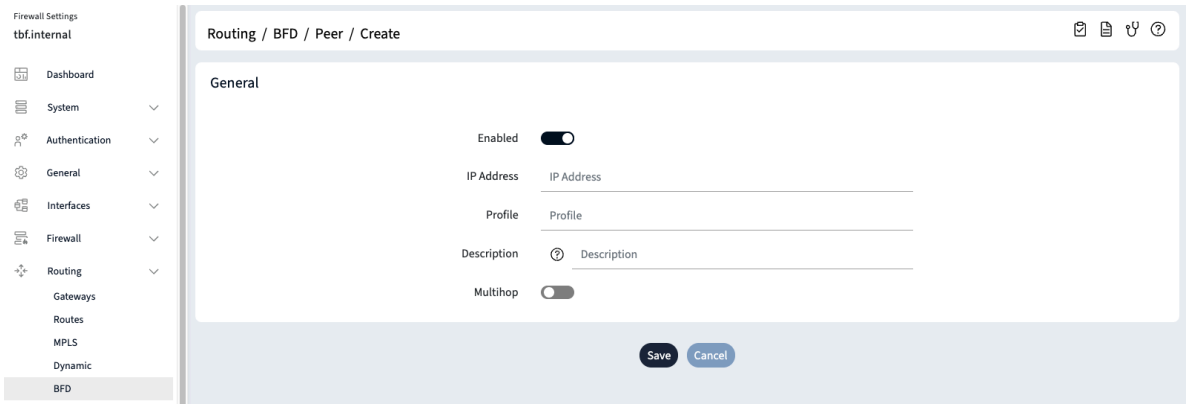


Figure 155: Create peer

2. Enter the **IP Address** of the remote device you want to monitor.
   This is the IP address of the device you want to establish a BFD session with.

3. If you need custom settings that differ from the default BFD profile, then select a **Profile**.
   The default profile works for most scenarios. If you need specific timing or detection parameters, then select a different profile.

4. If the BFD session must pass through multiple network devices to reach the peer, then enable **Multihop**.
   If the peer device does not connect directly to the network segment, then use this option.

BGP

To enable routing between different autonomous systems on the network, set up Border Gateway Protocol (BGP) settings.

Find the BGP settings under **Routing > BGP**.

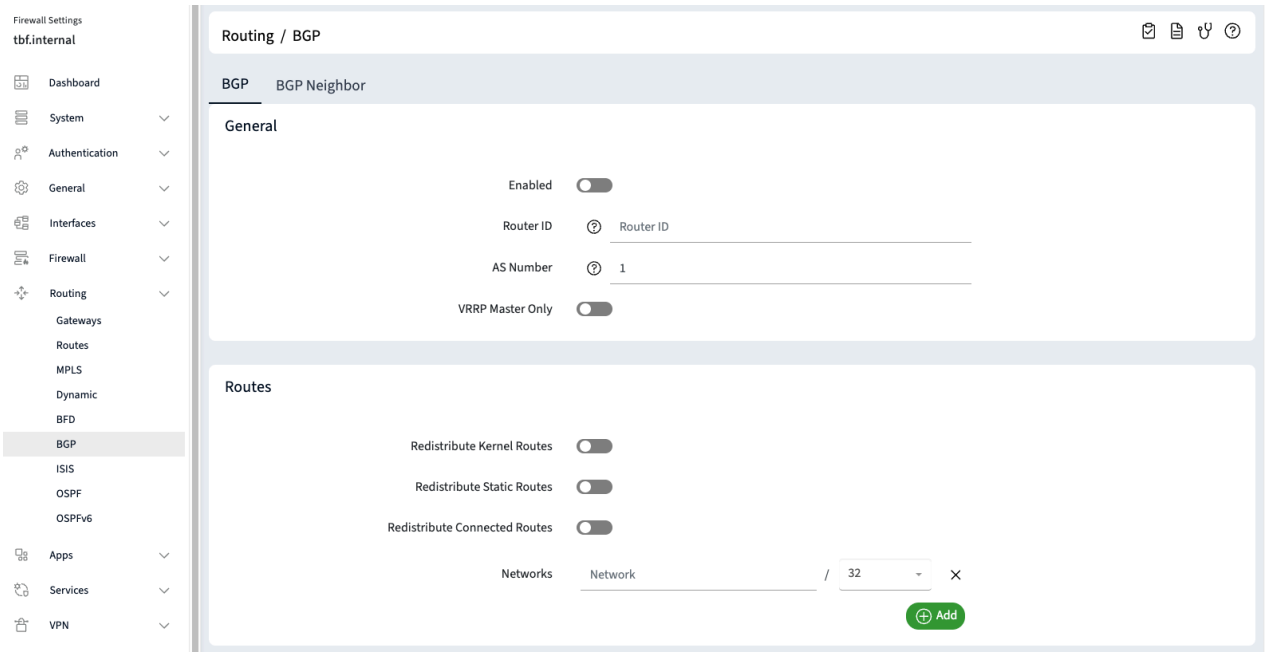


Figure 156: BGP

BGP is the standard protocol that manages routing between different autonomous systems on the internet. RFC 1771 defines the current version, BGP-4, which allows networks operated by different organizations to exchange routing information and communicate effectively.

Settings

Router ID

Set a unique identifier for the router that other BGP peers recognize. You can specify this manually or TBF assigns one automatically.

AS Number

Enter the autonomous system number that identifies the network. Verify that this number matches the officially assigned AS number and is unique within the autonomous system.

Redistribute Kernel Routes

To automatically share kernel routes with the BGP neighbors, enable this option. This includes routes that the operating system kernel manages.

Redistribute Static Routes

To automatically share manually set up static routes with the BGP neighbors, enable this option.

Redistribute Connected Routes

To automatically share routes for directly connected network interfaces with the BGP neighbors, enable this option.

Networks

To advertise to BGP peers beyond those that the redistribution options above share, specify additional networks. Add multiple networks as required.

Custom Options

Enter any additional configuration parameters that the standard options above do not cover.

Note: TBF classifies static routes as kernel routes.

Related topics

For BGP neighbor configuration, see *BGP neighbor* on page 226.

BGP neighbor

Set BGP neighbor settings to establish peering relationships with other routers.

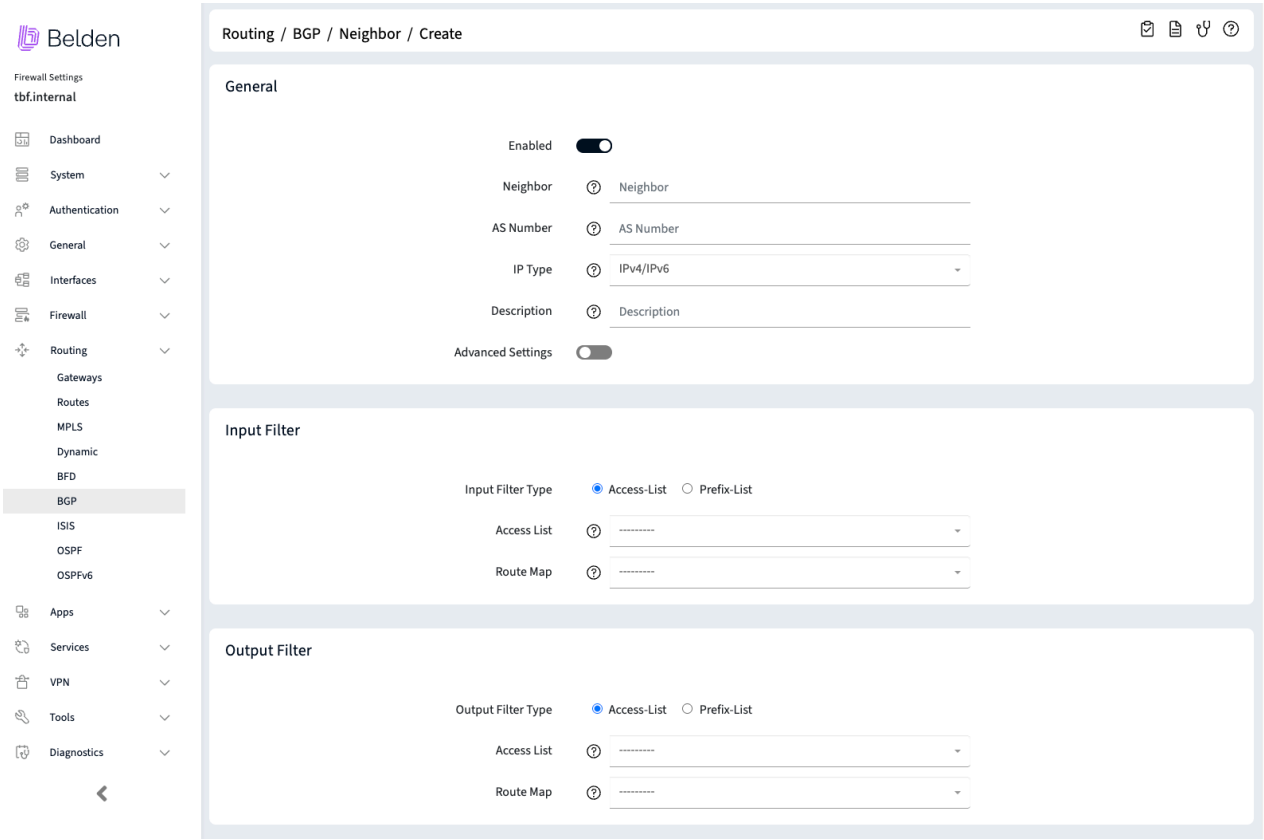


Figure 157: BGP neighbor

Create each BGP peer that you want to communicate with as a BGP neighbor under **Routing > BGP > BGP Neighbor**. Each BGP neighbor represents a BGP peering relationship with another router.

Settings

Neighbor

Specify the IP address of the BGP neighbor you want to establish a peering relationship with.

AS Number

Specify the autonomous system number of the BGP neighbor.

IP Type

Determine which IP protocol versions this BGP neighbor uses for route advertisements. Options include IPv4 only, IPv6 only, or both protocols.

Advanced Settings

Customize the neighbor relationship with additional parameters. See the FRR manual for a detailed explanation: *FRR BGP*.

Input Filter

Apply filtering to routes you receive from this BGP neighbor with an *access list* on page 236, *prefix list* on page 236, or *route map* on page 237. This controls which routes you accept into the routing table.

Output Filter

Apply filtering to routes you advertise to this BGP neighbor with an *access list* on page 236, *prefix list* on page 236, or *route map* on page 237. This controls which routes you send to the BGP neighbor.

Custom Options

Enter any additional configuration parameters that the standard options above do not cover.

ISIS

Intermediate System to Intermediate System (ISIS) is a link-state routing protocol that efficiently distributes routing information within a network.

Find the ISIS settings under **Routing > ISIS**.

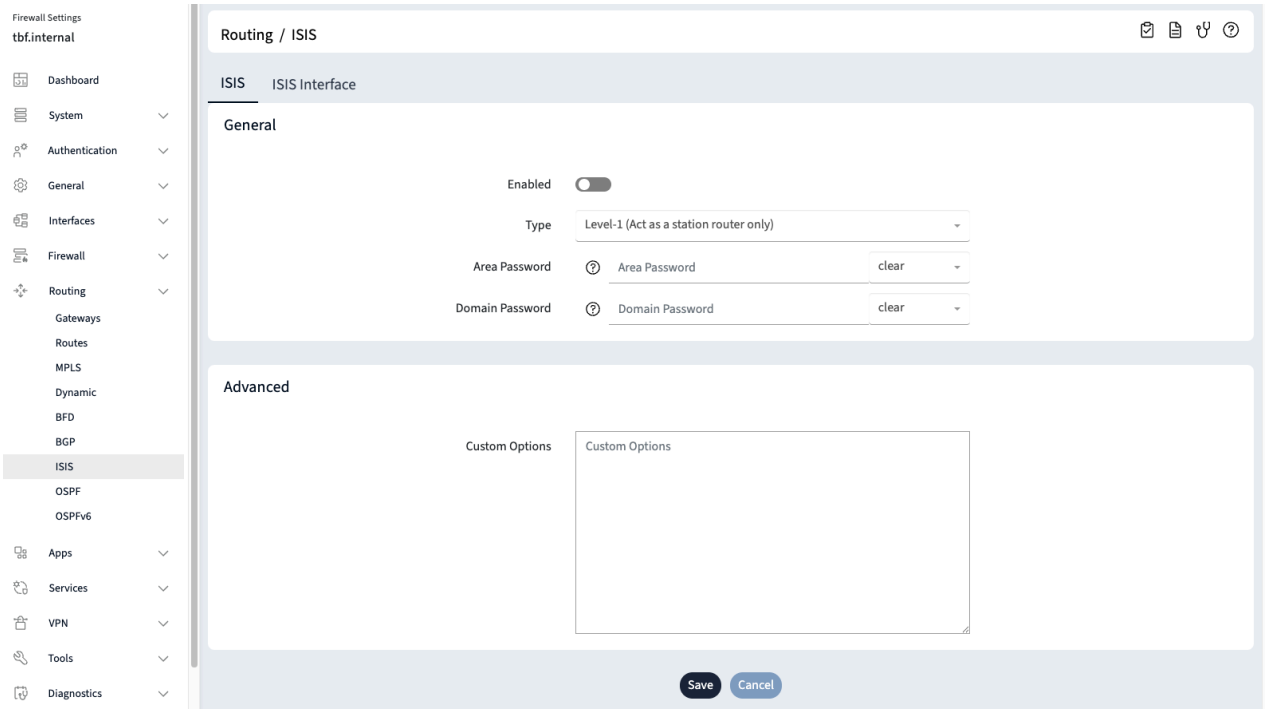


Figure 158: ISIS

Overview

ISIS creates a complete topology map of the network. This map enables routers to calculate the efficient paths to reach their destinations. As a link-state protocol, ISIS maintains detailed information about network connections and updates this information when changes occur.

Settings

Area Password

This field sets the authentication password for an ISIS area. This password helps secure routing communications within the area.

Area Password Type

This setting determines how TBF encrypts the area password. Select from:

- **clear**: This option is for unencrypted text. This is the default.
- **md5**: This option is for MD5 hash encryption.

Domain Password

This field sets the authentication password for domain-level routing communications. This password helps secure routing information exchanges between different ISIS areas within the same domain.

Domain Password Type

This setting determines how TBF encrypts the domain password. Select from:

- **clear**: This option is for unencrypted text. This is the default.
- **md5**: This option is for MD5 hash encryption.

Type

This setting defines how the router operates within the ISIS hierarchy. Select from:

- **Level-1 (Act as a station router only)**: This option routes traffic only within the local area.
- **Level-1-2 (Act as both a station router and an area router)**: This option routes traffic within the local area and between different areas.
- **Level-2-Only (Act as an area router only)**: This option routes traffic only between different areas.

Custom Options

This field adds custom parameters for advanced ISIS settings that standard options do not cover.

Related topics

For interface-specific ISIS settings, see *ISIS interface* on page 228.

ISIS interface

Set up ISIS interface settings to control how the ISIS routing protocol works on specific network interfaces.

To set up an ISIS interface, navigate to **Routing > ISIS > ISIS Interface**.

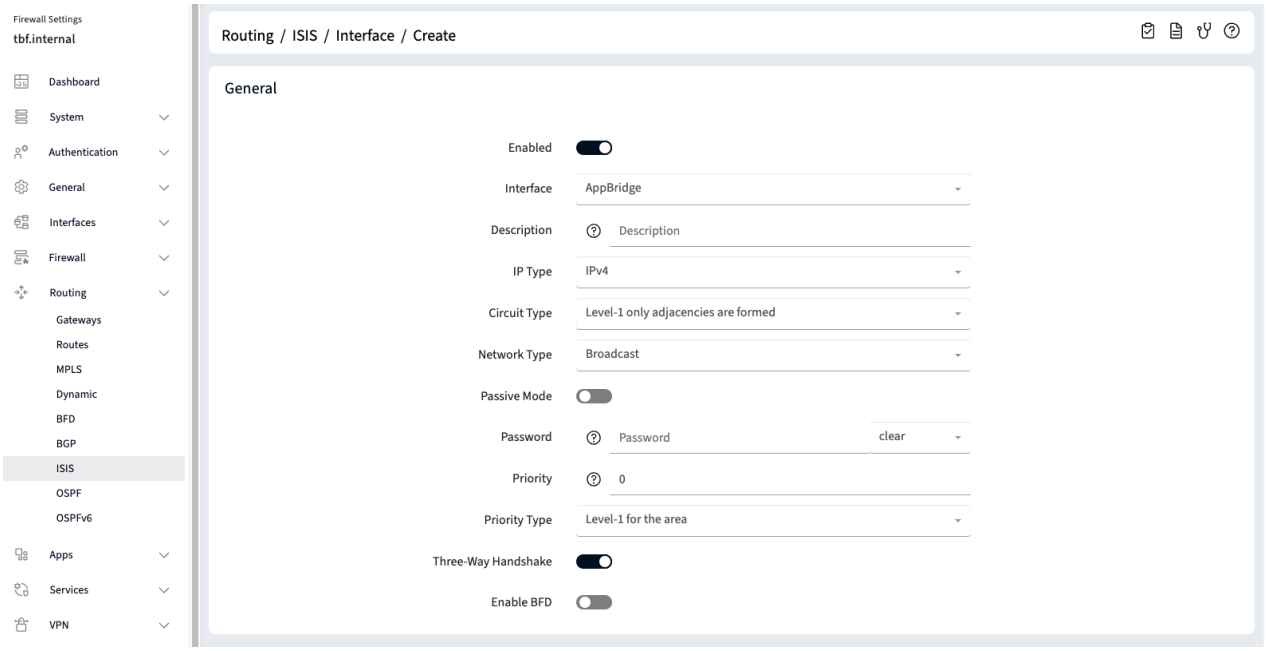


Figure 159: ISIS interface

Settings

Interface

Select the network interface that uses ISIS routing.

IP Type

Select the IP protocol version: **IPv4** or **IPv6**.

Circuit Type

Set the level of ISIS connections this interface establishes with neighboring routers. Select from these options:

- **Level-1 only adjacencies form**: This option connects only with routers in the same area.
- **Level-1-2 adjacencies form**: This option connects with routers in the same area and other areas.
- **Level-2 only adjacencies form**: This option connects only with routers in other areas.

Network Type

Select the physical network type. Use **Broadcast** for multi-access networks like Ethernet, or **Point-to-Point** for direct connections between two routers.

Passive Mode

Enable this option to make the interface passive. The interface does not send ISIS hello packets but still advertises the connected networks.

Password

To set up authentication, enter a password and select whether to use cleartext or MD5 encryption.

Priority

Set the priority value for designated router election on broadcast networks. Higher values make this router more likely to become the designated router.

Priority Type

Select which ISIS level the priority setting affects:

- **Level-1 for the area**: Priority applies to designated router election within the area.
- **Level-2 for the domain**: Priority applies to designated router election between areas.

Three-Way Handshake

Enable this option to use a three-way handshake process to establish point-to-point connections, which helps provide more reliable adjacency formation.

Enable BFD

Enable bidirectional forwarding detection (BFD) to quickly detect link errors and improve network convergence time.

Custom Options

Enter additional custom parameters for advanced ISIS interface settings not covered by the standard options.

OSPF

OSPF version 2 is a routing protocol for IPv4 networks that offers better scalability and faster convergence than RIP.

Find the OSPF settings under **Routing > OSPF**.

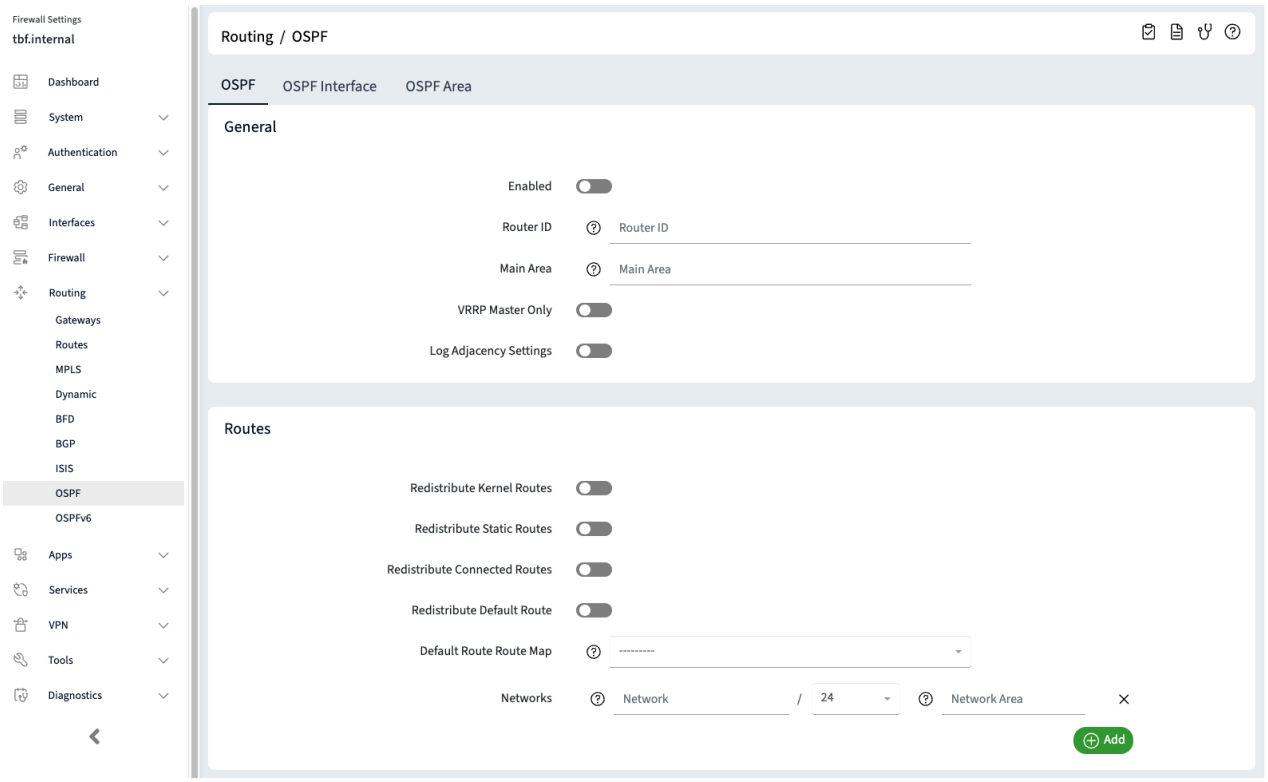


Figure 160: OSPF

OSPF version 2 is a routing protocol that RFC 2328 defines. OSPF operates as an interior gateway protocol (IGP) that designers created for IPv4 networks. Unlike RIP, OSPF offers better scalability and faster convergence when network changes occur. These benefits make OSPF the preferred choice for large-scale networks, including internet service provider (ISP) backbones and enterprise environments.

General settings

Router ID

This option sets a unique identifier for this router that other OSPF routers will recognize. You can manually assign an ID or allow the system to generate one automatically.

Main Area

This option defines the primary OSPF area that this router instance will participate in. Areas help organize and optimize routing within the OSPF network.

Redistribute Kernel Routes

When enabled, this option automatically shares kernel-level routes with other OSPF routers in the network.

Redistribute Static Routes

When enabled, this option automatically shares manually configured static routes with other OSPF routers.

Redistribute Connected Routes

When enabled, this option automatically shares routes for directly connected network interfaces with other OSPF routers.

Default Route Route Map

This setting applies filtering rules to the default route with a *route map* on page 237. This filter controls how the system advertises the default route to other routers.

Networks

This option specifies which network interfaces participate in OSPF routing. Only interfaces that match the defined networks will advertise their routes.

Custom Options

This option provides a field for advanced users to add specialized configuration parameters not covered by the standard settings.

Related configuration

For detailed configuration options, see:

- *OSPF interface* on page 231: Set up OSPF interface settings
- *OSPF area* on page 232: Set up OSPF area settings

Notes

Note: Keep in mind that TBF treats static routes as kernel routes in the system.

Note: Remember that OSPF only advertises networks where the host system has an assigned IP address. You cannot advertise networks that are not directly connected to the system.

OSPF interface

To include specific network interfaces in OSPF routing advertisements, set up OSPF interfaces.

Define each OSPF interface that you want to advertise individually. To set up these interfaces, navigate to **Routing > OSPF > OSPF Interface**.

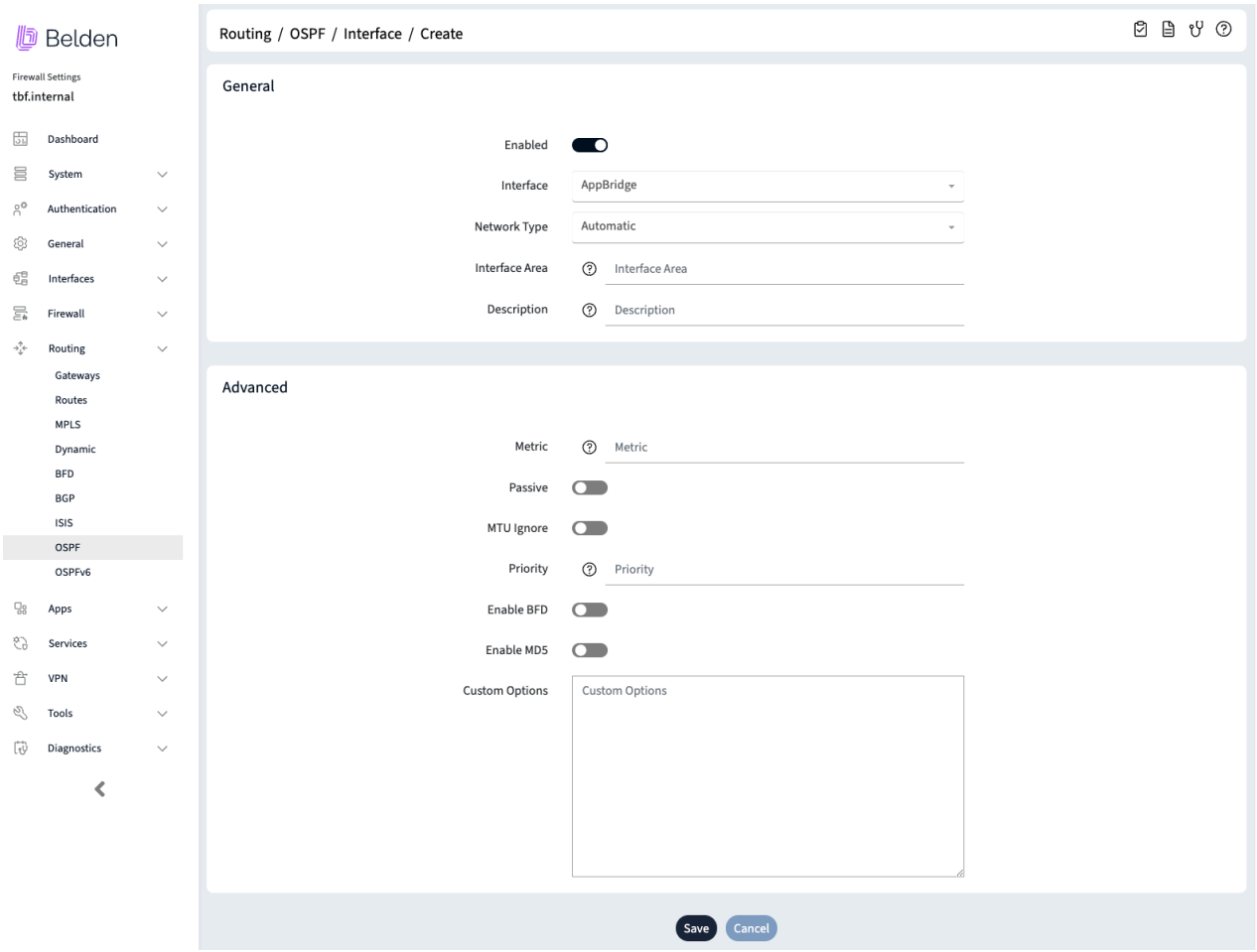


Figure 161: OSPF settings

Settings

Interface

This setting selects the network interface to use for OSPF. Assign each interface only once.

Network Type

TBF automatically detects the network type. You can manually override this setting if necessary.

Interface Area

This setting assigns the interface to the appropriate OSPF area. This setting is required.

Metric

This field sets a custom cost value for this interface. This overrides the default metric calculation.

Passive

This setting sets this interface to stop sending OSPF packets but continue to advertise its routes through other interfaces. The interface does not discover OSPF neighbors.

MTU Ignore

This setting bypasses MTU size conflicts between OSPF neighbors

Priority

This field sets the priority level for designated router (DR) election. Higher values increase the chance of becoming the DR.

Enable BFD

This setting is for faster inoperability detection between OSPF neighbors.

Enable MD5

This setting assists in establishing security for OSPF communications with encrypted passwords.

MD5 Password

This field specifies the authentication password for this interface. The password requires 16 characters or fewer. TBF automatically assigns key ID 1.

Custom Options

This field adds any additional OSPF parameters that the standard settings do not cover.

OSPF area

To segment the network, define additional OSPF areas beyond the default backbone area.

To segment the network with additional OSPF areas, navigate to **Routing > OSPF > OSPF Area**. The default backbone area from the main OSPF page runs in normal mode and does not need configuration here.

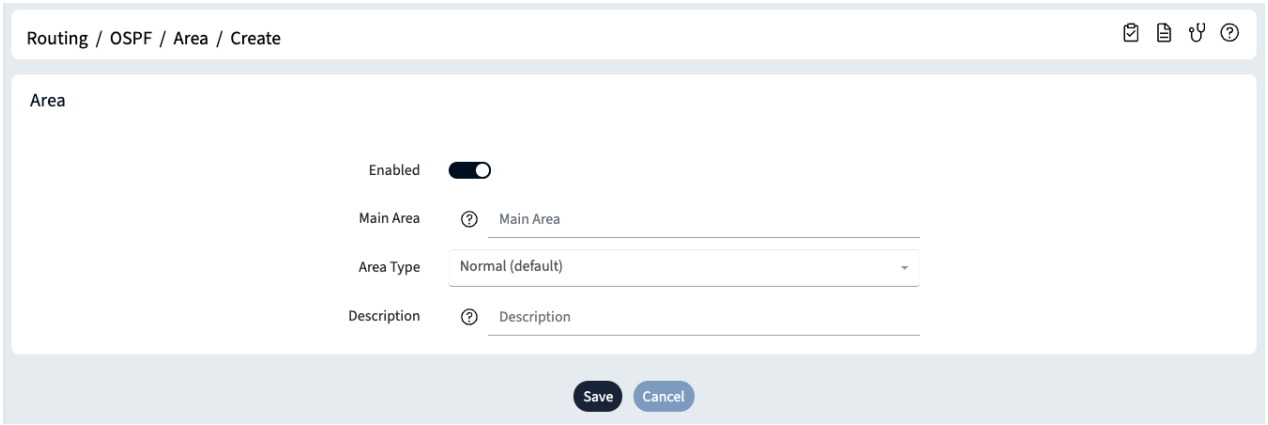


Figure 162: OSPF area

Settings

Main Area

This field sets the unique area ID for this OSPF area. The backbone area uses `0.0.0.0` as its standard identifier.

Area Type

This setting controls how the OSPF area operates. Select from:

- **Normal (default)**: This option is a standard OSPF area that accepts all LSA types.
- **Stub Area (stub)**: This option blocks external LSAs to reduce the routing table size.
- **Totally Stub (stub no-summary)**: This option blocks both external and summary LSAs to simplify the configuration.
- **Not so Stub Area (nssa)**: This option helps allow limited external routes while blocking others.

Description

To help identify and manage the OSPF area, type a descriptive name.

OSPFv6

OSPF version 6 (OSPFv6) is a routing protocol designed specifically for IPv6 networks that automatically manages routing information between network devices.

Find the OSPFv6 settings under **Routing > OSPFv6**.

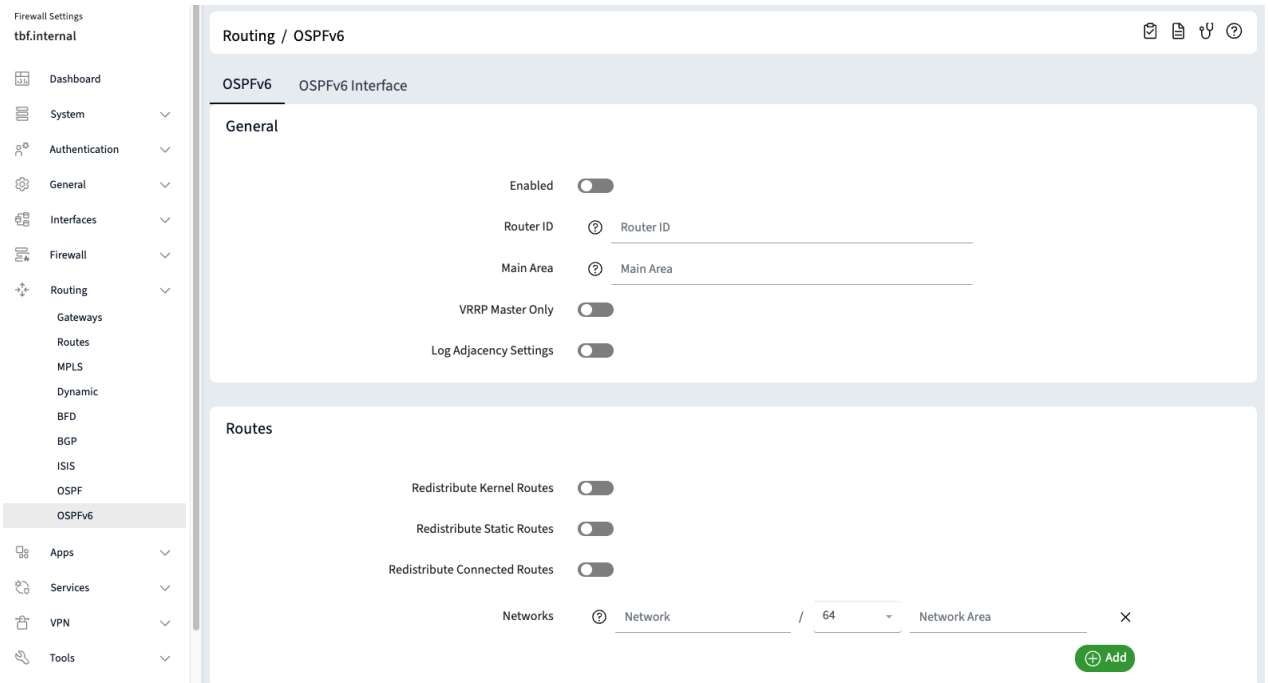


Figure 163: Ospfv6 settings

Overview

RFC 2740 defines OSPFv6 as a link-state routing protocol. Unlike its predecessor OSPFv4, developers built OSPFv6 specifically for IPv6 networks. This protocol enables routers to automatically discover network topology and calculate the efficient paths for data transmission. The protocol maintains a database of network links and uses this information to build routing tables dynamically.

Note: TBF does not fully implement OSPFv6 and lacks some features.

Settings

Router ID

This option sets a unique router identifier that is advertised to other routers, or allows TBF to automatically generate one.

Main Area

This option defines the primary area for this OSPF instance. Areas help organize the network topology and reduce routing overhead.

Redistribute Kernel Routes

This option automatically shares kernel routes with other routers in the OSPF domain.

Redistribute Static Routes

This option automatically shares static routes with other routers in the OSPF domain.

Redistribute Connected Routes

This option automatically shares directly connected network routes with other routers in the OSPF domain.

Networks

This option defines which network interfaces are advertised when they belong to the specified network range.

Custom Options

This option allows you to add custom options for advanced settings not covered by the standard options.

Note: Keep in mind that TBF treats static routes as kernel routes in the system.

Note: OSPFv6 only advertises networks where the host system has an assigned IP address. You cannot advertise arbitrary networks.

Related topics

For interface-specific configuration options, see the *OSPFv6 interface* on page 234.

OSPFv6 interface

To control how the network interfaces participate in OSPFv6 routing, set OSPFv6 interface parameters.

To include OSPFv6 interfaces in routing advertisements, set up each interface. To access the OSPFv6 settings, navigate to **Routing > OSPFv6 > OSPFv6 Interface**.

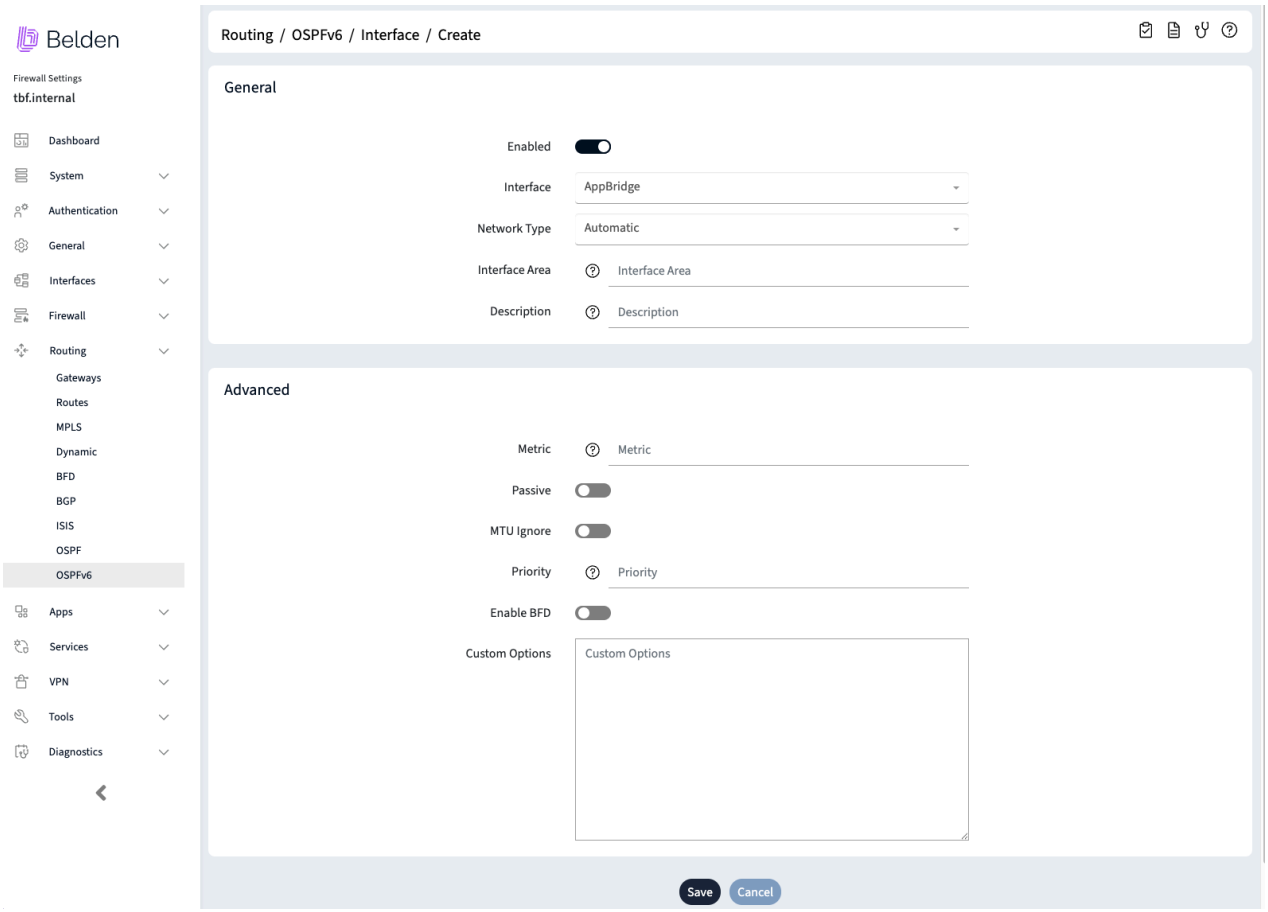


Figure 164: Ospfv6 interface

Settings

Interface

This setting selects the network interface to use for OSPF. Assign each interface only once.

Network Type

TBF automatically detects the network type. You can manually override this setting if necessary.

Interface Area

This setting assigns the interface to the appropriate OSPF area. This setting is required.

Metric

This field sets a custom cost value for this interface. This overrides the default metric calculation.

Passive

This setting sets this interface to stop sending OSPF packets but continue to advertise its routes through other interfaces. The interface does not discover OSPF neighbors.

MTU Ignore

This setting bypasses MTU size conflicts between OSPF neighbors

Priority

This field sets the priority level for designated router (DR) election. Higher values increase the chance of becoming the DR.

Enable BFD

This setting is for faster inoperability detection between OSPF neighbors.

Enable MD5

This setting assists in establishing security for OSPF communications with encrypted passwords.

MD5 Password

This field specifies the authentication password for this interface. The password requires 16 characters or fewer. TBF automatically assigns key ID 1.

Custom Options

This field adds any additional OSPF parameters that the standard settings do not cover.

Access list

Access lists are network filtering tools that control traffic based on IP address ranges.

Find the access list settings under **Routing > Dynamic > Access List**.

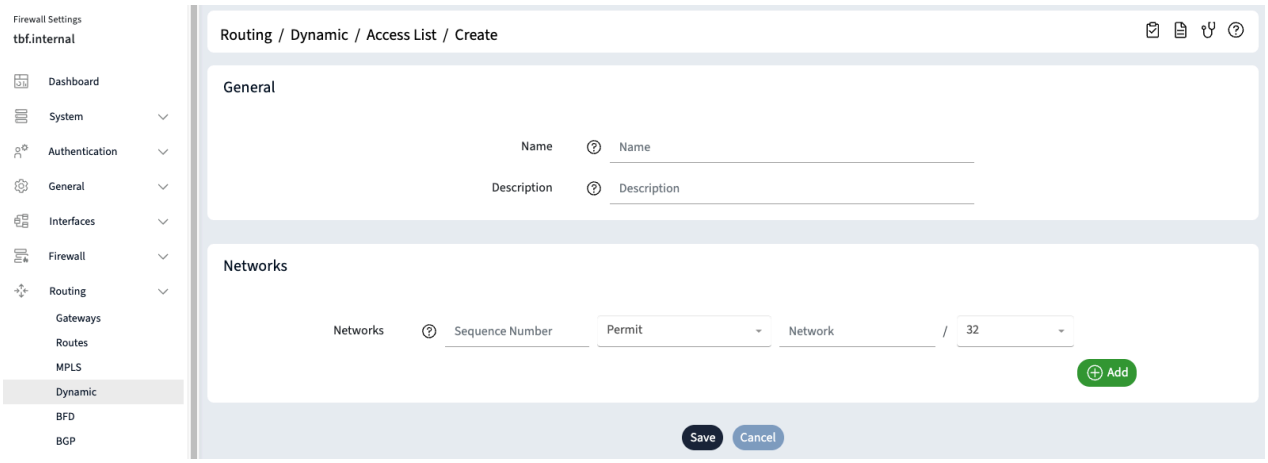


Figure 165: Access list

Access lists help you manage network security levels by defining which IP addresses or IP address ranges access the network and which you block.

To create an access list, enter a unique name and specify the IP addresses or IP address ranges that serve as the filtering rules.

The **Sequence Number** sets the priority order for processing access list rules. TBF processes lower sequence numbers first.

Prefix list

A prefix list is an advanced filtering mechanism that controls network traffic based on IP address ranges and prefix lengths.

Find the prefix list settings under **Routing > Dynamic > Prefix List**.

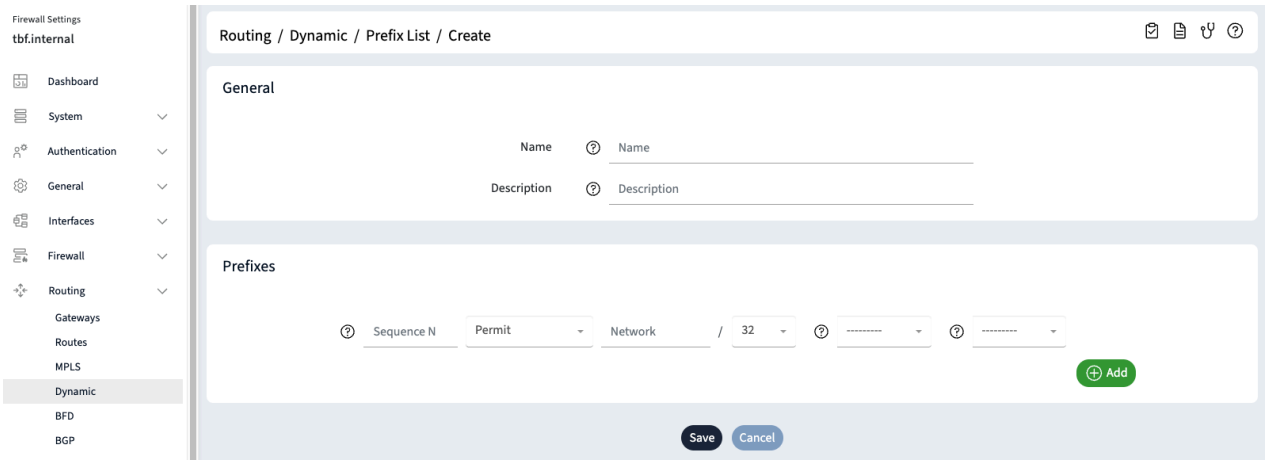


Figure 166: Prefix list

Prefix lists have an advanced filtering mechanism that controls network traffic based on IP address ranges. Prefix lists offer more precise control than basic access lists because they allow you to filter based on both IP addresses and their associated prefix lengths.

Each prefix list requires a unique name and contains a list of IP addresses or IP address ranges that define the filtering criteria.

Sequence Number

This setting determines the processing order of the entries. TBF processes lower sequence numbers first.

Action

This setting determines whether the TBF allows or blocks traffic that matches the criteria. To allow traffic, select **Permit**, or to block traffic, select **Deny**. To create complex filtering rules, combine both actions within the same prefix list.

Minimum Netmask

This setting defines the minimum prefix length for matching IP address ranges. The TBF applies the prefix rule when the prefix length is greater than or equal to this `ge` value. This number must be larger than the base network mask of the specified IP address.

Maximum Netmask

This setting defines the maximum prefix length for matching IP address ranges. TBF applies the prefix rule when the prefix length is less than or equal to this `le` value. This number must be smaller than or equal to 32 for IPv4 addresses.

When you create a prefix list and you do not specify the **Minimum Netmask** or **Maximum Netmask** values, the prefix list functions similarly to a standard *access list* on page 236 and matches only exact IP addresses or networks.

Route map

Route maps have an advanced filtering mechanism that uses different options for routing control.

Location

Find the route map settings under **Routing > Dynamic > Route Map**.

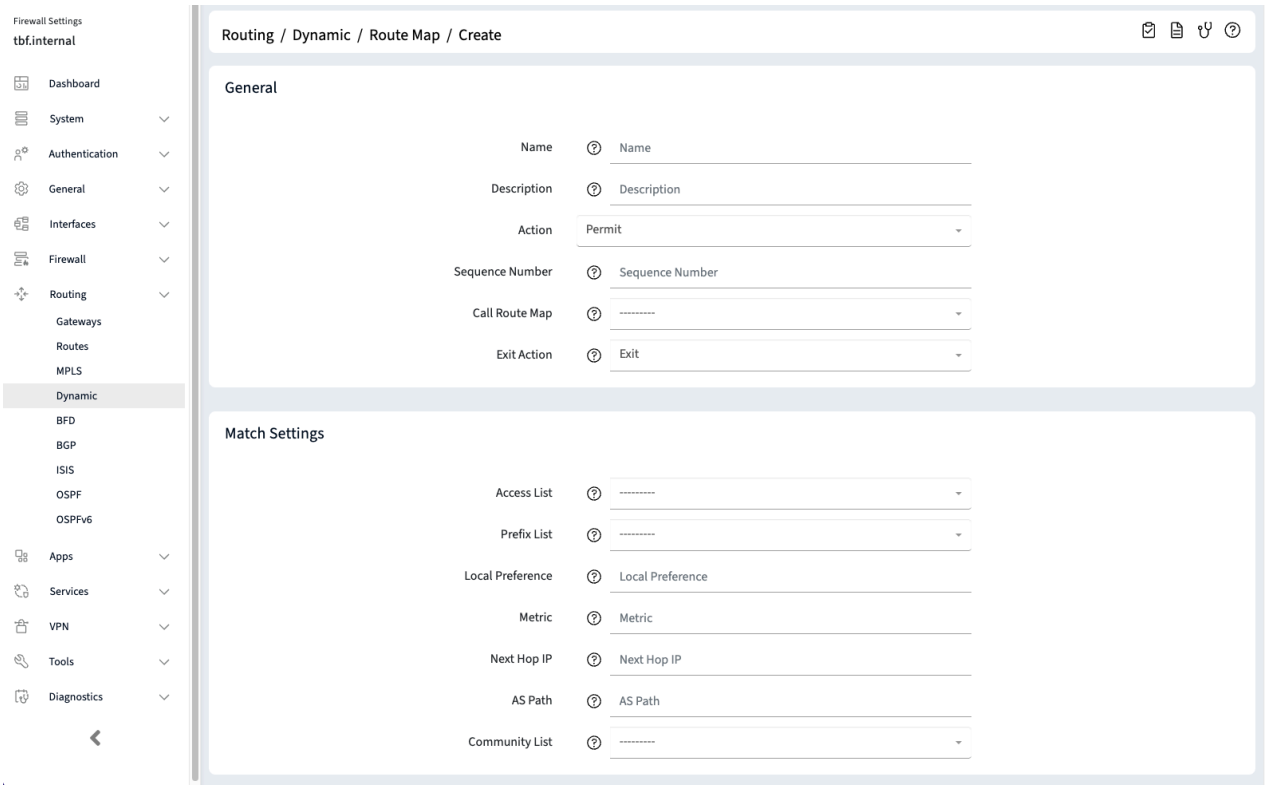


Figure 167: Route map

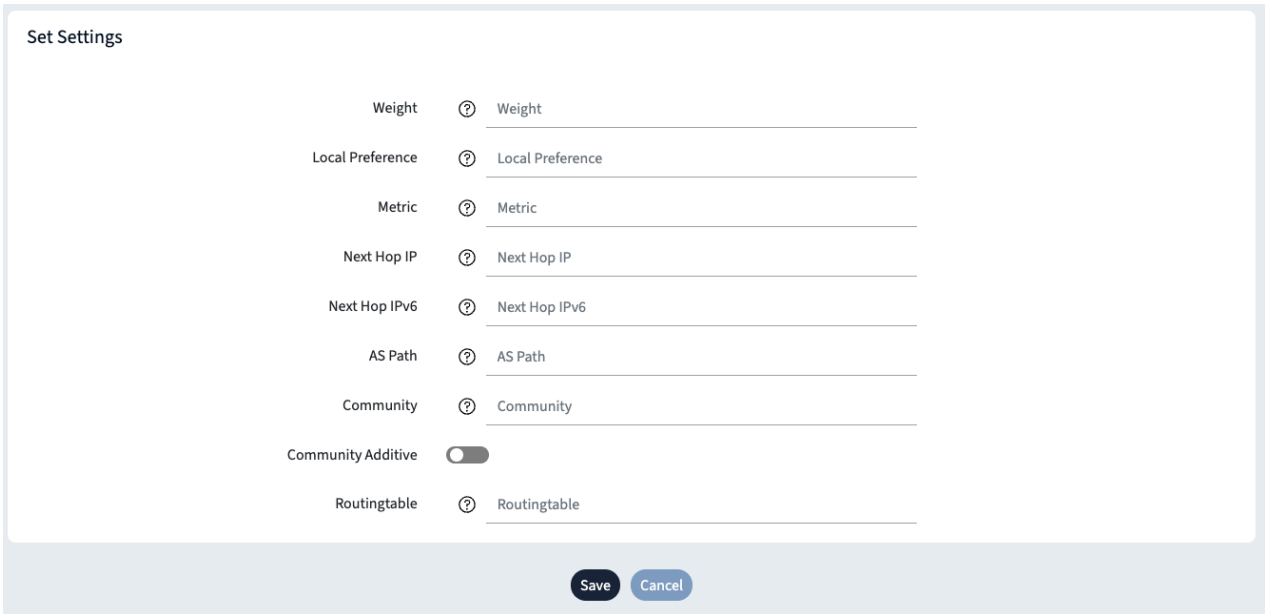


Figure 168: Route map set settings

Overview

Route maps offer sophisticated filtering capabilities that help ensure you control routing behavior with various configuration options. These tools help you manage how the system processes and distributes routes throughout the network.

Integration capabilities

To create complex routing policies, chain multiple route maps together or integrate them with *access lists* on page 236 and *prefix lists* on page 236.

Route maps also enable you to modify route attributes for both incoming and outgoing traffic, such as the **Weight** or **Metric** values to influence path selection.

Additional resources

For detailed technical information and advanced configuration examples, see the FRR documentation: *FRR routemap*.

Use cases

This topic describes common routing scenarios and configuration examples for network setup and management.

This section provides practical examples for common routing scenarios to implement effective network configurations.

Available use cases

- *Multi-gateway setup* on page 239: Set up multiple gateways for automatic failover switching when connections become unavailable.
- *Policy routing* on page 241: Route different types of traffic through specific ISP connections using routing tables and firewall rules.

Multi-gateway setup

To help provide automatic failover switching when one connection becomes unavailable, set up multiple gateways for an interface.

This setup helps provide automatic failover switching when one connection becomes unavailable.

Perform the following steps:

1. To view the gateway configurations, navigate to **Routing > Gateways**.

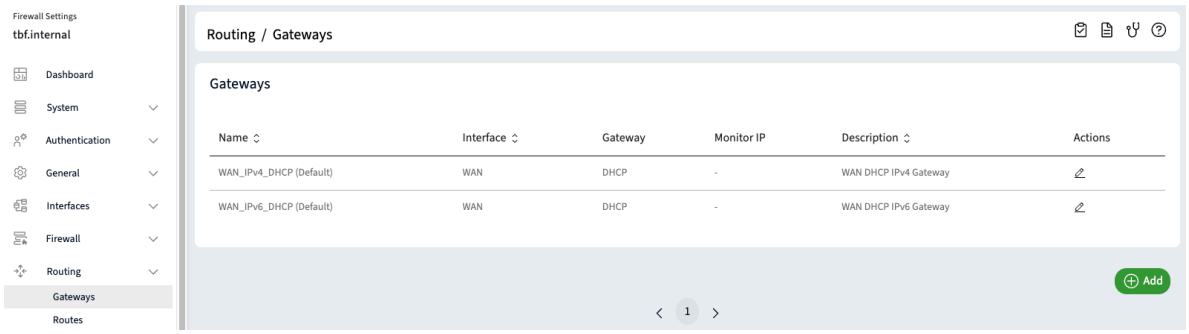


Figure 169: Overview of available gateways

2. If you need to create new gateways, select **Add**.

   a. Enter the **Gateway IP** address.

   b. If this gateway functions as the primary connection, enable **Default Gateway**.

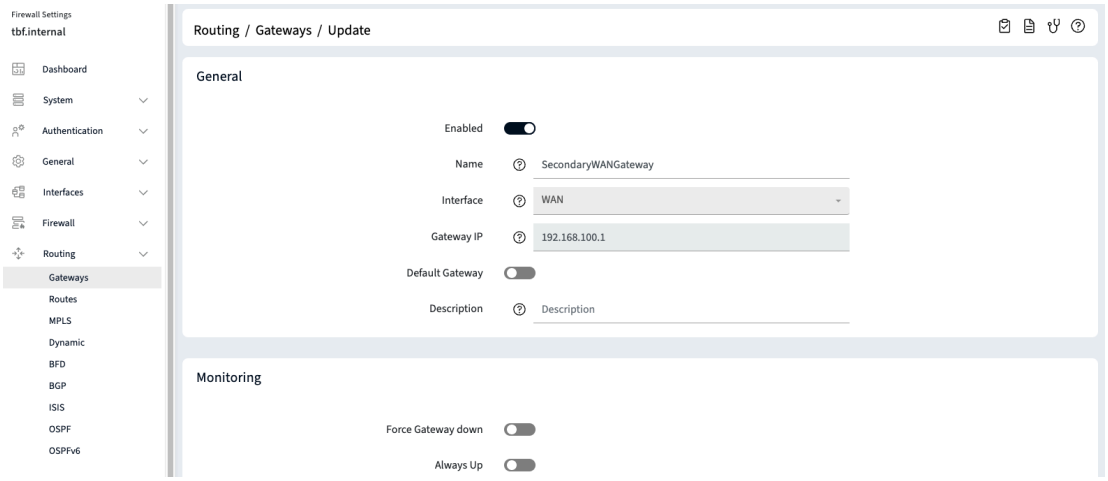


Figure 170: Gateway configuration

3. In the **Monitoring** section, disable **Always Up**.
   When enabled, TBF assumes the gateway is operational and does not perform automatic failover when the connection is unsuccessful.

4. For the **Monitoring IP Address**, specify an external server (such as Google DNS servers) that maintains high availability.

   The system verifies connectivity beyond the local gateway. Use different monitoring IP addresses for each gateway for accurate monitoring.

   The **LB Weight** parameter determines gateway priority ranking. Lower numerical values indicate higher priority levels.

   For high availability setups where you need different gateways between master and slave devices, enable **Master HASync No Overwrite**.

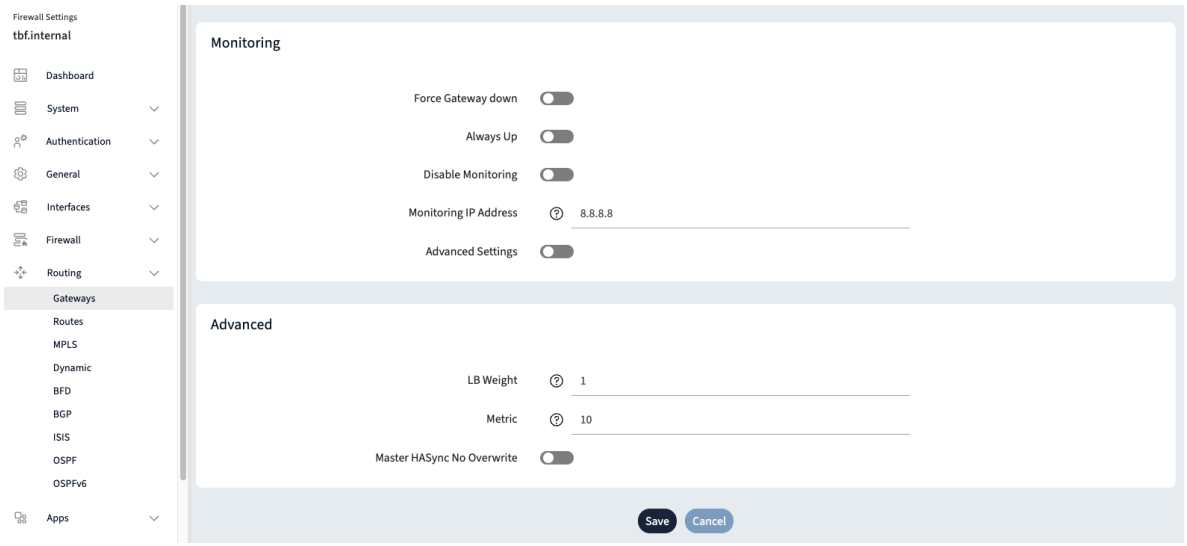


Figure 171: Gateway configuration monitoring

5. Select **Save**.
6. To set up the default gateway, disable **Always Up** and enter a **Monitoring IP Address** that is external to the network.
7. Go to **Routing > Routes > Routing Tables** and select **Edit** for the main routing table.
8. Add the new gateway in the gateway section and assign a priority value.
   Lower numbers represent higher priority levels.

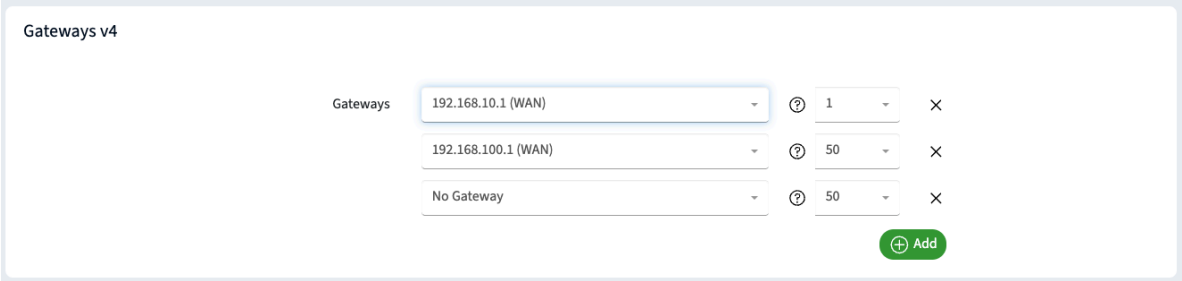


Figure 172: Routes gateway configuration

Policy routing

Routing tables help you create sophisticated yet manageable network configurations in TBF.

To create sophisticated network configurations, route general traffic through the primary internet service provider (ISP) and direct specific services such as VoIP through a secondary ISP connection.

First create a gateway for each WAN connection. For more information, see *Multi-gateway setup* on page 239.

Perform the following steps:

1. To create a new routing table, navigate to **Routing > Routes > Routing Tables** and select **Add**.
2. To restrict VoIP traffic exclusively to the secondary ISP connection, under **Gateways**, add only the secondary gateway.
   Omit the primary gateway from this table.
3. To apply the settings, select **Save**.
4. Go to **Firewall > Rules > WAN**.
5. Select **Add**.
6. Create a new firewall rule. For more information, see *firewall rules (forward and input)* on page 131.

   Include specific conditions that determine which routing table to use. For example, if the VoIP phones operate on a dedicated subnet, select that subnet in the **Sources** section.

   For proper priority, position this rule at the **Top** of the rule list.
7. In the **Advanced Settings**, select the new routing table under **Routing Table**.
   To handle return traffic properly, you can enable **Routing Table Reverse Main**.

Note: Custom routing tables do not automatically include interface-based routes like the **Main** routing table does. This means return traffic lacks proper routing paths unless you manually create routes in the custom table or enable **Routing Table Reverse Main** to force TBF to use the **Main** routing table for return traffic.

16. Services

This section offers access to network services, security services, and system utilities that enhance device functionality and network management capabilities.

Services are background processes that run continuously to offer specific functionality for the network infrastructure. The device includes a set of services that the system organizes into several categories to support different aspects of network operation and management.

Categories

TBF organizes the available services into the following functional categories:

Network services

Essential networking protocols that help provide fundamental network functionality. These include DHCP for automatic IP address assignment, DNS for domain name resolution, and NTP for time synchronization across the network.

Security services

Services that strengthen network security through authentication, filtering, and monitoring capabilities. Examples include 802.1X authentication for device access control, captive portal for user authentication, intrusion detection for threat monitoring, and web application firewall to help protect web services.

Monitoring services

Tools for network monitoring and analysis that help you understand network performance and usage. These include SNMP for device management, Ntopng for traffic analysis, and Netflow for detailed network flow monitoring.

Infrastructure services

System-level services that support device operation and management. Examples include Cron for scheduling automated tasks, HAProxy for load balancing across multiple servers, and startup scripts for custom device initialization.

Specialized services

Additional services that the system designs for specific use cases and advanced networking scenarios. These include Apcupsd for UPS management, Avahi for automatic service discovery, QoS for traffic prioritization, and ZeroTier for software-defined networking.

Management

Enable or disable each service based on specific network requirements. Set up services through dedicated configuration pages that you access through the **Services** menu.

Most services have these common management features:

- Enable or disable the service
- Set service-specific settings and parameters
- View current service status and operational information
- Export configuration settings for backup or migration purposes
- Access service logs for troubleshooting issues

Access service settings

To access any service configuration, navigate to **Services > [Service Name]** in the main menu. Each service has a dedicated interface for configuration and management tasks.

View service diagnostics and status information under **Diagnostics > Services**, where you monitor the operational status of enabled services from a single location.

Apcupsd

Apcupsd is a daemon service that enables communication with and monitoring of American Power Conversion (APC) uninterruptible power supply (UPS) devices.

Find the Apcupsd settings under **Services > Apcupsd**.

Apcupsd is a daemon service that enables the computer to communicate with and monitor APC UPS devices. This utility helps the system respond appropriately to power events and maintain proper UPS monitoring.

Before you use Apcupsd, install it. Install the service under **System > Addons**.

The Apcupsd configuration consists of 3 main areas:

- *Settings* on page 244: Set up basic UPS connection and monitoring parameters
- *Power failures settings* on page 245: Define how the system responds to power outages and inoperability
- *UPS sharing settings* on page 246: Set up network sharing options to allow other systems to monitor the same UPS

General settings

Use the Apcupsd settings to enable and set up the APC UPS daemon service to monitor and manage a UPS device.

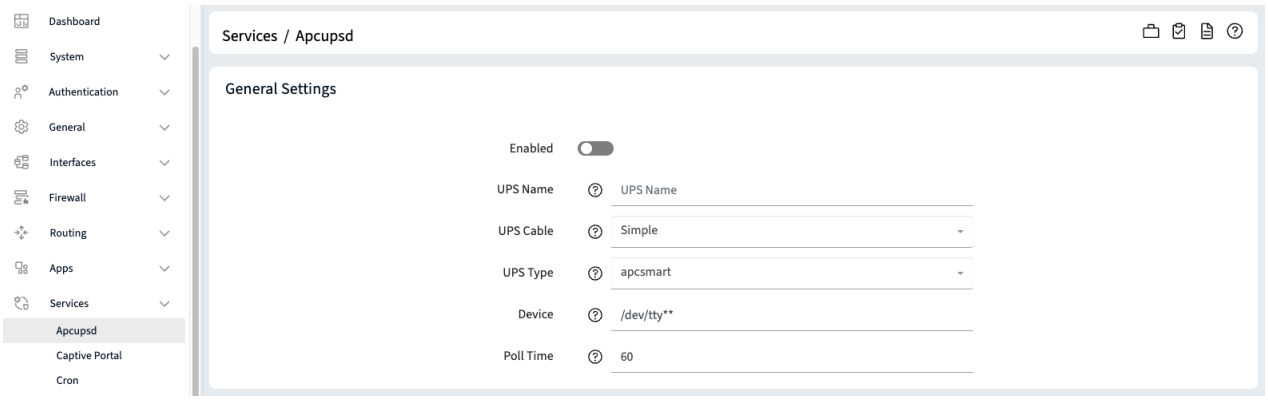


Figure 173: APC UPS daemon

Use this interface to enable or disable the Apcupsd service. This service monitors and manages a UPS device.

UPS Name

This field is for a descriptive name for the UPS device. This name appears in log files and system messages to identify which UPS is reporting status information. Use this setting when you manage multiple UPS devices.

UPS Cable

This setting sets the cable type that connects the UPS to the computer. Select from these options:

- **Simple**: Basic serial cable for former UPS models
- **Smart**: Advanced serial cable for SmartUPS models
- **Ether**: Ethernet network connection
- **USB**: USB connection (common for modern UPS devices)
- **Custom**: Specific cable model (requires additional configuration)

UPS Cable Custom

This field appears when you select **Custom** for the **UPS Cable** setting. Enter the specific cable model number, such as `940-0119A`, `940-0127A`, or `940-0128A`.

UPS Type

This setting sets the communication protocol that matches the UPS model and connection method. Select from:

- **apcsmart**: APC SmartUPS models with serial communication
- **usb**: Modern UPS devices with USB connections
- **net**: Network connection to another Apcupsd server
- **snmp**: SNMP-enabled UPS devices over network
- **netsnmp**: Legacy SNMP driver (use snmp instead)
- **dumb**: Simple UPS devices with basic signaling
- **pcnet**: PowerChute Network Shutdown protocol
- **modbus**: SmartUPS models with MODBUS protocol support

Device

This field sets up the device connection based on the selected **UPS Type**:

- **apcsmart**: Enter the serial device path for SmartUPS models with serial cables. The format is `/dev/tty**`.
- **usb**: Leave blank for automatic detection. Automatic detection works for most USB UPS devices. You rarely need manual configuration.
- **net**: Enter the network address of another computer that runs Apcupsd and monitors the UPS. Use this when the computer connects to a UPS through a network rather than directly. The format is `hostname:port`.
- **snmp**: Enter connection details for SNMP-enabled UPS devices. Use the UPS network address, port (default: 161), vendor (default: `APC`), and community string (default: `private`). The format is `hostname:port:vendor:community`.
- **netsnmp**: Similar to SNMP but uses the former net-snmp library. Use the standard `snmp` option instead unless there are specific legacy requirements. The format is `hostname:port:vendor:community`.
- **dumb**: Enter the serial device path for former UPS models with simple signaling. The format is `/dev/tty**`.
- **pcnet**: Enter connection details for PowerChute Network Shutdown protocol. Include the UPS management card IP address, username, password, and port (default: 3052). The format is `ipaddr:username:passphrase:port`.
- **modbus**: For MODBUS over USB, leave blank. For serial connections, enter the device path. Also specify the UPS serial number to connect to a specific device when multiple USB UPS units are present. The format is `/dev/tty**`.

Poll Time

This field sets how frequently in seconds Apcupsd checks the UPS for status updates. The default is `60` seconds. Lower values provide more frequent updates but increase system load.

Power failures settings

Power failures settings control how the system responds to power outages and battery conditions. Set the shutdown thresholds based on battery level, remaining runtime, or elapsed time on battery power, along with user notification settings.

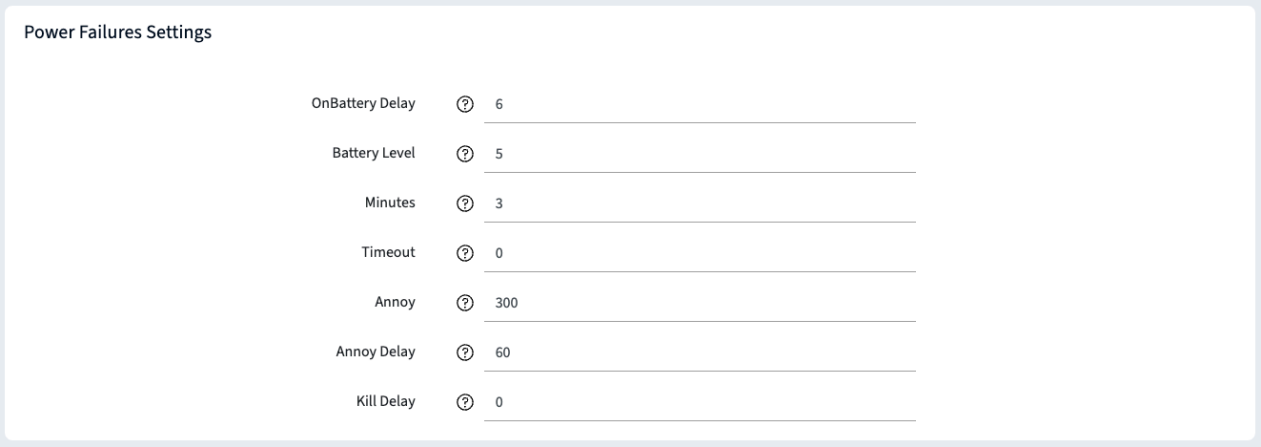


Figure 174: Power failures settings

OnBattery Delay

This field specifies the delay time in seconds between power outage detection and triggering the onbattery event. This delay helps prevent false alarms from brief power interruptions. The default is `6`.

Battery Level

This field specifies the minimum battery charge percentage that triggers an automatic shutdown. When the UPS reports that battery charge drops to or below this level during a power outage, the system shuts down to help prevent data loss. The default is `5`.

Minutes

This field specifies the minimum remaining runtime in minutes that triggers an automatic shutdown. When the UPS calculates that battery runtime drops to or below this value during a power outage, the system shuts down to help provide graceful shutdown completion. The default is `3`.

Timeout

This field specifies the maximum time in seconds that the system runs on battery power before automatic shutdown. This field establishes a hard limit regardless of battery level or remaining runtime. To disable the timeout, set the value to 0. The default is `0`.

Annoy

This field specifies the interval in seconds between repeated warning messages that notify users to save work and log off before system shutdown. These messages help users prepare for the impending shutdown. To disable the warning messages, set the value to 0. The default is `300`.

Annoy Delay

This field specifies the initial delay in seconds after power outage detection before sending the first warning message to users. This delay allows time to restore power without alarming users unnecessarily. The default is `60`.

Kill Delay

This field specifies the time in seconds that Apcupsd waits after it initiates a shutdown before it attempts to cut UPS power. This delay helps provide time for the system to shut down completely before the UPS removes power. Use this on systems where Apcupsd cannot regain control after shutdown. To disable this delay, set the value to 0. The default is `0`.

UPS sharing settings

UPS sharing settings enable multiple systems to share a single UPS device using APC ShareUPS cards. To establish master-slave relationships for coordinated power management across multiple connected systems, set the UPS class and mode.

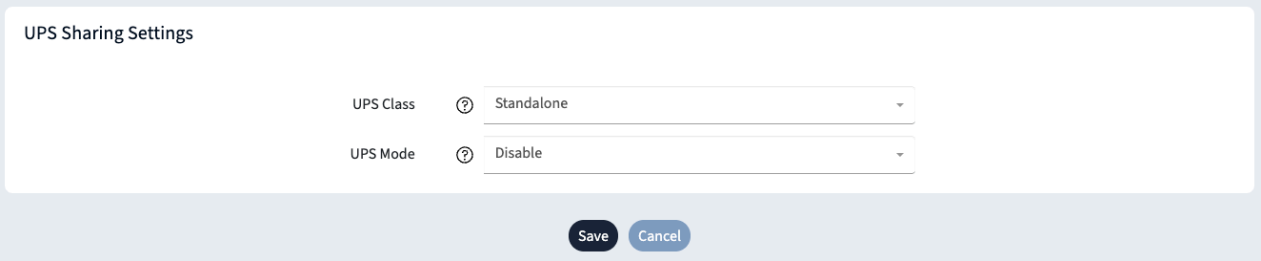


Figure 175: UPS sharing settings

UPS Class

This setting specifies the role this system plays in a shared UPS configuration. Select from:

- **Standalone**: Single system connected directly to UPS (default). Select this option for single-system UPS connections.
- **Share Master**: Primary system that controls UPS shutdown decisions
- **Share Slave**: Secondary system that follows master shutdown commands

UPS Mode

This setting becomes available when you select **Share Master** or **Share Slave** for **UPS Class**. The mode defines specific operational parameters for shared UPS configurations using APC ShareUPS cards. This setting remains disabled for standalone configurations.

802.1X authenticator

The 802.1X authenticator assists in establishing network security by verifying devices at layer 2 before they access the network.

To access the 802.1X authenticator settings, navigate to **Services > 802.1X Auth.**.

The 802.1X authenticator acts as a checkpoint that verifies devices at layer 2 before it allows network access. This authentication process accepts either username and password credentials or digital certificates. When you implement this layer, you help prevent unauthorized devices from connecting to the network.

Before you use the 802.1X authenticator, install the authenticator. To complete the installation, navigate to **System > Addons**.

When you activate the 802.1X authenticator on a network interface, the authenticator blocks traffic forwarding on that interface until a device completes successful authentication. To monitor authenticated devices, use the *802.1X authenticator diagnostics* on page 390 page.

WARNING: You cannot enable the **DHCP Server** on any interface when you use the 802.1X authenticator. The authenticator blocks the DHCP server port and does not start if a DHCP server is active. This restriction allows the 802.1X authenticator to properly detect new devices that join the network. Verify that you have an alternative DHCP server available in the network environment.

802.1X authenticator settings

This section provides configuration options to enable and set up the 802.1X authenticator on physical interfaces.

Enable or disable the 802.1X authenticator feature. TBF allows you to create a separate 802.1X authenticator for each physical interface on the device.

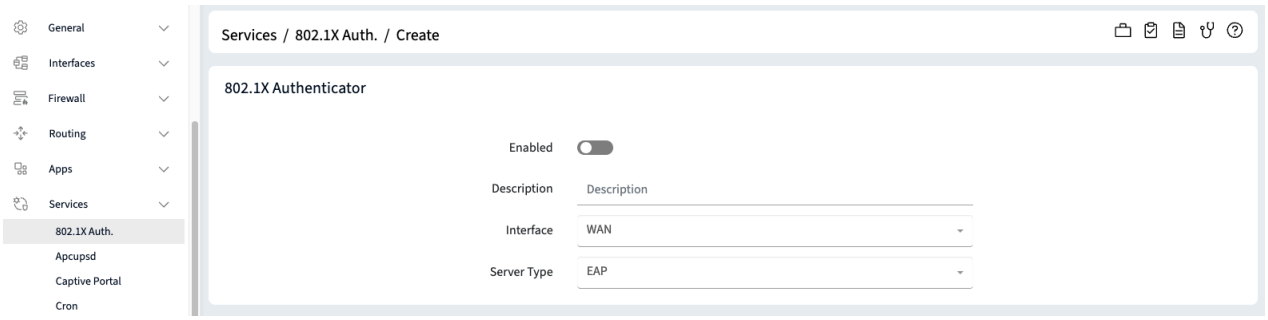


Figure 176: 802.1X authenticator

Server Type

This setting determines how the system handles authentication requests. There are two options: forward authentication requests from devices to an external RADIUS server, or set up the 802.1X authenticator to work as a local EAP authentication server.

Set up RADIUS authentication

Set up the 802.1X authenticator to forward authentication requests to a RADIUS server.

Set up both an authentication RADIUS server and an accounting RADIUS server. The RADIUS server processes authentication requests from devices and sends back a response that determines whether a device gets network access.

Perform the following steps:

1. Specify the RADIUS server IP address.
2. Enter the identifier for the 802.1X authenticator.
3. Set up a shared secret to assist in establishing secure communication between the authenticator and RADIUS server.

Set up an EAP authentication server

Set up the 802.1X authenticator to work as a local EAP authentication server.

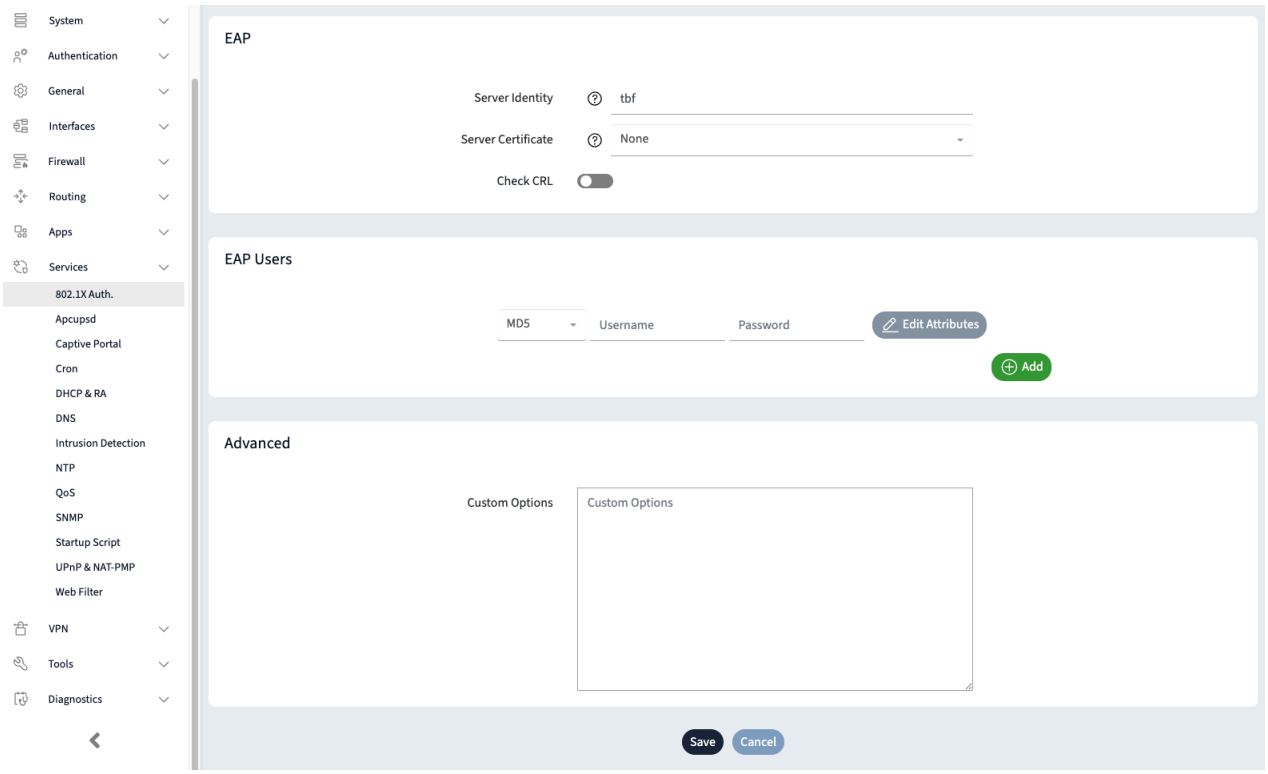


Figure 177: EAP

Use the 802.1X authenticator as an EAP authentication server to verify client credentials locally.

Perform the following steps:

1. Set the server identity that the EAP authentication server presents during authentication.

   Server Identity

   This field specifies the identity name that the EAP authentication server presents to clients during the authentication process. This helps clients verify they are connecting to the correct server.

2. Add a server certificate to assist in establishing secure, certificate-based authentication.

   Server Certificate

   This setting specifies the certificate to assist in establishing secure authentication methods. This setting is optional but required when you use certificate-based authentication protocols like TLS or TTLS.

   Check CRL

   This setting verifies that client certificates have not been revoked by checking against a certificate revocation list (CRL).

3. Add devices to the EAP client list and specify their authentication details:

   Authentication Algorithm

   This setting specifies the authentication method that matches the security requirements. Select from:

   - **MD5**: This option uses basic password authentication (least secure).
   - **PSK**: This option uses pre-shared key authentication.
   - **MSCHAPV2**: This option uses Microsoft Challenge Handshake Authentication Protocol version 2.
   - **TLS**: This option uses Transport Layer Security (certificate-based).
   - **TTLS**: This option uses Tunneled Transport Layer Security (certificate with inner authentication).

   Username

This field specifies the unique username that identifies this device during authentication.

Password

This field specifies the password that corresponds to the username. This field is required for password-based authentication methods.

Custom Options

This field specifies additional configuration parameters specific to the authentication setup or network requirements.

Avahi

Avahi enables Apple services to communicate across different network segments by forwarding multicast DNS traffic between network boundaries.

Access Avahi settings

Find the Avahi settings under **Services > Avahi**.

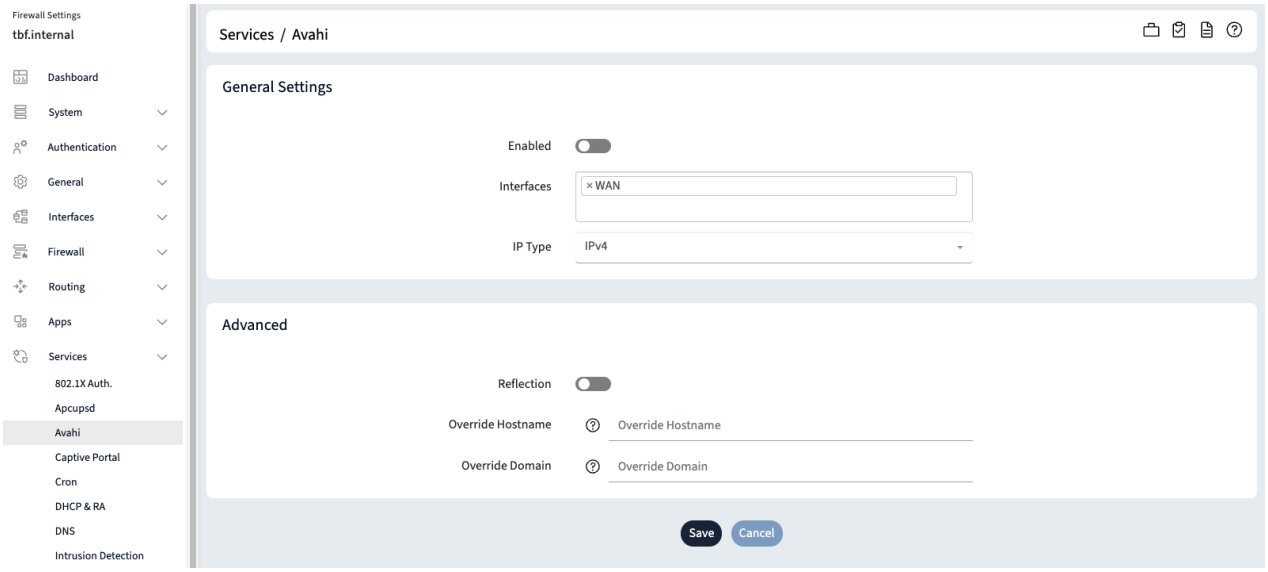


Figure 178: Avahi settings

About Avahi

Avahi allows Apple services to communicate across different network segments. Without Avahi, these services cannot cross layer 2 boundaries because broadcast traffic stays within local network segments.

Important: Before you use Avahi, install it. To install it, navigate to **System > Addons**.

Next steps

For detailed setup options, see the *Avahi settings* on page 250.

Avahi settings

This section covers settings for the Avahi service that controls multicast DNS forwarding between network segments.

Settings

Enabled

This setting enables or disables Avahi.

Interfaces

This setting specifies which network interfaces Avahi monitors and uses for service discovery. Select the interfaces that connect to the networks where you want to enable service propagation.

IP Type

This setting determines the IP protocol version that Avahi uses. Select **IPv4**, **IPv6**, or both protocols depending on the network setup.

Reflection

This setting enables mDNS packet forwarding between different network subnets. When you enable this setting, clients on one subnet discover and access services on other subnets.

Override Hostname

This field specifies a custom hostname for mDNS record publishing instead of using the default hostname of the system.

Override Domain

This field specifies a custom domain name for mDNS record publishing. The default is `local`.

Captive portal

A captive portal authenticates users before they access the internet, helping ensure only authorized users connect to the network.

Overview

Find the captive portal settings under **Services > Captive Portal**.



Figure 179: Captive portal

Tip: To export the settings as an Excel spreadsheet, use the export button in the upper-right corner. This feature helps you create backups or maintain documentation.

Configuration areas

The captive portal includes 5 main configuration areas:

- *Captive portal instance* on page 251: Set up and manage configuration parameters for captive portal instances
- *Captive portal files* on page 254: Manage HTML file requirements and customize the portal appearance
- *Redirect device IDs* on page 254: Control device ID management for authentication redirects
- *Voucher* on page 255: Set up voucher-based authentication for temporary access
- *Redirect mode* on page 256: Set up distributed captive portal deployment

Captive portal instance

This section provides configuration settings and parameters for captive portal instances, including interface binding, authentication options, and access control.

General settings

Bind each captive portal instance to one or more network interfaces.

Note: Currently, no user authentication is required. Users only need to check a checkbox to log in.

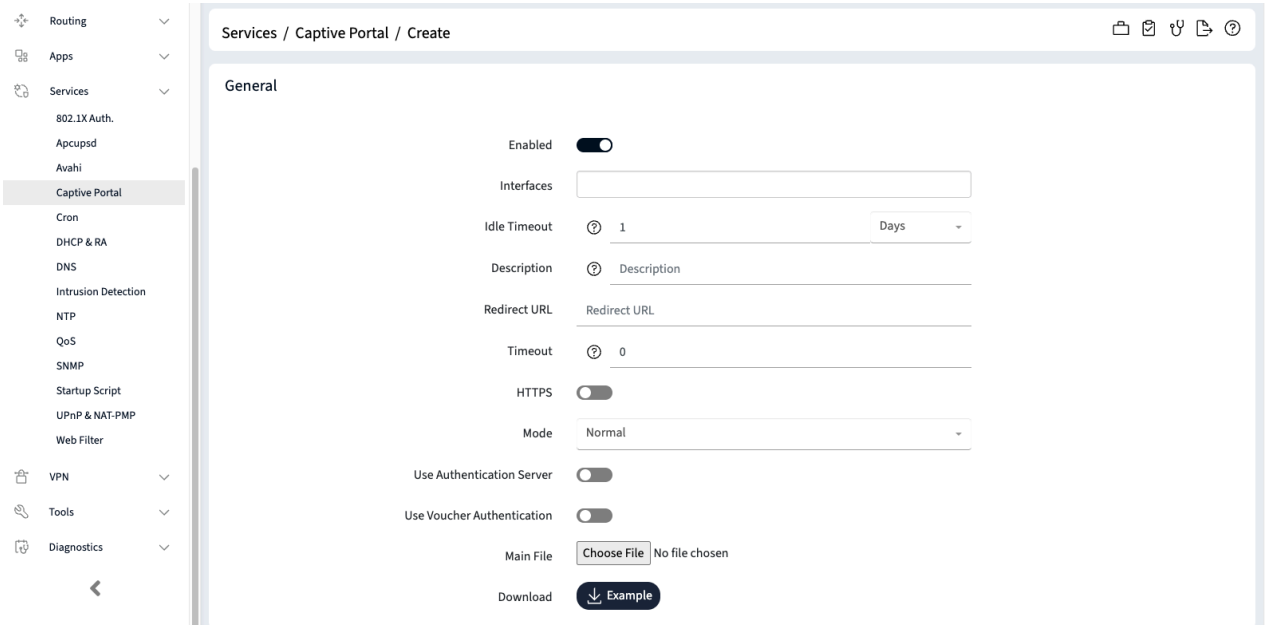


Figure 180: General

The system identifies users by the MAC address of their device.

Important: Verify that only a layer 2 network connects to the captive portal interface.

Settings

Enabled

This setting enables or disables the captive portal instance.

Interfaces

This setting specifies which network interfaces the captive portal monitors and controls.

Idle Timeout

This setting automatically logs out users who do not send network traffic within the specified time period.

Redirect URL

This setting redirects users to the specified URL after successful authentication.

Timeout

This setting determines how long (in minutes) a user remains logged in. The default is `0`, which means no timeout applies.

HTTPS

This setting allows the captive portal to run on HTTPS. It requires selecting a certificate that connecting clients trust.

Mode

Set this option to either **Normal** or **Redirect** mode.

Redirect Address

This setting is available when you select **Redirect** mode. The system redirects requests to a different TBF firewall for processing. This puts the captive portal in forwarding mode, where the remote server handles authentication, for example, `http://tbf.server:port/`.

Use Authentication Server

This setting enables the use of an external authentication server for the captive portal.

Authentication Server

Select one or multiple authentication servers that you define in the *authentication server* on page 80. Verify that the login template includes both user and password fields. The `index_auth.html` template provides an example.

Use Voucher Authentication

This setting enables authentication using vouchers.

Main File

This setting specifies the `index.html` file that the system presents to users when they connect. Upload only HTML files.

Additional files and access control

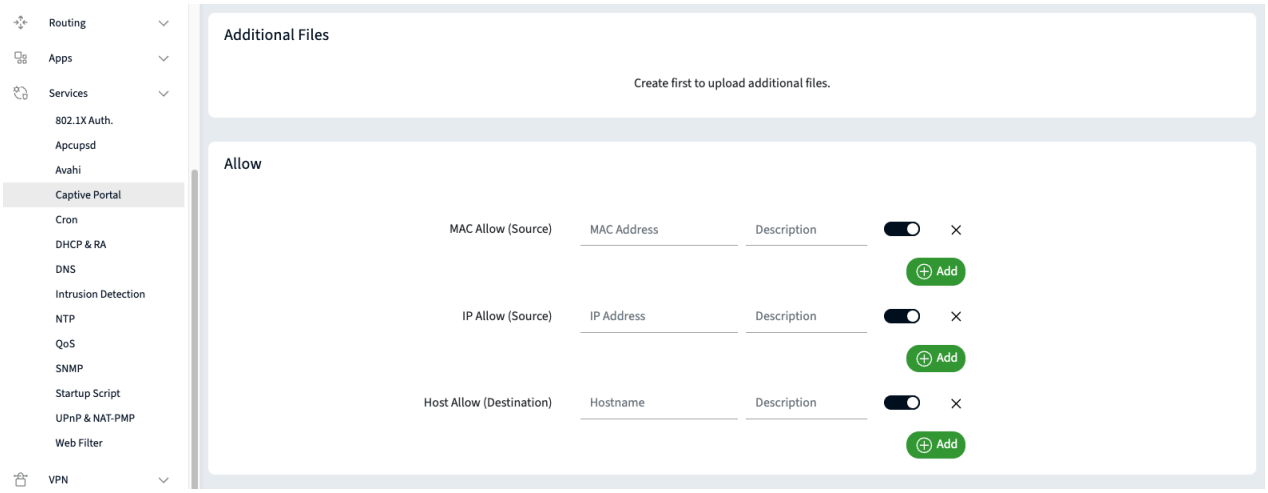


Figure 181: Additional files and allow

To customize the captive portal, upload additional files. For example:

- CSS stylesheets
- JavaScript files
- Images
- Error pages
- Success pages

Mac Allow (Source)

This setting excludes specified MAC addresses from the portal authentication requirement.

IP Allow (Source)

This setting excludes specified IP addresses from the portal authentication requirement.

Host Allow (Destination)

This setting excludes specified destination hosts from the portal authentication requirement. To allow access to specific websites without authentication, use this option.

Note: The system processes captive portal settings before DNAT. For allow and exclude rules, use the address before the system applies any DNAT changes.

Captive portal files

This section provides HTML file requirements and customization options for captive portal login pages, including required elements and optional features.

The primary file for the captive portal is the `index.html` file. You can include CSS styling and JavaScript functionality in this file. Either embed the styles and scripts directly in the HTML file or upload them as separate files to the portal.

Perform the following steps:

1. Add the required `accept` element to the HTML file.

   Include an `accept` element with the following structure in the HTML file:

   ```
   <input type="submit" name="accept" class="login login-submit" value="Login" id="login">
   ```

   This element enables the captive portal to handle user authentication properly.

2. To customize post-login navigation, include an optional redirect input element.

   To direct users to a specific page after successful login, add this redirect input element:

   ```
   <input name="redirurl" type="hidden" value="https://www.belden.de">
   ```

   This setting takes precedence over the **Redirect URL** option that you set up in the GUI.

3. To improve the user experience, add optional error and success pages.

   To display custom messages when authentication is unsuccessful, you can create an `error.html` file. Additionally, to show a custom success page after authentication, you can create a `success.html` file.

   Note: The success page only displays when you set up no redirect URL.

The system automatically serves the `index.html` file when no other specific page is available.

To help you get started with customization, you can download example files for each captive portal instance.

Redirect device IDs

Device ID management assists in establishing secure authentication redirects between different captive portal instances in the network.

Device ID configuration

To establish trusted connections, set up one or more **Device IDs** for a captive portal. When you specify these device IDs, the system allows authentication redirects from other TBF devices that match the IDs.

This feature helps assure that only authorized devices redirect users to the captive portal for authentication. This assists in establishing network security and enables seamless user access across multiple portal instances.

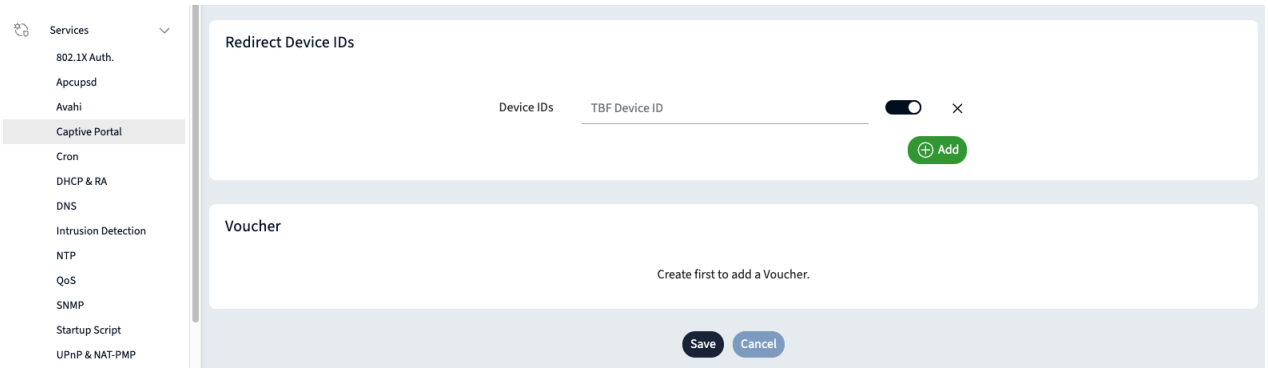


Figure 182: Redirect device IDS and voucher

Tip: To assist in establishing a secure network of interconnected captive portals that redirect users between different authentication points, use device ID management.

Voucher

Create and manage vouchers for temporary captive portal authentication, including single-use and multiple-use options.

Vouchers help provide a convenient way to grant temporary network access through the captive portal without requiring permanent user accounts. Each voucher contains multiple tickets that users redeem for authentication.

When you create a voucher, you specify 2 key settings:

Number of Tickets

This setting specifies the total number of authentication codes the voucher contains. The default is `100`.

Usage Type

How tickets are used:

- **Single Use**: Each ticket works only once and expires after authentication.
- **Multiple Use**: Each ticket works repeatedly until the voucher expires.

After you enable **Use Voucher Authentication**, users access the network by entering a voucher code instead of traditional login credentials.

Perform the following steps:

Add the voucher input field to the HTML page.

To enable voucher authentication on the captive portal, add a password input field to the HTML login form:

```
<input type="password" name="auth_voucher" placeholder="Voucher Code" id="auth_voucher">
```

The system automatically detects and processes voucher codes when it finds the `auth_voucher` field name in the submitted form.

Users now authenticate using voucher codes instead of traditional username and password combinations. This helps provide a flexible solution for guest access, temporary users, or situations where you need to grant network access without creating individual user accounts.

Tip: Voucher authentication is ideal for conference attendees, temporary contractors, or guest users who need limited-time network access.

Redirect mode

This section describes the distributed captive portal setup that forwards authentication requests to a central server for processing.

Distributed architecture

Redirect mode enables you to create a distributed captive portal architecture. When you enable **Redirect Mode**, the local TBF device forwards HTTP authentication requests to a central TBF device that you specify in the **Redirect URL** field.

This setup allows multiple satellite TBF devices to share a single authentication portal to simplify management and maintain a consistent user experience across the network.

User management

In this distributed setup, the central TBF device handles authentication processing and serves the captive portal interface. However, each satellite TBF device that forwards requests continues to track and display authenticated users locally.

To disconnect users, remove them directly from the satellite TBF device where they appear, not from the central authentication server. Standard authentication features remain functional in **Redirect Mode**.

Important: Perform user disconnection on the satellite TBF device that shows the authenticated user, not on the central authentication server.

Benefits

- Centralized authentication management
- Consistent user experience across multiple locations
- Simplified portal maintenance and updates
- Reduced administrative overhead

Cron

Cron offers automated task scheduling through cron jobs, which are commands that execute periodically at specific times or intervals.

Cron is a time-based job scheduler that allows you to automate repetitive tasks on the system. Set up commands to run automatically at specific times, dates, or intervals without manual intervention.

Find the cron settings under **Services > Cron**.

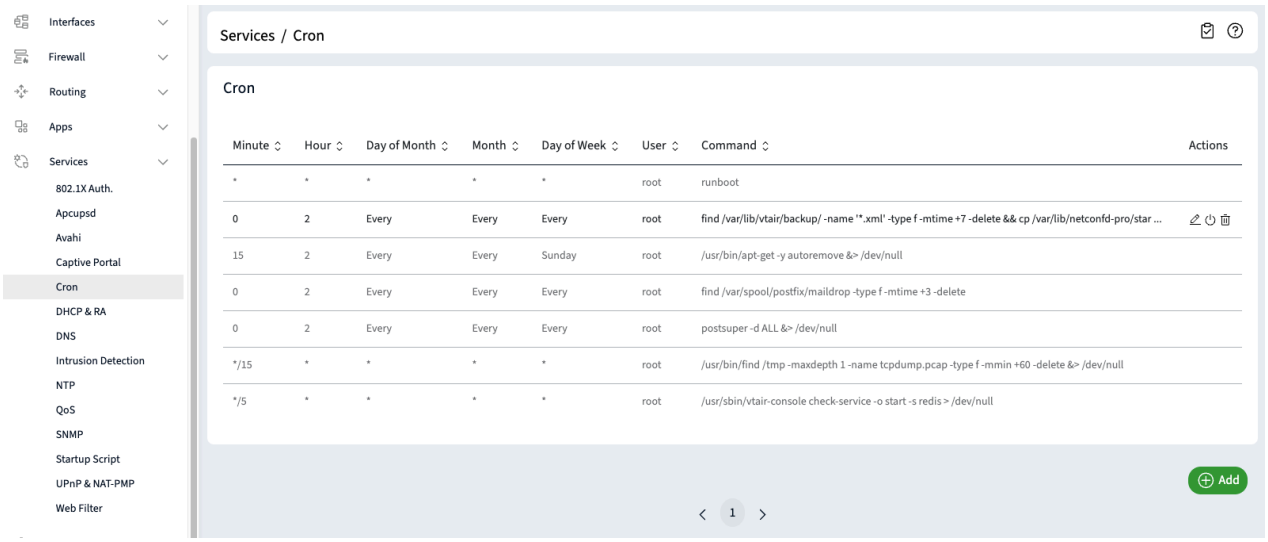


Figure 183: Cron

In this section, create and manage cron jobs. Each cron job is an automated command that runs according to the schedule you define. For example, set up a cron job to back up files every night at midnight or to clean temporary files every week.

TBF comes with several preconfigured cron jobs that handle essential maintenance tasks. TBF has already set up these jobs and runs them to keep the system operating smoothly.

Related topics:

- *Preconfigured crons* on page 257
- *Create a new cron job* on page 257

Preconfigured crons

TBF includes 3 preconfigured cron jobs that handle system startup, service monitoring, and automated backups.

TBF comes with 3 essential cron jobs that it sets up and runs automatically.

The `runboot` command helps ensure that necessary startup scripts run properly when the system boots up.

The `check_service` command continuously monitors two critical services: the Daphne HTTP WebSocket protocol server and the Redis in-memory database. This monitoring helps ensure both services remain operational.

Note: Because they are essential for proper system operation, you cannot delete these 2 cron jobs.

The third cron job automatically creates a complete TBF backup every day at 2:00 AM. TBF saves these backups to the `/var/lib/vtair/backup/` folder. Each backup file contains the current date in its filename for easy identification. To help prevent storage issues, TBF automatically deletes any backups that are older than 7 days.

Create a new cron job

To create a new cron job, specify the user, command, and time interval settings for automated task execution.

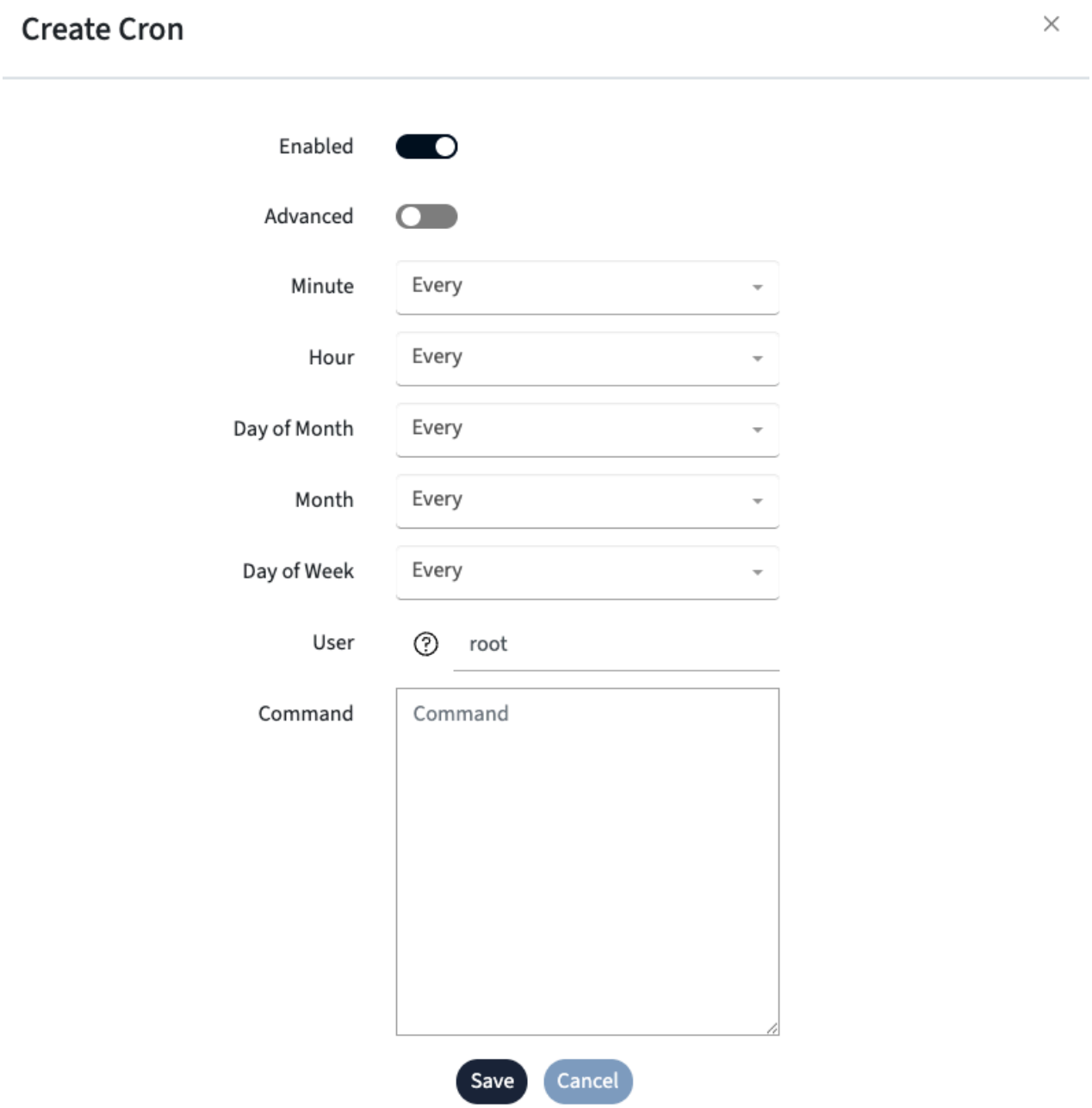


Figure 184: Create cron

When you create a new cron job, you specify which user runs which command and when it runs. This setup allows you to automate tasks at specific times or intervals.

Perform the following steps:

1. To set the time parameters, select values for each of the following fields:
   - **Minute**: 0..59
   - **Hour**: 0..23
   - **Day of Month**: 1..31
   - **Month**: 1..12
   - **Day of Week**: 0..7, where 0 and 7 represent Sunday

   Use an asterisk (*) in any field to indicate *every*. For example, an asterisk in the minute field means every minute.

2. To create more complex scheduling patterns, enable the **Advanced** option and use standard cron syntax.

   Advanced mode allows you to use special characters like commas (,) for lists, hyphens (-) for ranges, and forward slashes (/) for intervals. Examples include 1,3,5, 1-5, and */15.

DHCP and RA

Dynamic Host Configuration Protocol (DHCP) and IPv6 Router Advertisement (RA) services automatically assign network settings to devices. This helps eliminate the need for manual configuration.

Find the DHCP and RA settings under **Services > DHCP and RA**.

DHCP overview

Important: To use DHCP, the interface must have a static IP address.

TBF offers DHCP support for both IPv4 and IPv6 networks. Enable DHCP on selected interfaces and customize settings such as:

- IP address ranges
- Lease durations
- Specialized options for client devices

Watch this video tutorial: *DHCP Configuration Tutorial*

Related topics

For step-by-step setup instructions, refer to these topics:

- *Set up network booting* on page 259
- *MAC and DUID deny* on page 260
- *Extra options* on page 260
- *Relay server* on page 260
- *IPv6 router advertisement* on page 261
- *IPv6 prefix delegation* on page 262
- *Host reservation* on page 262

Set up network booting

To allow devices to start from a network server instead of their local hard drives, set up network booting.

This approach proves particularly valuable for diskless workstations, thin clients, and automated system deployments where you require centralized boot management.

Perform the following steps:

1. Select the **Enable network booting** checkbox.
2. Enter the IP address of the boot server in the **Next server** field.
3. Type the name of the boot file in the **Default boot filename** field.

Network devices now start from the designated network server with the specified boot file.

MAC and DUID deny

To block specific devices from obtaining DHCP addresses, deny their MAC addresses or DHCP Unique Identifier (DUID) identifiers.

For IPv4 networks, block MAC addresses. For IPv6 networks, block DUID values. This measure helps you control which devices access the network.

MAC address blocking

For IPv4 DHCP services, block devices with their MAC addresses. Each network interface has a unique MAC address that serves as its hardware fingerprint. When you add a MAC address to the deny list, the DHCP server refuses to assign an IP address to that specific device.

DUID blocking

For IPv6 DHCP services, block devices with their DUID. DUIDs work similarly to MAC addresses but provide more reliable identification for IPv6 clients. Unlike MAC addresses, DUIDs remain consistent even when a device has multiple network interfaces or changes its network setup.

Extra options

DHCP extra options help provide additional setup parameters to client devices beyond standard network settings.

While DHCP typically helps provide IP addresses, subnet masks, and gateway settings, extra options deliver specialized parameters that help devices connect and function properly on the network. These options use standardized codes that RFC documents define to help ensure compatibility across different systems.

Option format

DHCP options use numbered codes that follow industry standards. Each option code corresponds to a specific type of setup data and has a defined format. This standardization helps ensure that different DHCP servers and client devices communicate effectively, regardless of manufacturer.

Common DHCP options

Popular DHCP options include:

- **DNS server addresses**: Help devices resolve domain names
- **Domain name settings**: For network identification
- **NTP server information**: For time synchronization
- **Vendor-specific data**: For device-specific setups

These options help automate network setup and reduce manual work on client devices.

Relay server

DHCP relay servers forward DHCP requests between network segments. This forwarding allows a single DHCP server to serve multiple subnets.

A DHCP relay server acts as a bridge between clients and DHCP servers on different network segments. When you have multiple subnets, the relay server enables a single DHCP server to help provide IP addresses and setup to clients. This setup eliminates the need for separate DHCP servers on each segment.

How relay functionality works

The relay server operates as an intermediary in the DHCP process. When a client broadcasts a DHCP request on the local network segment, the relay server captures and converts this broadcast message into a unicast message. The relay server then forwards this message directly to the designated DHCP server on a different network segment. This process allows DHCP communication to cross network boundaries that block broadcast traffic.

Set up the relay server

To set up the relay server, specify the IP address of the remote DHCP server in the **Relay server** field. This IP address tells the relay server where to forward the DHCP requests the relay server receives from clients on the local network segment.

IPv6 router advertisement

IPv6 Router Advertisement (RA) allows routers to advertise their presence and network setup information to IPv6 hosts.

RA is a protocol mechanism that enables routers to announce their presence and share network setup details with IPv6 devices on the same network segment. This automatic discovery process helps devices connect to the network without manual setup.

Important: Before you use IPv6 router advertisement, set up a static IPv6 address for the interface.

Router advertisements enable IPv6 devices to automatically set up their network settings, including default gateways, network prefixes, and the methods for address assignment. This process helps remove the need for manual network setup on each device.

Setup modes

TBF offers several IPv6 address setup modes to meet different network requirements:

- *Router only* on page 261: Announces router presence without providing address setup services
- *SLAAC* on page 261: Stateless Address Autosetup allows devices to generate their own addresses
- *DHCP* on page 261: DHCPv6 helps provide centralized address management
- *Both* on page 262: Combined SLAAC and DHCP mode offers flexible address setup options

Router only

Router only mode advertises the device as a router but provides no address configuration information.

In **Router only** mode, the TBF device advertises itself as a router but does not provide address configuration information. Hosts require other methods to obtain IPv6 addresses.

Use cases

To provide routing services but delegate address assignment to other mechanisms or servers, use this mode.

SLAAC

Stateless Address Autoconfiguration (SLAAC) allows IPv6 hosts to automatically generate their own IP addresses.

SLAAC allows IPv6 hosts to automatically generate their own IP addresses with the network prefix that the router advertises and their interface identifier.

Address generation

In **SLAAC** mode, hosts combine the network prefix from router advertisements with their own interface identifier to create unique IPv6 addresses.

Benefits

SLAAC helps provide automatic address configuration without a centralized DHCP server to simplify network administration.

DHCP

DHCP mode directs IPv6 hosts to obtain their addresses and configuration from a DHCPv6 server.

In **DHCP** mode, the router advertisement directs hosts to obtain their IPv6 addresses and configuration from a DHCPv6 server.

Centralized management

This mode helps provide centralized address management and allows for more detailed configuration options through DHCPv6.

Configuration options

DHCPv6 helps provide additional network parameters beyond IP addresses. These parameters include DNS servers, domain names, and other network settings.

Both

Both mode combines SLAAC and DHCP.

The **Both** mode combines SLAAC and DHCP and allows hosts to use SLAAC for address configuration and obtain additional parameters from DHCPv6.

Hybrid approach

This mode helps provide the flexibility of SLAAC for address assignment combined with the rich configuration options available through DHCPv6.

Best of both worlds

Hosts automatically set up their addresses with SLAAC and still receive specific network parameters like DNS servers from DHCPv6.

IPv6 prefix delegation

IPv6 prefix delegation automatically assigns IPv6 network prefixes to downstream routers. This creates an organized hierarchy for address distribution.

Important: Before you use IPv6 prefix delegation, verify that the interface has a static IPv6 address, or set it up to receive a delegated prefix from an upstream router.

When you enable this feature, TBF shares portions of its IPv6 address space with other routers. This method allows those routers to help provide IPv6 connectivity to their own network segments without manual address setup.

Hierarchical addressing

Prefix delegation creates a structured IPv6 addressing system where upstream routers distribute address space to downstream routers. This approach organizes the network into clear layers and makes address management more efficient and scalable.

Automatic assignment

The delegation process works automatically once you set it up, which significantly reduces the manual setup that complex network layouts require. This automation helps prevent incorrect setups and simplifies network expansion.

Host reservation

To assign specific IP addresses to particular devices based on their MAC addresses or DUID, create host reservations.

Host reservations let you assign specific IP addresses to particular devices based on their MAC addresses (for IPv4) or DUID (for IPv6). This feature helps ensure that specific devices receive the same IP address, which is useful for servers, printers, or other network equipment that need consistent addressing.

Perform the following steps:

1. To create a new reservation, select **Add**.
2. Enter the MAC address of the device (for IPv4 devices) or DUID (for IPv6 devices).
   Find the MAC address on the device label or in its network settings.
3. Specify the IP address you want to assign to this device.
   Verify that the IP address falls within the DHCP pool range and is not already in use.
4. For easier identification of the device, optionally provide a hostname.
5. To apply the reservation, save the setup.

Host reservations take priority over dynamic address assignment. This means the DHCP server assigns the reserved IP address to the specified device and does not give that address to any other device on the network.

To test the reservation, restart the target device or renew its DHCP lease. Verify that it receives the correct reserved IP address.

DNS

Domain Name System (DNS) translates human-readable domain names into their corresponding numerical IP addresses. This translation makes it easier to access websites and services.

Find the DNS settings under **Services > DNS**.

DNS acts as the phone book of the internet. When you type a website address like *example.com* into a browser, DNS translates that human-readable name into the numerical IP address to locate the website. This translation process happens automatically in the background. This process makes it much easier for you to navigate the internet without the need to memorize complex number sequences.

Watch this video tutorial: *DNS Configuration Tutorial*

The DNS service offers several configuration options to help you manage how the network handles domain name resolution:

- *General settings* on page 263: Set basic DNS server options and primary settings
- *Advanced settings* on page 265: Access detailed configuration parameters to fine-tune DNS behavior
- *Host overrides* on page 267: Create custom DNS entries that point specific hostnames to designated IP addresses
- *Domain overrides* on page 267: Specify custom DNS servers to handle queries for particular domains
- *DNS firewall* on page 267: To block access to unwanted or malicious websites, filter DNS requests
- *Redirect DNS requests* on page 269: Force DNS traffic to go through the device for centralized control
- *Wildcard domain DNS entries* on page 270: Set up wildcard DNS entries that apply to multiple subdomains at once
- *DNS troubleshooting* on page 270: Diagnose and resolve common DNS-related connectivity issues

General settings

This section provides basic DNS server settings, including interfaces, port, security options, and forwarding behavior.

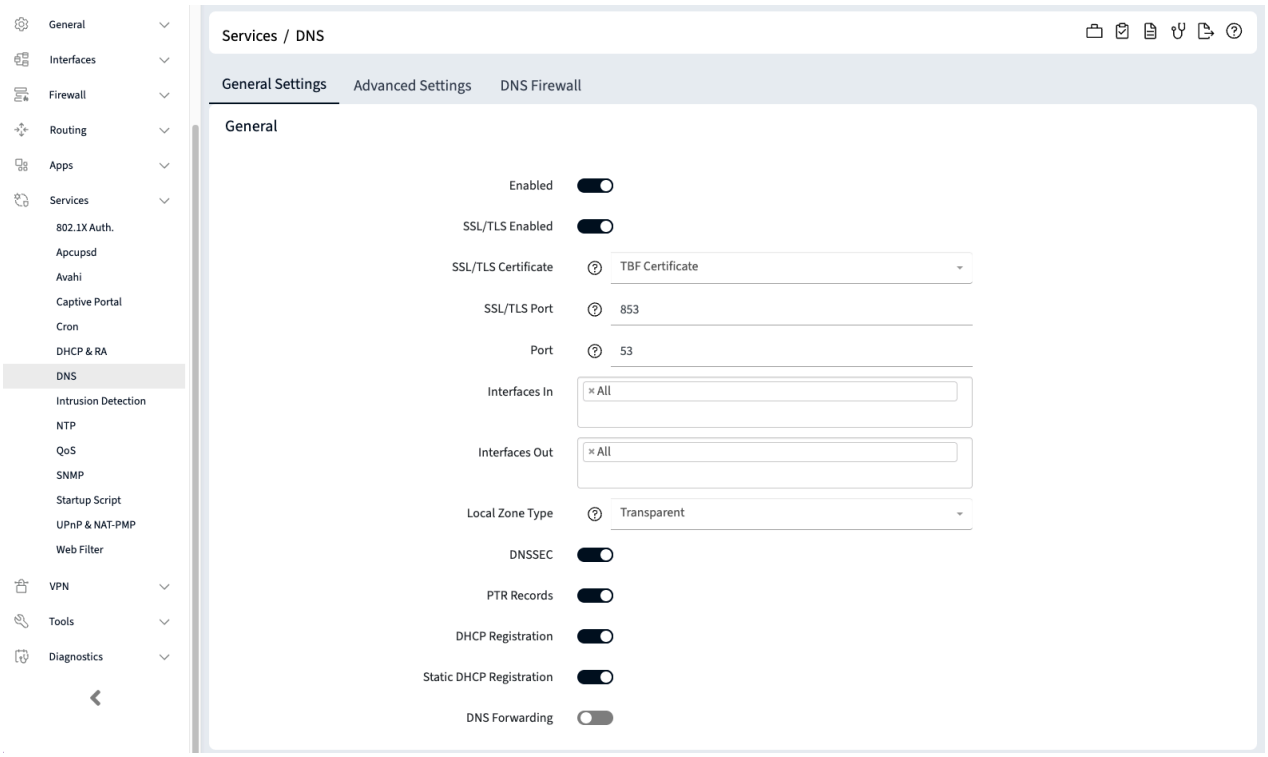


Figure 185: DNS

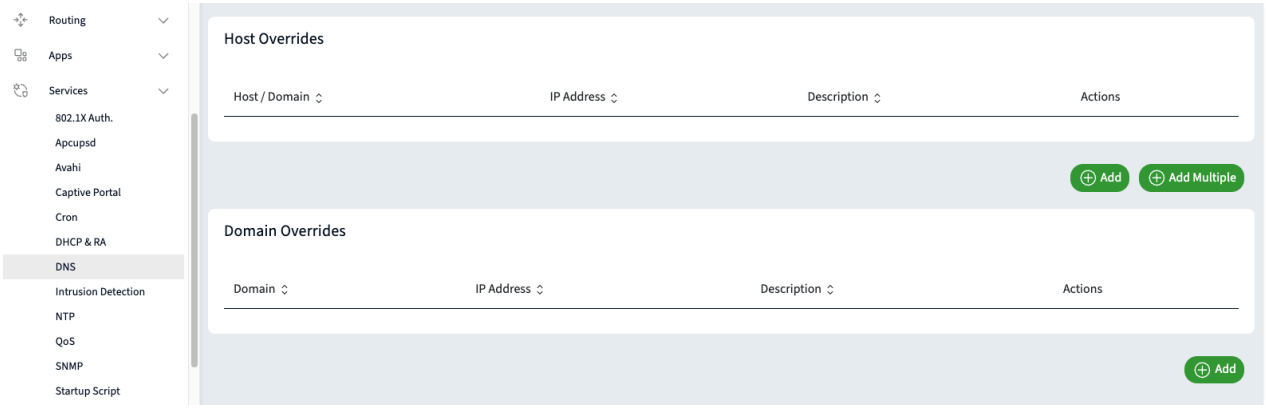


Figure 186: Host overrides and domain overrides

Enable or disable the DNS server. By default, TBF enables **SSL/TLS** and uses the TBF certificate with port 853.

Settings

Interfaces In and Interfaces Out

Specify which network interfaces the DNS server listens on for incoming requests and uses for outbound queries.

Port

This field specifies the port number where DNS operates. The default is `53`.

Local Zone Type

This setting determines how the DNS server processes local queries. The default is **Transparent**.

DNSSEC

This setting enables DNS security extensions to assist in the protection against unauthorized DNS access.

PTR Records

When you enable this setting, TBF automatically creates reverse DNS lookup records for host overrides.

DHCP Registration

This setting automatically adds DHCP lease information to the DNS server for name resolution.

Static DHCP Registration

This setting adds DHCP host reservations to the DNS server for consistent name resolution.

DNS forwarding

By default, the DNS server queries the DNS root servers directly without forwarding requests to other servers. To specify multiple **DNS Forward Servers**, enable **DNS Forward** and enter their IP addresses. TBF then routes DNS queries to these servers instead of contacting the root servers directly.

Export settings

To export the current settings as an Excel spreadsheet, use the export option in the upper-right corner of the interface.

Advanced settings

Set advanced DNS server settings for caching, logging, custom options, and to assist in establishing security.

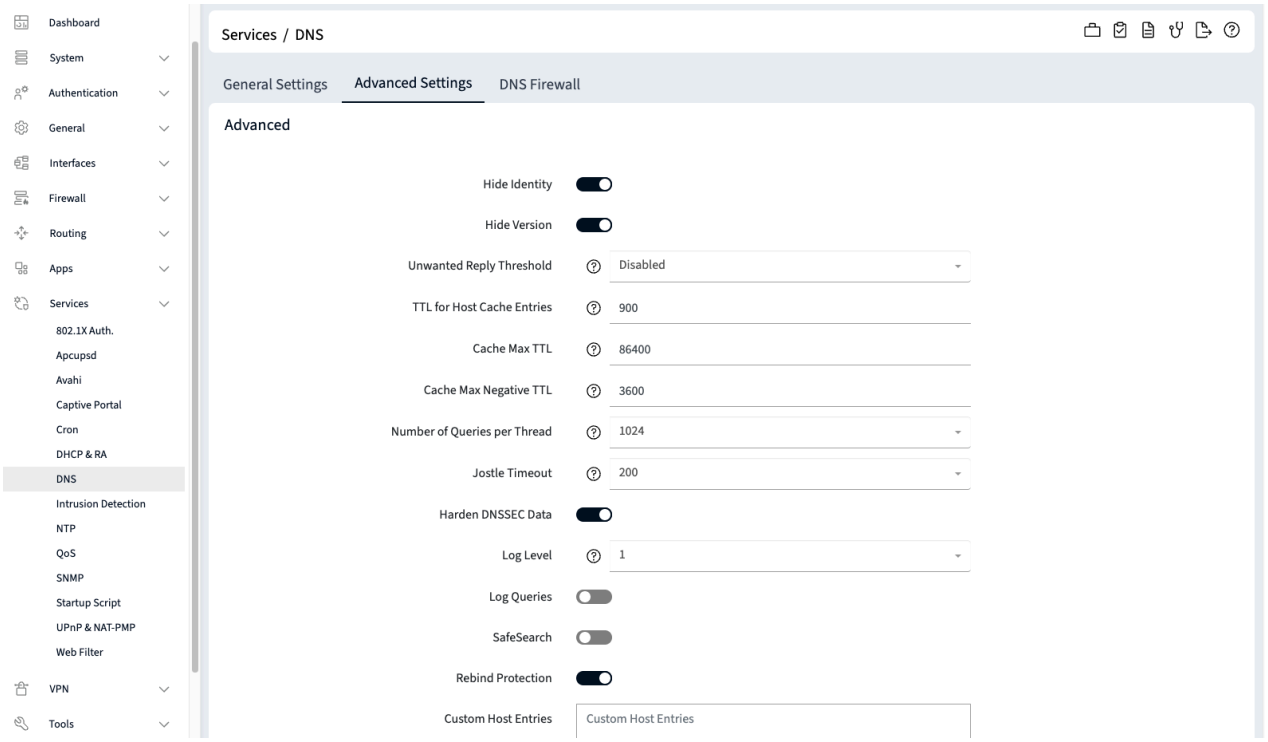


Figure 187: Advanced

Settings

Hide Identity

This setting blocks `id.server` and `hostname.bind` queries to help prevent external systems from identifying the DNS server. This blocking assists in establishing security by hiding server information.

Hide Version

This setting blocks `version.server` and `version.bind` queries to help prevent external systems from discovering the DNS server version. This blocking assists in establishing security by concealing version details.

Unwanted Reply Threshold

This setting sets the maximum number of unwanted replies allowed per thread before TBF takes defensive measures. When this limit is exceeded, TBF logs a notification and takes action to assist with protection and maintain performance.

TTL for Host Cache Entries

This field controls how long the DNS server keeps host information in cache, including response times, server availability, and EDNS support data. The default is `900` seconds (15 minutes).

Cache Max TTL

This field sets the maximum time that DNS records and messages remain in cache before automatic removal. This removal helps prevent outdated information from being served. The default is `86400` seconds (24 hours).

Cache Max Negative TTL

This field determines how long negative responses (such as *domain not found* messages) stay in cache. These responses include SOA records that specify timing limits. The default is `3600` seconds (1 hour).

Number of Queries per Thread

This setting specifies how many DNS queries each processing thread can handle at the same time. The default is **1024**.

Note: Higher values allow more concurrent processing but use more system resources.

Jostle Timeout

This setting sets the timeout period used when the server is under heavy load. This value allows enough time for 1 complete communication cycle with authoritative DNS servers. The default is **200** milliseconds.

Harden DNSSEC Data

This setting enforces strict DNSSEC validation for trusted zones. When enabled, zones without proper DNSSEC data are marked as invalid, and queries to these zones are unsuccessful. This enforcement assists in establishing security but blocks some legitimate queries.

Log Level

This setting controls the amount of detail in log messages. The default is **1**.

Note: Higher numbers produce more detailed logs, which are useful for diagnostics but impact performance.

Log Queries

This setting records every DNS query in the system logs. For monitoring DNS activity and diagnostics, enable this feature.

Note: This feature generates large log files.

SafeSearch

This setting automatically redirects search engine queries to their filtered search versions, helping filter inappropriate content. This feature is disabled by default and primarily useful in family or educational environments.

Rebind Protection

This setting helps prevent DNS rebinding attacks by blocking responses that point to private IP addresses. This feature is enabled by default and helps protect against malicious websites attempting to access internal network resources.

Custom Host Entries

This field helps ensure you add specific host or domain mappings that override normal DNS resolution. Enter these entries in standard DNS format, and TBF applies them directly to the server configuration.

Custom Options

This field provides a way to add advanced configuration parameters that are not available through the standard interface. Use this field for specialized settings that require direct configuration file entries.

Additional information

For detailed information about these settings, refer to the Unbound documentation at *Unbound*.

Host overrides

Host overrides let you create custom DNS entries that take precedence over standard DNS resolution for specific hostnames.

When you set up a host override, TBF uses the custom entry instead of querying external DNS servers.

Use host overrides to:

- Direct specific hostnames to internal servers within the network
- Block access to websites by redirecting them to a nonfunctional IP address
- Set up local DNS entries for internal services and applications
- Replace public DNS records with custom values

Domain overrides

Domain overrides let you assign specific DNS servers to particular domains, creating backup options and custom DNS resolution paths.

Domain overrides let you assign specific DNS servers to particular domains. When you define the same domain multiple times, TBF uses all specified DNS servers together. This creates redundancy, so if one server is inoperable, others can still handle DNS requests.

This feature helps when you need to:

- Use dedicated DNS servers for internal company domains
- Direct DNS queries for certain domains through trusted servers that assist in establishing security
- Distribute DNS traffic across multiple servers for better performance
- Maintain DNS service availability even when some servers go offline

DNS firewall

DNS firewall helps provide an effective method to block access to unwanted websites by returning fake IP addresses for blocked domains.

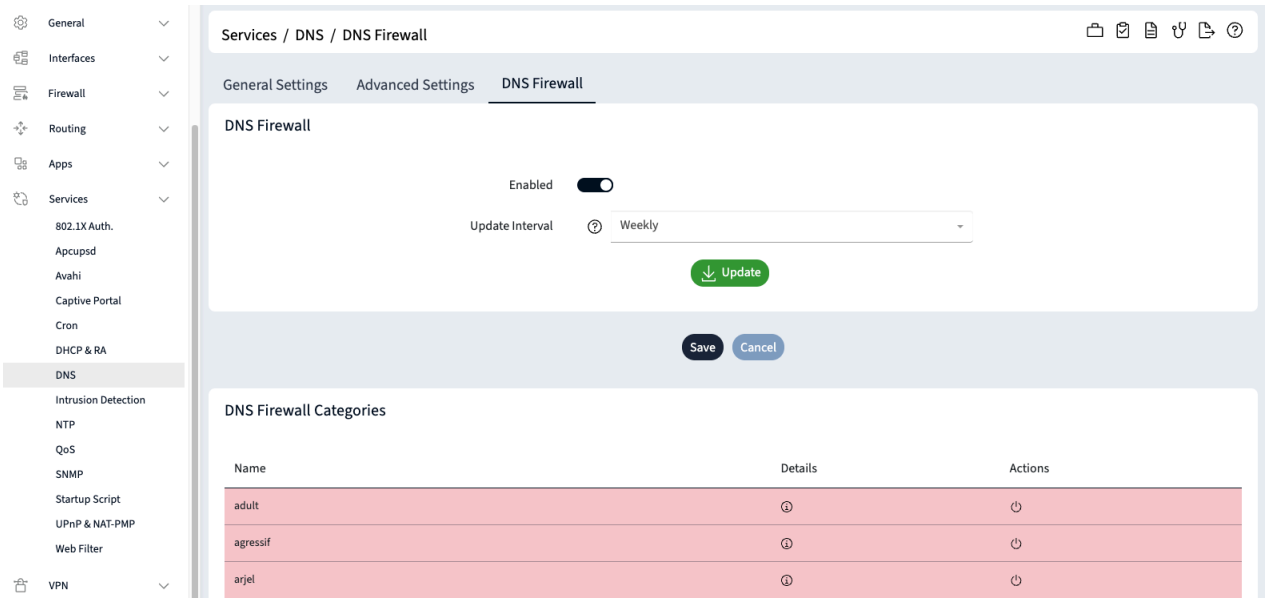


Figure 188: DNS firewall

Most web traffic today uses encryption, which makes traditional content filtering difficult. DNS blocklisting offers an effective solution by intercepting domain name requests and redirecting blocked websites to fake IP addresses before users can access them.

TBF helps provide ready-to-use domain lists organized by threat categories, making it easy to help protect against common security threats. Also create custom blocklists to address the specific needs of the organization by adding custom domains and hostnames.

TBF updates blocklists weekly by default through the **Update Interval** setting. Adjust this frequency to daily or monthly updates based on security requirements and network policies.

DNS firewall components

The DNS firewall consists of two main components that work together to help protect the network:

- *DNS firewall categories* on page 268: Ready-made blocklist categories that target specific threat types such as malware, phishing, and inappropriate content
- *Custom blocklists* on page 269: User-defined lists where you can add specific domains and hostnames that require blocking

DNS firewall categories

Blocklist categories enable you to block hosts based on different threat types and content categories.

Default categories

TBF enables the following categories by default:

- cryptojacking
- ddos
- malware

Enable or disable each category individually based on security needs.

Category details

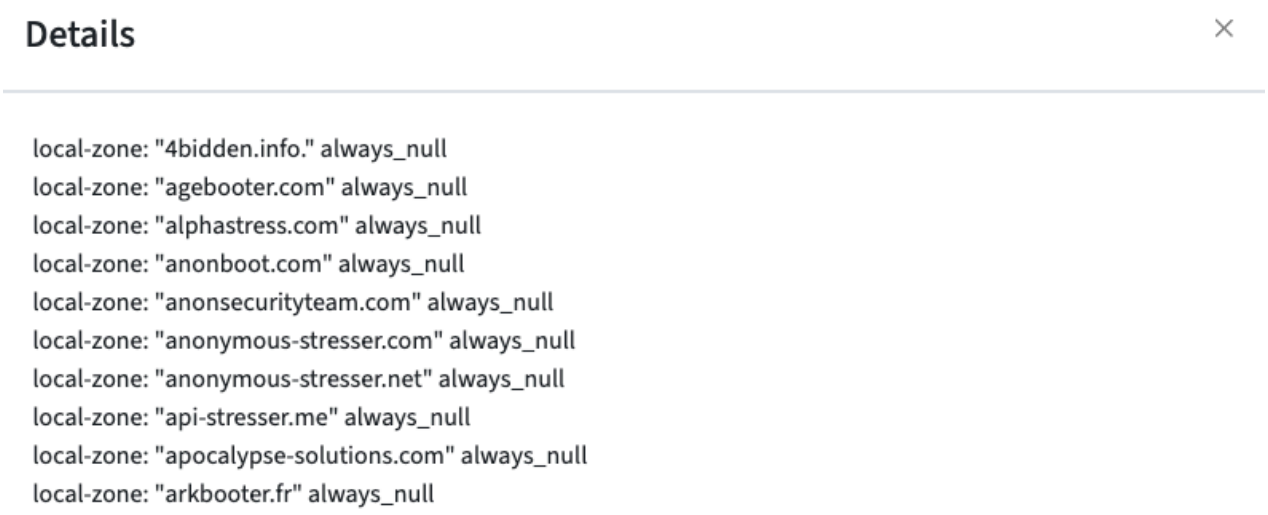


Figure 189: DNS blocklist details

Each category includes a **Details** icon that displays comprehensive information about the blocklist, including the specific hosts and IP addresses that are blocked.

Custom blocklists

Create multiple custom blocklist entries by specifying hostnames or domains that you can enable or disable independently.

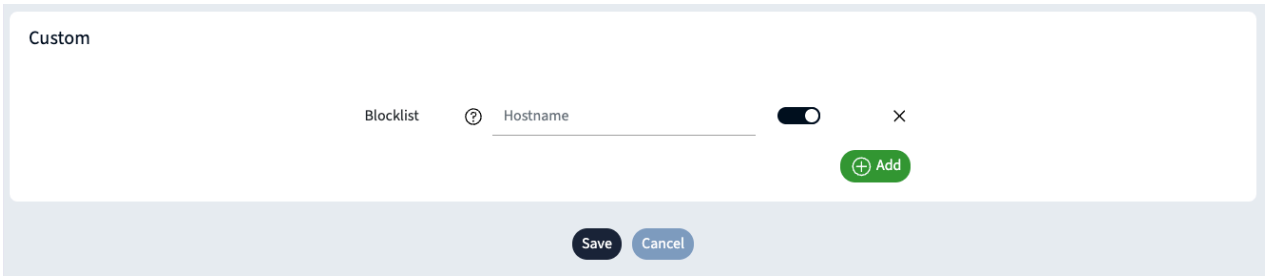


Figure 190: Custom blocklist

You can create multiple custom blocklist entries by specifying hostnames or domains. You can enable or disable each custom entry independently from the predefined categories.

Subdomain blocking

Tip: When you add a domain to the blocklist, TBF automatically includes all subdomains of that domain.

Redirect DNS requests

Redirect all DNS requests to the device by creating a DNAT rule.

Redirect all DNS requests to the TBF device and block access to external DNS servers, particularly those outside the network.

Perform the following steps:

1. Navigate to **Firewall > NAT > DNAT**.
2. Select **Add**.
3. Set the following settings for the redirect rule:
    a. Select the **Interface** of the network where you want to apply the rule, for example, **LAN**.
    b. Select **TCP/UDP** as the **Protocol** to cover both DNS query types
    c. Select the appropriate **Address Family** (IPv4 or IPv6) for which you want the rule to be active
    d. Enable **Invert IP Match** and select **LAN_Address** with **Port** DNS (53) as the destination. This configuration targets DNS requests that do not route to the local DNS server.
    e. Select **Localhost** and DNS (53) as the **Redirect IP** and **Port**. This setting redirects all DNS requests to the TBF device.
    f. Enable **Associate Firewall Rule** and set **NAT Reflection** to **Disabled** to help prevent potential routing issues.

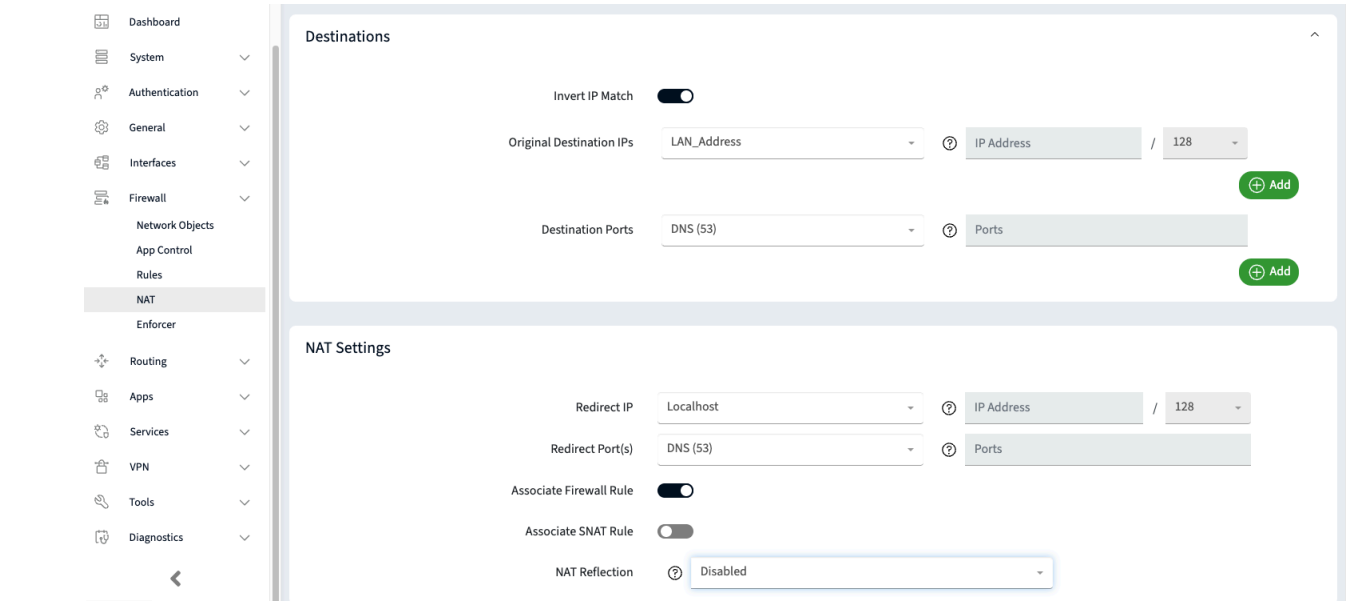


Figure 191: Request redirect

Tip: By blocking external DNS requests, you effectively use the DNS firewall blocklist feature across the entire network without needing to set up each device individually.

Create wildcard domain DNS entries

Create wildcard DNS entries using the **Custom Host Entries** field to handle subdomain queries.

To create a wildcard DNS entry such as `*.subdomain.domain.com`, you can use the **Custom Host Entries** field.

Perform the following steps:

1. Access the DNS advanced settings and locate the **Custom Host Entries** field.
2. Add the wildcard configuration using the following format:

```
local-zone: "subdomain.domain.com." redirect
                        local-data: "subdomain.domain.com. A 192.168.1.10"
```

3. Replace the IP address with the host that the system returns for subdomain queries.

DNS troubleshooting

Resolve common DNS issues related to private IP addresses and multiple WAN connections.

Common DNS issues

DNS servers may encounter issues with private IP addresses and multiple WAN connections that require specific configuration adjustments.

DNS rebind attack messages

When using a forward server that returns private IP addresses, the DNS server rejects these responses. The server interprets them as a DNS rebind attack.

Disable security check for specific domains

Perform the following steps:

1. Access the DNS advanced settings.
2. In the **Custom Host Entries** field, declare the domains as private using the format:

```
private-domain: "mydomain.ending"
```

DNS query timeouts and delays

If you use an internet load balancer or multiple WAN connections simultaneously, the system cannot resolve DNS requests correctly. This results in DNS query timeouts and delays.

Set up upstream DNS server

Perform the following steps:

1. Set up an upstream DNS server in the DNS settings.
2. Enable the **DNS Forwarding** mode.

HAProxy

HAProxy is a high-availability load balancer and proxy server that distributes incoming TCP and HTTP traffic across multiple servers.

Overview

This helps provide that applications remain available even if individual servers become inoperable.

Before you use HAProxy, install it first. Navigate to **System > Addons** to install the HAProxy package.

After installation, find the HAProxy settings under **Services > HAProxy**.

Configuration topics

HAProxy configuration includes the following key areas:

- *General settings* on page 272: Basic HAProxy configuration parameters and global options
- *Backend settings* on page 273: Server configuration that defines the actual servers behind the load balancer
- *Frontend settings* on page 276: Service configuration that handles incoming client connections
- *Advanced settings* on page 278: Additional frontend configuration options for complex scenarios
- *Error page settings* on page 279: Custom error page configuration for better user experience
- *Access control lists settings* on page 280: ACL rule configuration to control traffic flow
- *Actions settings* on page 280: Action configuration that defines what happens when ACL conditions match

Related tasks

- *Set up DNAT for HAProxy port redirection* on page 281: Set up port redirection when nginx already uses ports 443 and 80

General settings

This topic covers basic HAProxy configuration options for enabling the service and setting global parameters.

Overview

Enable or disable the HAProxy service as needed for the network configuration.

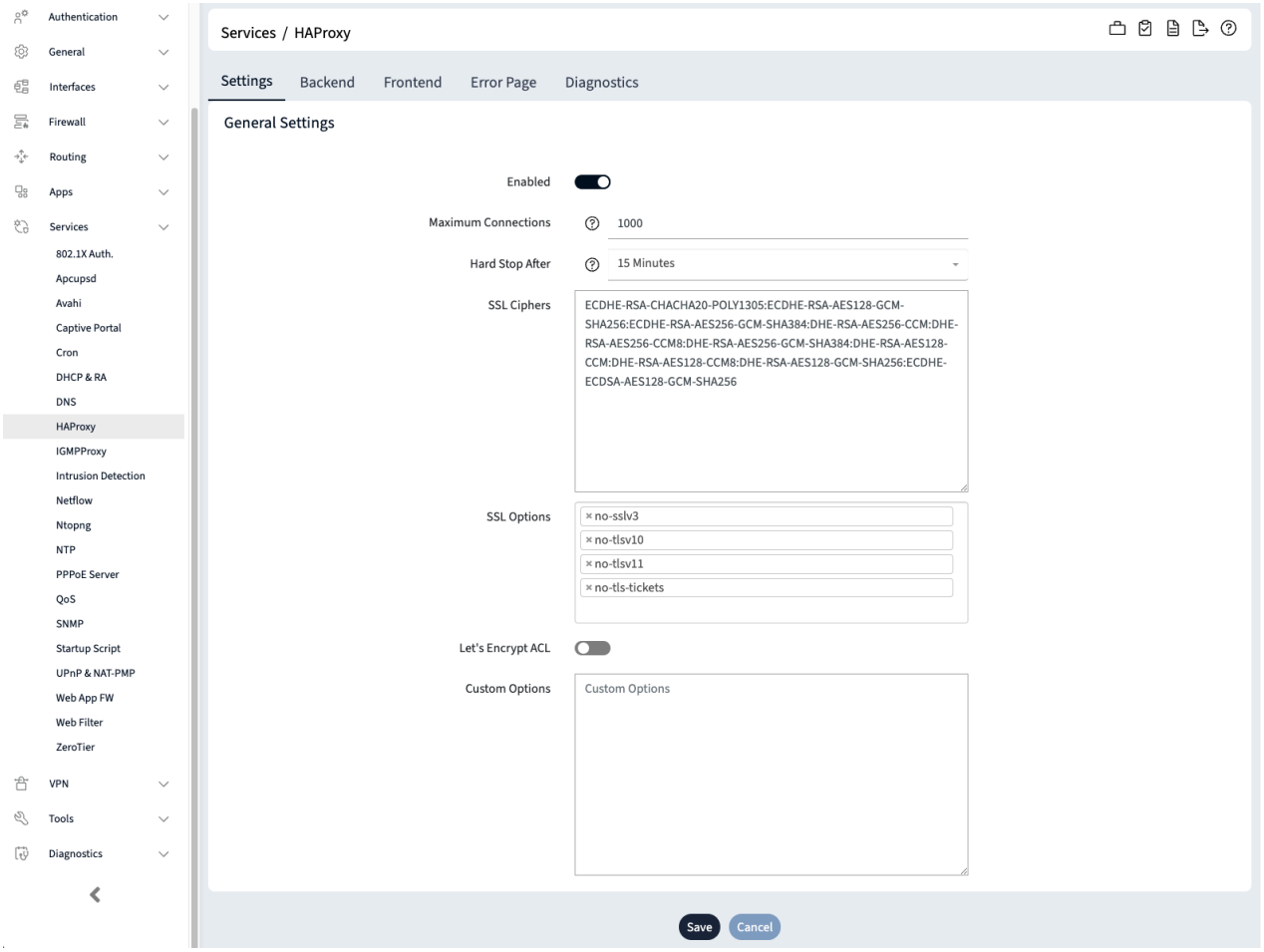


Figure 192: HAProxy

When you enable HAProxy, it automatically starts with 1 thread per CPU core. This design helps distribute incoming connections efficiently across the processing power of the system.

Settings

Maximum Connections

This field sets the maximum number of simultaneous connections that HAProxy handles. The default is `1000` connections. Adjust this value based on the expected traffic load and server capacity.

Hard Stop After

This setting specifies the maximum time (in seconds) that you allow for HAProxy to perform a clean shutdown. This setting helps provide that the service stops completely, even if some connections remain open during the shutdown process. This helps prevent the service from hanging indefinitely.

SSL Ciphers

This field contains a colon-separated list of SSL cipher suites that HAProxy uses for encrypted connections. These ciphers determine the encryption methods available for communications that assist in establishing security.

SSL Options

This setting lets you disable specific SSL/TLS protocol versions for purposes that assist in establishing security. Select from these options:

- **no-sslv3**: This option disables SSL version 3.
- **no-tlsv10**: This option disables TLS version 1.0.
- **no-tlsv11**: This option disables TLS version 1.1.
- **no-tls-tickets**: This option disables TLS session tickets.
- **no-tlsv12**: This option disables TLS version 1.2.
- **no-tlsv13**: This option disables TLS version 1.3.

Let's Encrypt ACL

This setting redirects Let's Encrypt certificate validation requests to the TBF firewall. Enable this setting when TBF manages SSL certificate renewal using web-based validation mode.

Custom Options

This field provides a text field where you can enter additional HAProxy configuration parameters that the standard interface options do not cover. Use this for advanced configurations specific to the environment.

Export settings

To export the current settings as an Excel spreadsheet, use the export option in the upper-right corner of the interface.

Backend settings

This topic describes the backend server configuration for HAProxy load balancing and health checking.

Overview

A backend is a server behind the firewall that receives data from HAProxy. The frontend receives incoming data, processes it, and forwards it to a backend.

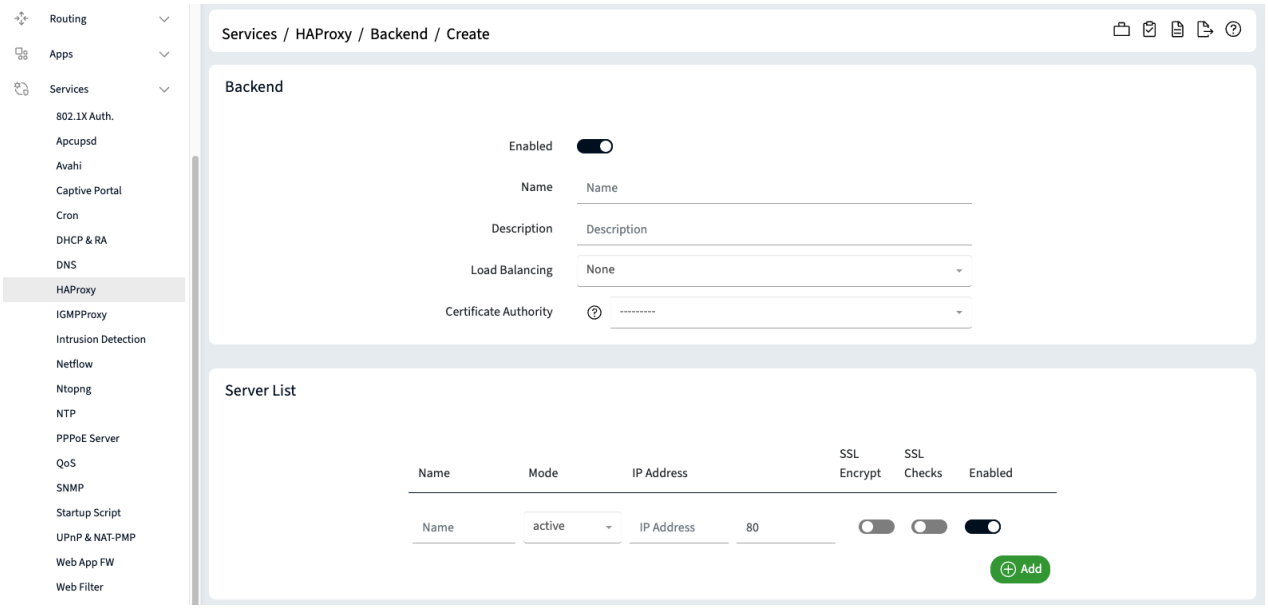


Figure 193: Backend and server list

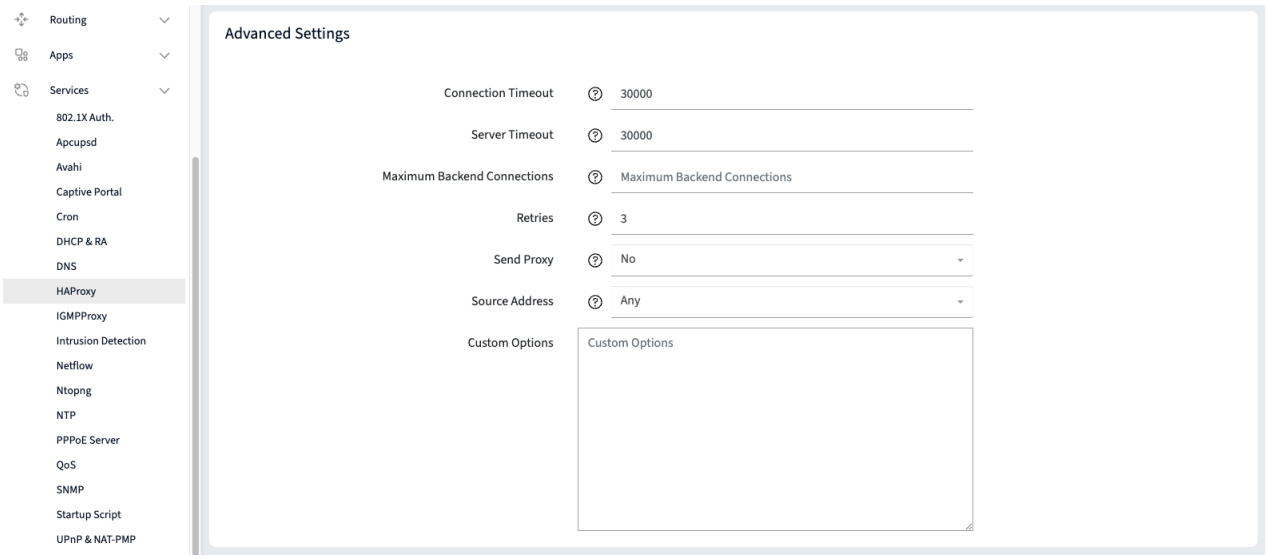


Figure 194: Advanced settings

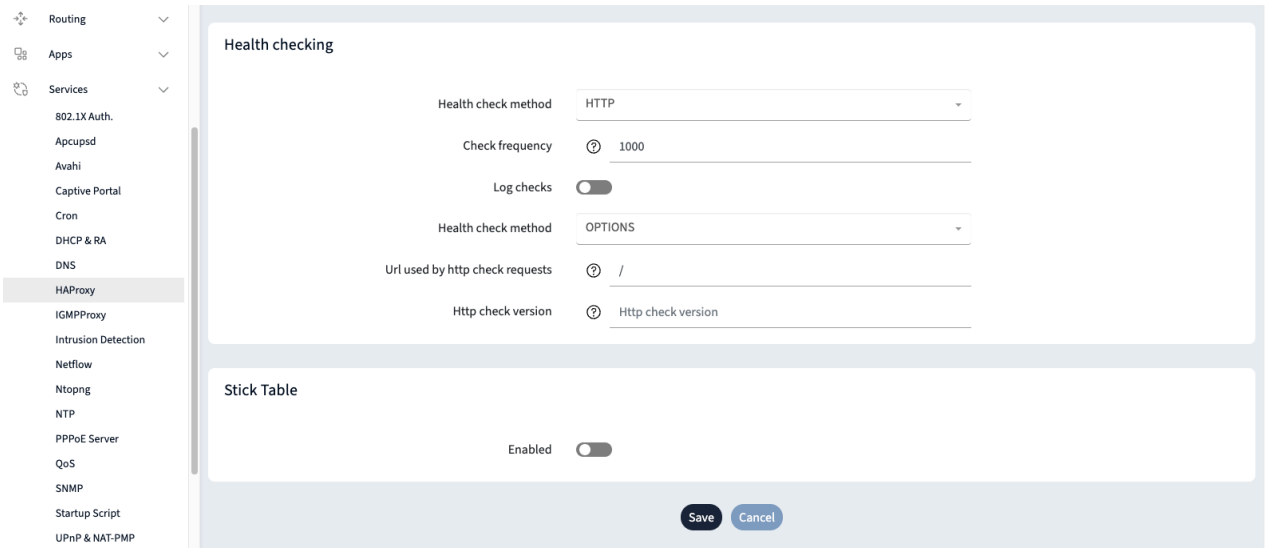


Figure 195: Health checking and stick table

Each backend has a **Name**, **Description**, and you can enable or disable it. You can add multiple **Servers** to each backend for load balancing. Each server includes settings for name, mode, IP address, port, SSL encryption, and SSL checks.

Server settings

SSL Encrypt

This setting enables SSL encryption on connections that this listener creates.

SSL Checks

This setting forces SSL encryption for health checks, regardless of whether the server uses SSL for normal traffic.

Load balancing settings

Load Balancing

This setting determines the load balancing method. Select from:

- **None**
- **Round Robin**
- **Static Round Robin**
- **Least Connections**
- **Source**

Connection Timeout

This field specifies the maximum time in milliseconds to wait for a connection attempt to a server to succeed. The default is `30000`.

Server Timeout

This field defines the maximum inactivity time in milliseconds on the server side. The default is `30000`.

Maximum Backend Connections

This field specifies the maximum number of connections that backends process. If you leave this field empty, TBF calculates the value based on the global maximum connection setting.

Retries

This field determines how many times the system retries a connection attempt when a server refuses the connection or times out. The default is `3`.

Send Proxy

This setting determines whether the proxy protocol connects to the backend and, if so, which version to use.

Source Address

This setting changes the source of the connection from HAProxy to the firewall. This is useful in high-availability setups to use the virtual IP address as the sender.

Custom Options

This field specifies custom configuration parameters.

Health checking settings

Health check method

This setting determines the health check method. Select from:

- **None**
- **Basic**
- **HTTP**
- **SMTP**
- **LDAP**

Check frequency

This field specifies how often health checks run, measured in milliseconds. The default is `1000`.

Log checks

This setting logs any changes to the health check status or server health.

HTTP Method

This setting determines the HTTP method used for health checks. Select from:

- **OPTIONS**
- **HEAD**
- **GET**
- **POST**
- **PUT**
- **DELETE**
- **TRACE**

URL used by http check requests

This field specifies the URL for HTTP health check requests. If left blank, it defaults to `/`.

Http check version

This field specifies the HTTP version for health checks. If left blank, it defaults to `HTTP/1.0`.

Stick table settings

Stick Table

This setting uses stick tables for session persistence.

Type

This setting specifies the stick table type.

Size

This field determines the stick table size in MB. The default is `1` MB.

Expire

This field specifies when stick table entries expire, measured in seconds. The default is `10` seconds.

Frontend settings

This topic describes the frontend listener configuration for HAProxy incoming connections and SSL offloading.

Overview

A frontend is the service where HAProxy listens for incoming connections. Each frontend listens on a specific IP address and port combination.

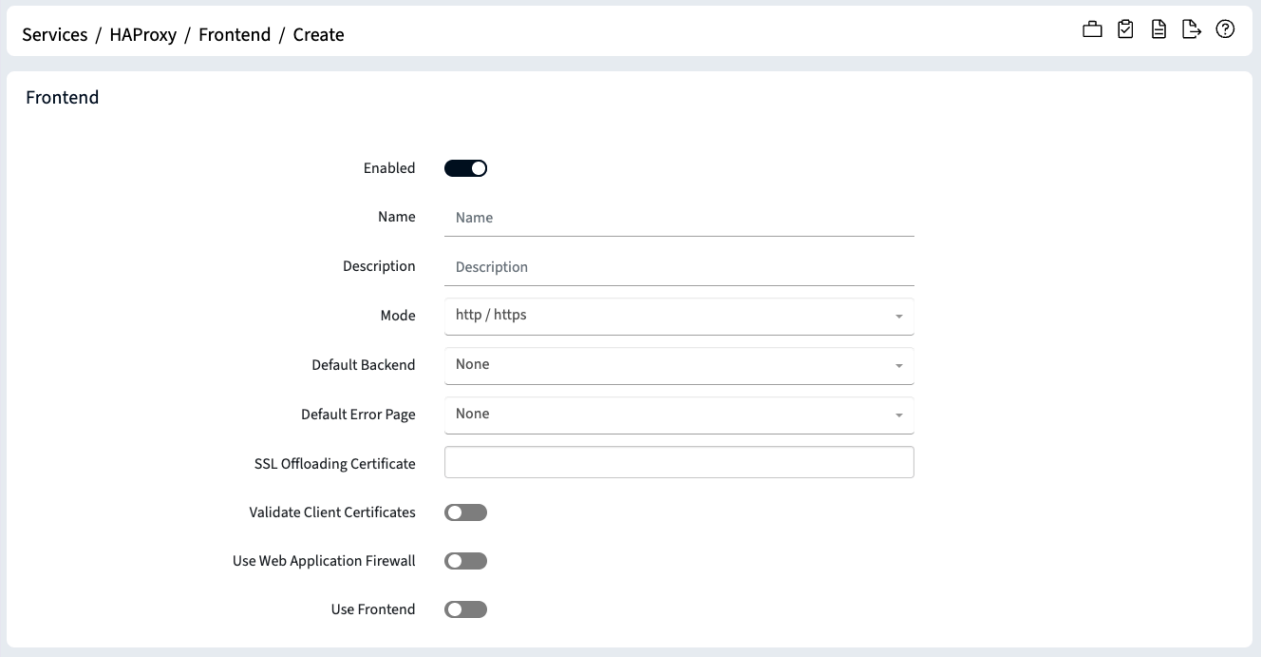


Figure 196: Frontend

Each frontend has a **Name**, **Description**, and you can enable or disable the frontend.

A frontend can operate independently or connect to another frontend. When connecting frontends, only the ACL and actions are available for configuration. The system inherits other settings from the connected frontend.

A frontend can listen on multiple addresses, each with its own IP address and port. You can also set up multiple ACLs with descriptions, expressions, backends, and values. Negate values and enable case sensitivity checking.

Settings

Mode

This setting sets the mode. Select from:

- **http / https**
- **ssl / https (TCP mode)**
- **tcp**

Default Backend

This setting specifies the default backend connection when you do not select a specific backend in subsequent options.

Default Error Page

This setting specifies the default error page. Create the error page in the system before selecting the error page here.

SSL settings

SSL Offloading Certificate

This setting enables SSL offloading for connections. HAProxy can use multiple certificates, selecting the certificates based on the Common Name (CName) and incoming requests. To enable SSL offloading, explicitly enable the IP address and port settings, even when you select certificates.

Note: SSL offloading only works with HTTP connections, not TCP or TLS connections.

Validate Client Certificates

This setting validates client certificates. When you enable this setting, select a **Certificate Authority**.

Note: The **Validate Client Certificates** feature is only available in **http / https** and **tcp** modes with SSL offloading enabled for external addresses.

You can also set **Client Certificate Verification** as either **Required** or **Optional**.

External addresses

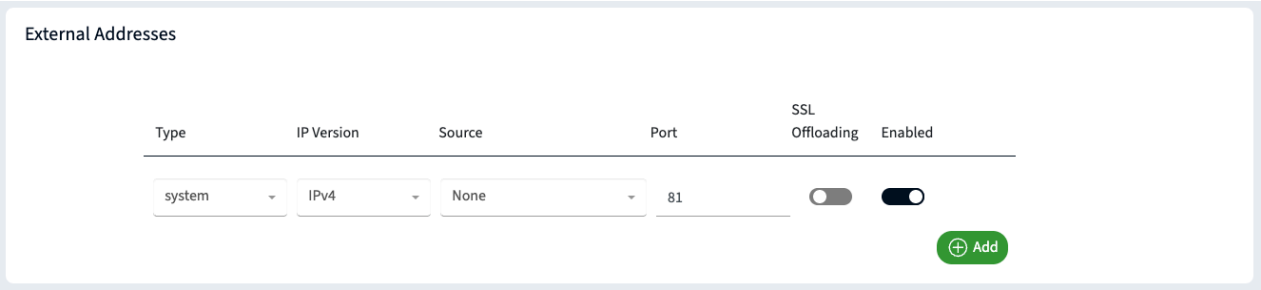


Figure 197: External addresses

Each frontend can listen on one or more IP addresses and ports. Set the type (IPv4 or IPv6) and the address to use. **System** addresses are network objects from the firewall. You can enable **SSL Offloading** for each address and port combination.

Web application firewall

Web Application Firewall

To use the **Web Application Firewall** feature with encrypted connections, enable **SSL Offloading**. This setting transparently handles encryption while helping provide WAF access to the traffic.

Use Web Application Firewall

This setting enables the web application firewall for this frontend. The WAF inspects traffic and blocks potential threats with a 404 error response.

Frontend linking

Use Frontend

This setting specifies and connects to another **HAProxy Frontend**. To simplify management of different frontends, you can link one frontend to another. The frontends share settings except for ACLs and actions. The system merges the frontends in the configuration to create a unified frontend. Frontend linking is useful for scenarios like managing multiple complex backends or organizing configurations by customer.

Advanced settings

This topic covers advanced configuration options for HAProxy frontend listeners including timeouts and connection handling.

Overview

The **Advanced Settings** section provides additional configuration options that apply to the entire frontend. These settings help you fine-tune performance, security, and connection behavior for the HAProxy frontend.

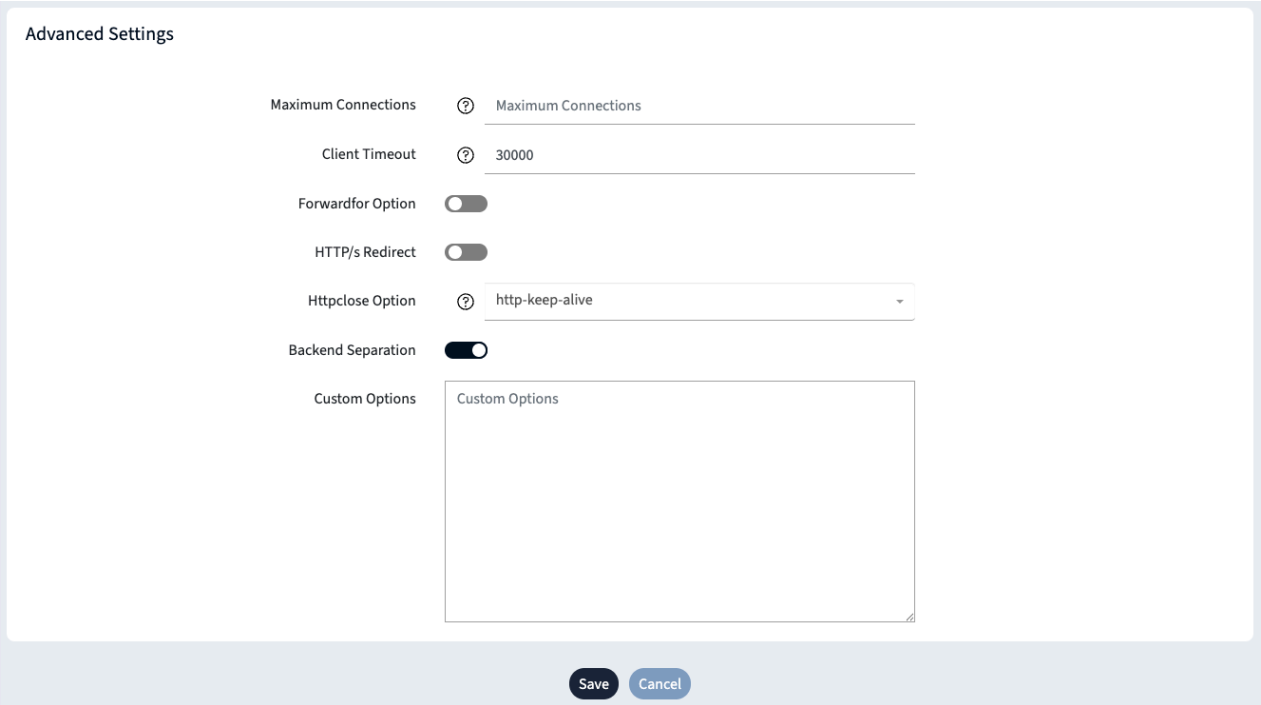


Figure 198: Advanced settings

Settings

Maximum Connections

This setting sets the maximum number of simultaneous connections that the frontend handles. When connections reach this limit, HAProxy queues or rejects new connection requests, which helps prevent server overload.

Client Timeout

This field specifies how long in milliseconds HAProxy waits for client activity before closing an inactive connection. The default is `30000` milliseconds (30 seconds). Adjust this value based on the typical response times of the application.

Forwardfor Option

When you enable this setting, HAProxy adds the `X-Forwarded-For` header to requests that HAProxy sends to backend servers. This header contains the IP address of the original client, which allows backend servers to identify the real client behind the proxy.

HTTP/s Redirect

This setting automatically redirects HTTP traffic to HTTPS, which helps provide connections that assist in establishing security. To enforce encrypted communication for client requests, enable this setting.

Httpclose Option

This setting forces HAProxy to close HTTP connections after each request-response cycle instead of keeping the connections open for reuse. Closing connections helps with certain applications that do not handle persistent connections well.

Backend Separation

When you use access control lists (ACLs) with hostname matching, this setting creates individual backend copies for each hostname. HAProxy then performs health checks using the specific hostname for each backend. This isolation helps provide that if one hostname experiences issues, only that specific service becomes unavailable while other hostnames on the same backend remain accessible.

Custom Options

This field allows you to add custom HAProxy configuration directives that the standard interface options do not cover. Enter valid HAProxy configuration syntax in this field.

Error page settings

This topic covers custom error page configuration for HAProxy with supported HTTP status codes.

Overview

You can set up custom error pages and connect them to specific actions and access control lists (ACLs). When creating error pages, include the HTML content that users see. Also include the proper HTTP status code with response headers that browsers expect.

To help you get started, you can download a sample error page file directly from the web interface.

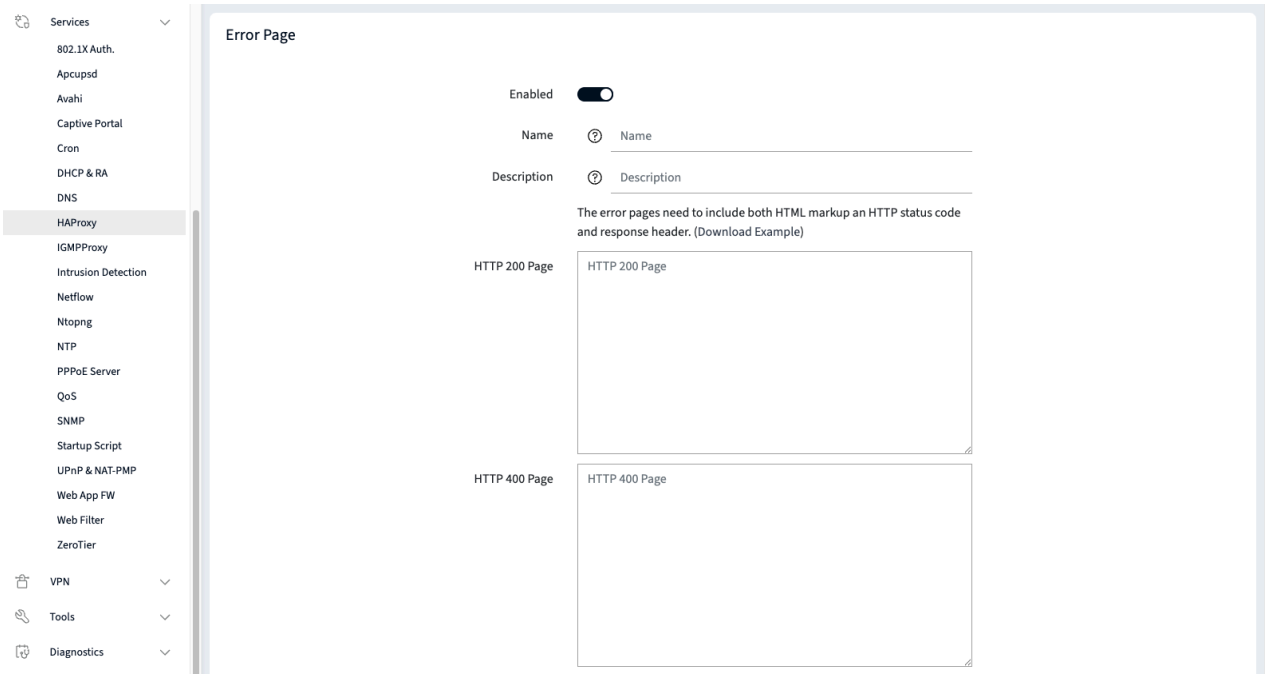


Figure 199: Error page

Supported HTTP status codes

HAProxy supports the following HTTP status codes for custom error pages:

- `200`: OK (successful request)
- `400`: Bad Request
- `401`: Unauthorized
- `403`: Forbidden
- `404`: Not Found
- `405`: Method Not Allowed
- `407`: Proxy Authentication Required
- `408`: Request Timeout
- `410`: Gone
- `425`: Too Early
- `429`: Too Many Requests
- `500`: Internal Server Error
- `502`: Bad Gateway
- `503`: Service Unavailable
- `504`: Gateway Timeout

Access control lists settings

ACL rule configuration for matching conditions in HAProxy frontends and backends.

Overview

Access control lists (ACLs) are conditional rules that determine how HAProxy handles incoming traffic in frontends and backends. These rules evaluate specific criteria to make routing decisions. In frontends, ACL criteria vary based on the mode you set up and can include elements like hostnames, source IP addresses, and request headers. Each ACL requires a unique name within the frontend configuration. This name allows you to reference the ACL when you define actions and routing behaviors.

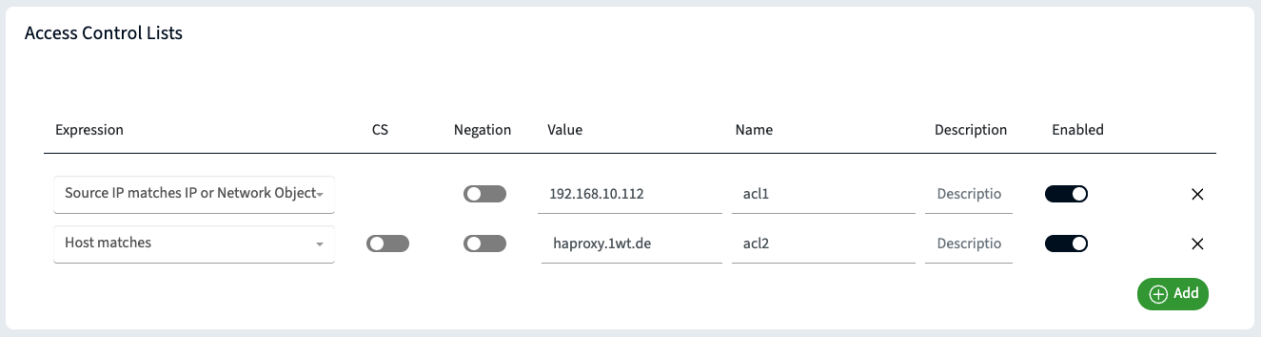


Figure 200: Access control lists

Actions settings

This topic describes the action configuration for linking to ACLs in HAProxy frontends and backends.

Overview

You can link actions to one or multiple ACLs. You can combine multiple ACLs with `AND` or `OR` logic to create flexible assignments. Frontend and backend actions offer different specific actions.

You can assign **Error Page** to a frontend action or set it as the default error page in a frontend. Create error pages in the system before you assign them.

You can assign **Error Page Backend Down** to a frontend action. The system displays this page when the selected backend is unavailable.

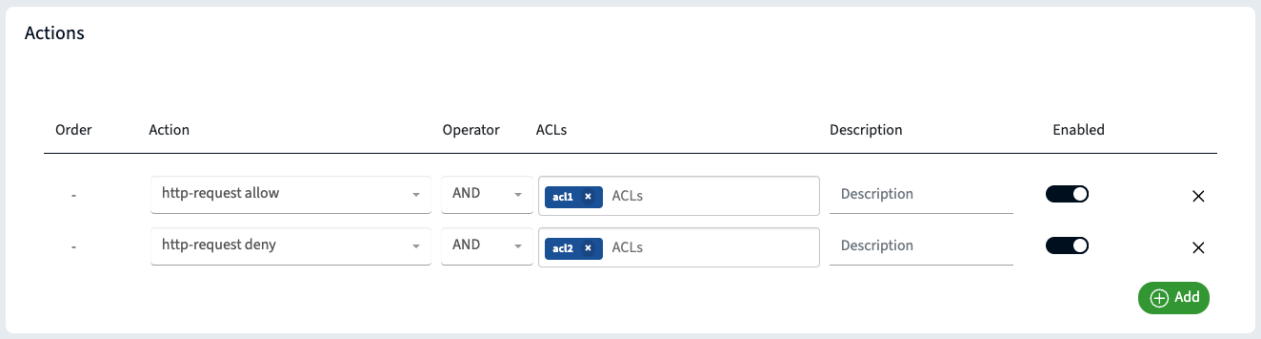


Figure 201: Actions

Set up DNAT for HAProxy port redirection

This topic explains how to set DNAT rules to redirect ports 443 and 80 from WAN to HAProxy when nginx uses these ports.

Note: Be aware that nginx uses ports `443` and `80`. To use these ports for HAProxy, set up DNAT on the interfaces. This redirects traffic to a different port that the HAProxy frontend uses.

Redirect ports `443` and `80` from WAN to HAProxy. HAProxy needs to run on different ports, for example, `444` and `81`.

Perform the following steps:

1. Navigate to the DNAT configuration.
2. Create a DNAT rule to redirect port 443 to port 444.
3. Create a DNAT rule to redirect port 80 to port 81.
4. Set up the HAProxy frontend to listen on ports 444 and 81.

DNAT rule configuration example

The following example shows a DNAT rule configuration:

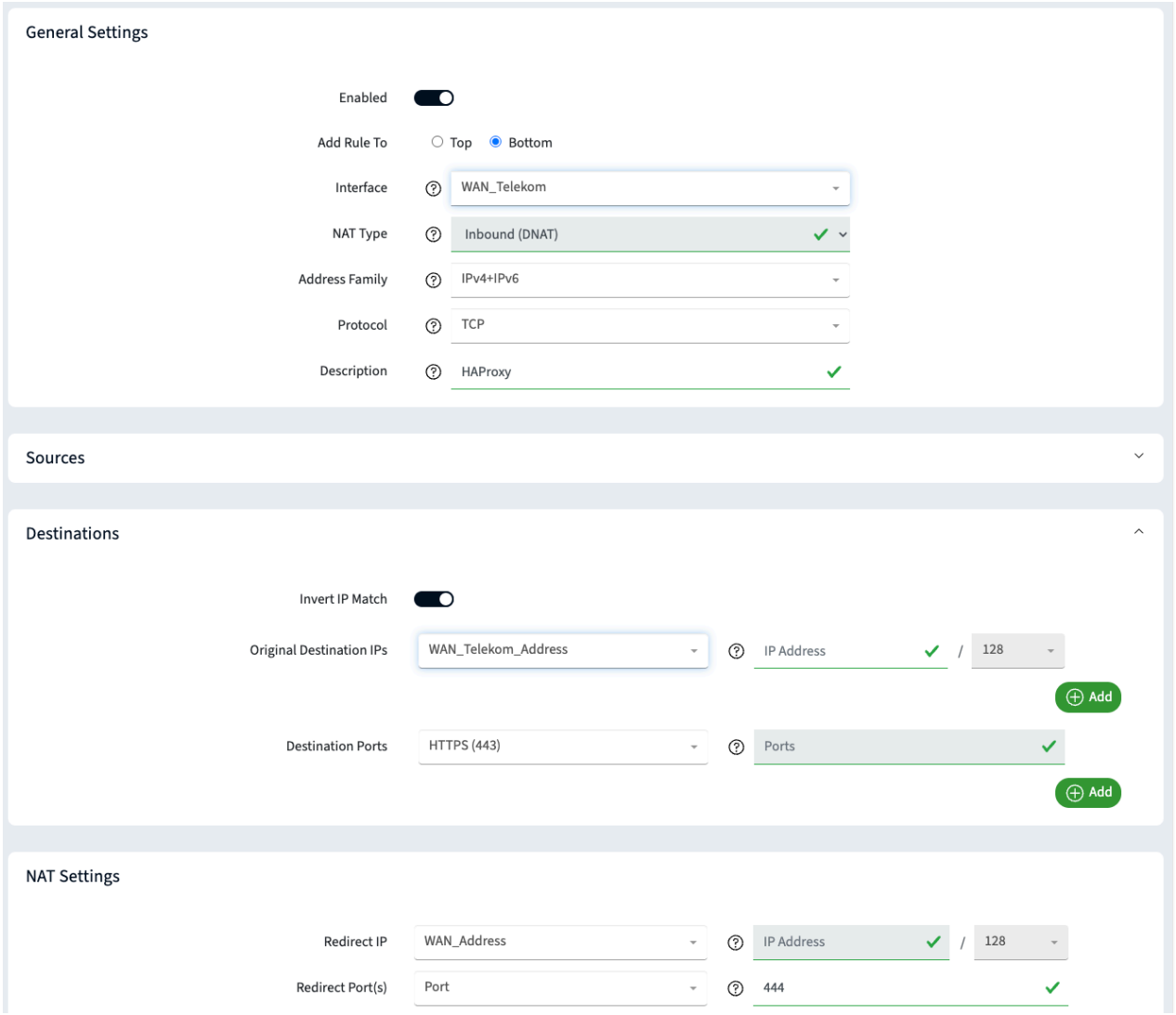


Figure 202: Redirect ports 443 and 80

IGMPProxy

IGMPProxy is a service that forwards multicast traffic between different network segments. This service allows devices on separate network segments to receive the same multicast content.

IGMPProxy is a service that forwards multicast traffic between different network segments. This service allows multicast data streams to cross layer 2 network boundaries, and enables devices on separate network segments to receive the same multicast content.

Find the IGMPProxy settings under **Services > IGMPProxy**.

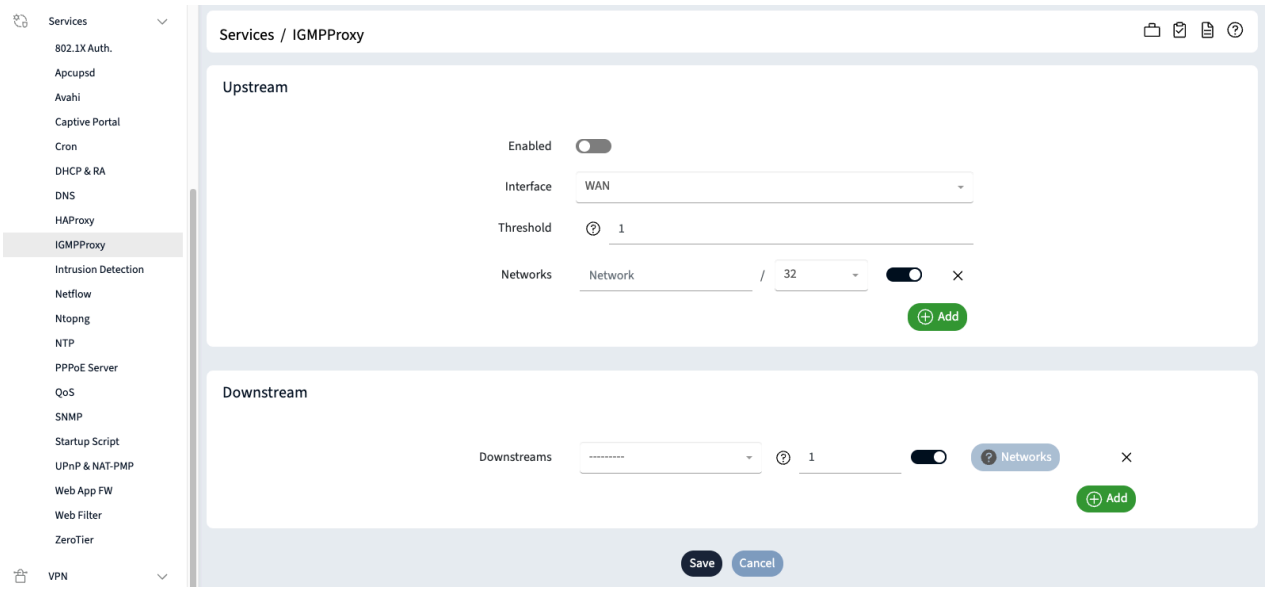


Figure 203: Igmpproxy

Before you set up IGMPProxy, install the service. To install the service, navigate to **System > Addons** and add the IGMPProxy package.

After you install the service, set up 2 main components for IGMPProxy:

- *Upstream settings* on page 283: Set up the interface that receives multicast traffic.
- *Downstream settings* on page 283: Set up the interfaces that forward multicast traffic to clients.

Upstream settings

To control multicast traffic routing from downstream devices to multicast sources, set up the upstream interface settings for IGMPProxy. To manage multicast packet forwarding and help prevent routing loops, set the network interface, TTL threshold, and authorized network segments.

To control multicast traffic routing, enable or disable the IGMPProxy functionality.

Interface

This setting selects the network interface that handles upstream traffic (traffic that flows toward the multicast source). This interface receives multicast requests from downstream devices and forwards them to the multicast source.

Note: The same interface cannot serve as both upstream and downstream interface.

Threshold

This field sets the time-to-live (TTL) threshold value that determines how far multicast packets travel. TBF drops packets with a TTL value below this threshold, which helps prevent routing loops and limits multicast traffic scope. The default is `1`. This value means that packets travel only 1 hop.

Networks

This field defines the network segments that receive multicast traffic. To add multiple network segments, specify the IP address and netmask combination of each network, for example, `192.168.1.0/24`.

Downstream settings

This topic provides information on setting up multiple downstream interfaces for the IGMP proxy, each with customizable interface and threshold settings. It also explains how to define specific network segments to control which client devices can communicate through each downstream interface.

You can set up multiple **Downstream** interfaces, each with its own interface and threshold value settings.

When you save a downstream interface setup, you can define specific network segments. These segments control which client devices the IGMP proxy permits to communicate through the interface.

Note: The upstream interface cannot serve as a downstream interface.

Intrusion detection

The intrusion detection system uses Suricata, an open-source intrusion detection system (IDS) and intrusion prevention system (IPS).

Find the intrusion detection settings under **Services > Intrusion Detection**.

The intrusion detection system monitors network traffic for suspicious activities and potential security threats. It uses Suricata, an open-source tool that functions as both an IDS for monitoring and alerting, and an IPS for actively blocking threats.

Configuration areas

The intrusion detection system offers several configuration areas to help you customize security monitoring for the network:

- *General settings* on page 284: Set up basic intrusion detection options, which include the network interfaces to monitor, logging preferences, and core security settings.
- *Performance settings* on page 286: Adjust memory allocation and processing parameters to help ensure performance based on network traffic volume.
- *Update rule packages* on page 288: Set up automatic updates to keep intrusion detection rules current with the latest threat signatures.
- *Manage intrusion detection rules* on page 289: Enable, disable, or customize specific rule files and categories to match security requirements.
- *Example Suricata configuration* on page 290: Review sample configurations and recommended practices to help you implement effective intrusion detection for the environment.

General settings

This reference describes the basic intrusion detection settings, which include interfaces, logging, and security options.

Find the intrusion detection settings under **Services > Intrusion Detection**.

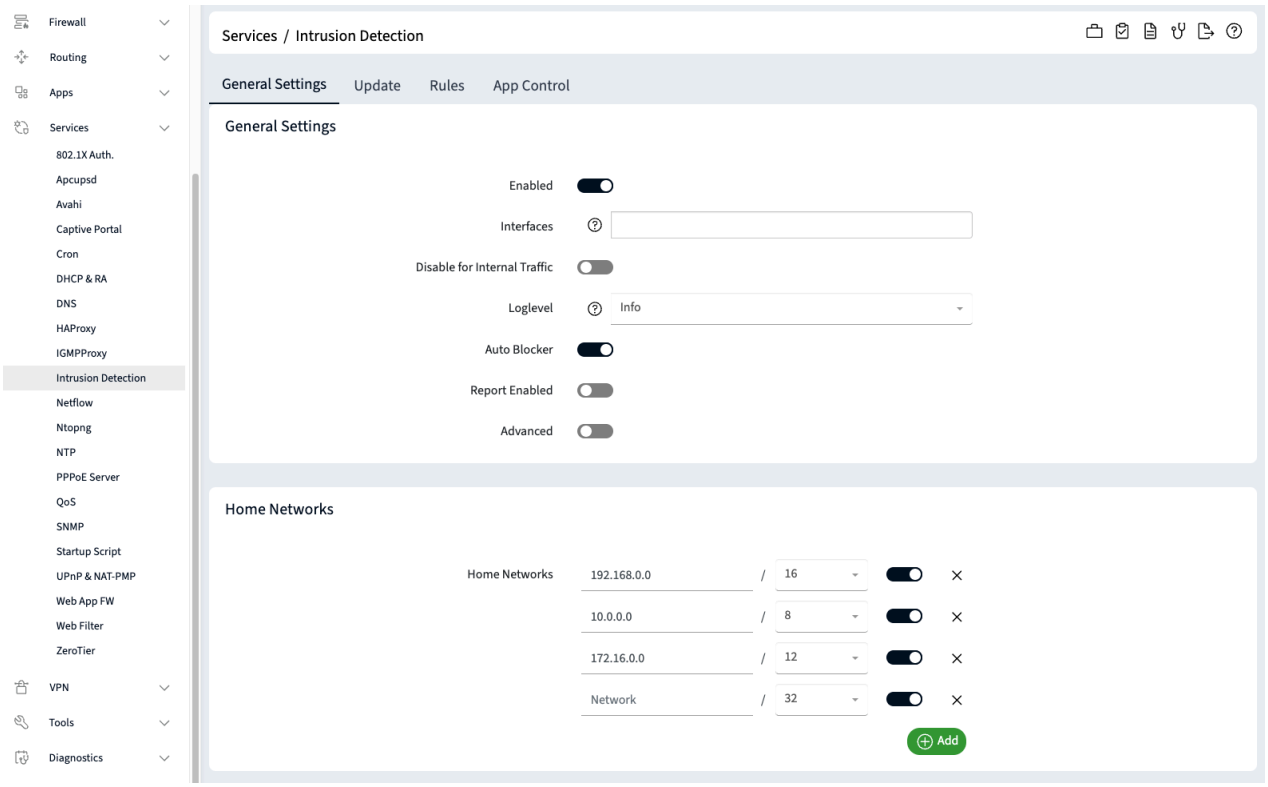


Figure 204: Intrusion detection

Use this page to enable or disable intrusion detection for the network.

Interfaces

This setting lets you select one or more network interfaces where Suricata monitors and analyzes traffic for security threats.

Disable for Internal Traffic

This setting disables intrusion detection for private network IP addresses. This action improves performance because it focuses security monitoring on internet-facing traffic and skips internal network communications.

Loglevel

This setting controls how much detail appears in the system logs. The default is **Info**, which helps provide a balanced amount of information.

Auto Blocker

This setting automatically blocks IP addresses that trigger security event rules, which helps provide immediate assistance in the protection against detected threats.

Note: TBF automatically blocks IP addresses for 60 seconds when they are not part of the **Home Networks** and match any of these security categories:

- Attempted Denial of Service
- Denial of Service
- A Network Trojan was detected
- Detection of a Denial of Service Attack
- Web Application Attack
- Exploit Kit Activity Detected
- Successful Credential Theft Detected
- Crypto Currency Mining Activity Detected
- Malware Command and Control Activity Detected

Report Enabled

To receive regular security summaries, enable email reporting. When enabled, specify a **Report Email** address to receive reports. Verify that you first set up an email server in the system settings. Reports include details about security alerts and blocked events during the specified time period.

Advanced

This setting enables advanced options, which include **Pass Last** and **Custom Options**, for experienced users who need fine-tuned control.

Pass Last

This setting changes the order in which security rules are processed. By default, TBF checks *allow* rules before *block* rules. This setting reverses that order.

IPS Check Firewall

This setting monitors traffic directed to and from TBF itself. This option is disabled by default.

Note: Use this setting carefully, as it may block access to the web interface or other firewall services. Instead, use strong firewall rules to help protect TBF.

IPS on Bridge

This setting monitors traffic that passes through network bridges at the data link layer (layer 2). This setting affects only traffic that bypasses normal firewall filtering, such as traffic on VLANs or other interfaces built on top of the bridge.

Home Networks

To define the internal network ranges, specify IP addresses and ports in this field. TBF includes these default private networks:

- 192.168.0.0/16
- 10.0.0.0/8
- 172.16.0.0/12

Security rules use home networks to distinguish between internal and external traffic and apply appropriate assistance in the protection policies.

Export settings

To export the current settings as an Excel spreadsheet, use the export option in the upper-right corner of the interface.

Performance settings

This topic covers how to optimize intrusion detection processing, set memory and performance parameters.

Intrusion detection performance depends on how you allocate memory resources. Higher memory allocation typically improves processing speed but uses more system resources.

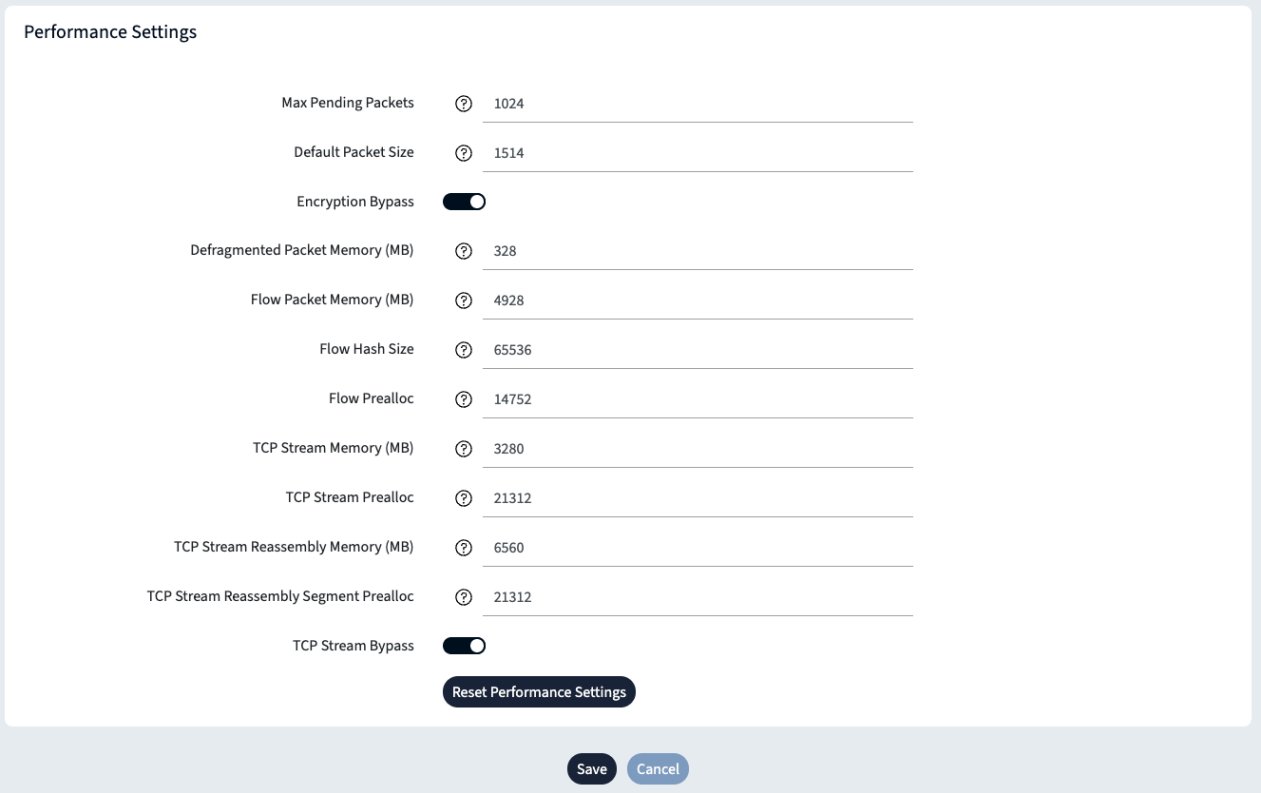


Figure 205: Performance settings

Max Pending Packets

This setting controls how many packets TBF processes simultaneously, ranging from 1 to hundreds of thousands.

Note: Higher values improve performance but consume more RAM. Lower values reduce memory usage but slow processing. Based on the available memory and performance requirements, specify the value.

Default Packet Size

This setting sets the expected packet size for the network traffic. When TBF encounters larger packets than this setting, the system still processes the packets but with reduced performance. For optimal results, match this setting to the typical network packet sizes.

Encryption Bypass

This setting skips detailed analysis of encrypted traffic after initial inspection, because encrypted data provides limited security insights. Enabling this setting significantly improves performance and frees resources to analyze more meaningful traffic.

Defragmented Packet Memory (MB)

This setting sets the maximum memory available to reassemble fragmented network packets. TBF reconstructs fragmented packets before analysis occurs.

Flow Packet Memory (MB)

This setting determines memory allocation for the flow engine, which tracks network connections. More memory speeds up connection tracking and analysis.

Flow Hash Size

This setting controls the hash table size for flow tracking.

Note: Larger hash tables improve lookup performance but require more memory. Based on the network connection volume, adjust the size.

Flow Prealloc

This setting maintains a pool of pre-allocated flow structures in memory to help prevent system overload during traffic spikes. This pre-allocation reduces processing delays when new connections arrive.

TCP Stream Memory (MB)

This setting sets memory allocation for TCP stream processing. Higher values enable faster analysis of TCP connections, which typically represent the majority of network traffic.

TCP Stream Prealloc

This setting specifies how many TCP session structures each processing thread keeps ready in memory. Pre-allocation reduces delays when the system establishes new TCP connections.

TCP Stream Reassembly Memory (MB)

This setting allocates memory to reconstruct TCP data streams from individual packets. More memory enables faster reassembly of complex data flows.

TCP Stream Reassembly Segment Prealloc

This setting sets the number of reassembly segments each processing thread keeps ready. Pre-allocated segments speed up the reconstruction of fragmented TCP streams.

TCP Stream Bypass

This setting automatically stops deep inspection when either side of a TCP connection reaches its analysis depth limit. While this bypass misses some traffic details, the bypass helps prevent performance bottlenecks and TBF enables the setting by default.

Reset Performance Settings

This setting automatically calculates recommended values for performance settings based on the available system memory. Use these values as a starting point, then fine-tune individual settings based on specific network requirements.

Update rule packages

This topic describes how to set up automatic updates for intrusion detection rule packages to keep security rules current.

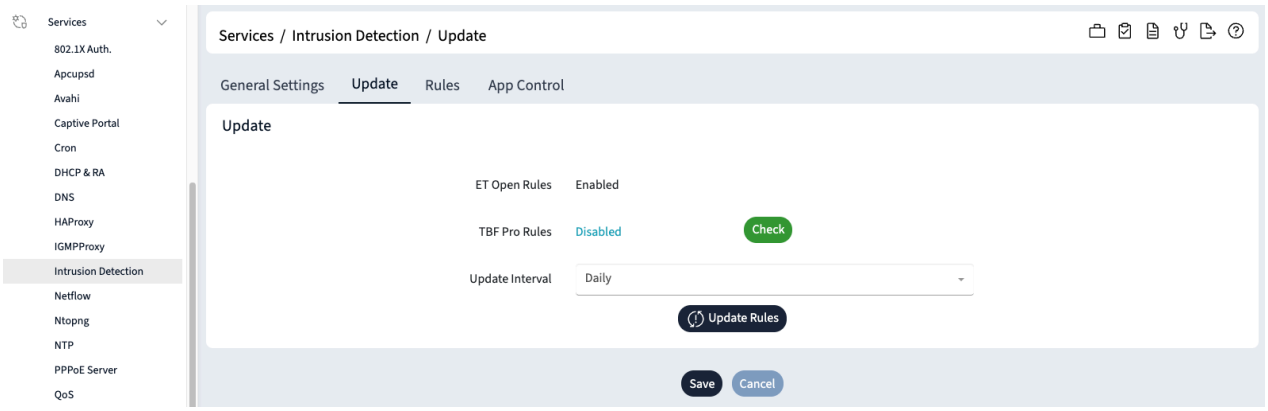


Figure 206: Update

Specify which rule sets to use for intrusion detection. If you have purchased TBF Pro Rules, the device automatically retrieves the licence from the Belden Portal. View both the licence details and expiration date in the interface. If you do not have Pro Rules, TBF uses the **ET Open Rules** by default.

The TBF Pro Rules package includes the **ET Open Rules** plus additional premium rules.

Perform the following steps:

1. Navigate to **Services > Intrusion Detection > Update**.
2. Set the **Update Interval** to the preferred schedule.
   This creates a scheduled task that automatically downloads and applies rule updates at the specified interval.

TBF now automatically updates the intrusion detection rules according to the schedule you set.

Important: Keep in mind that enabling more rules slows down the performance of Suricata. Consider activating only the rules and categories that match specific security needs.

Manage intrusion detection rules

This topic describes how to set up and manage individual rule files and categories for intrusion detection.

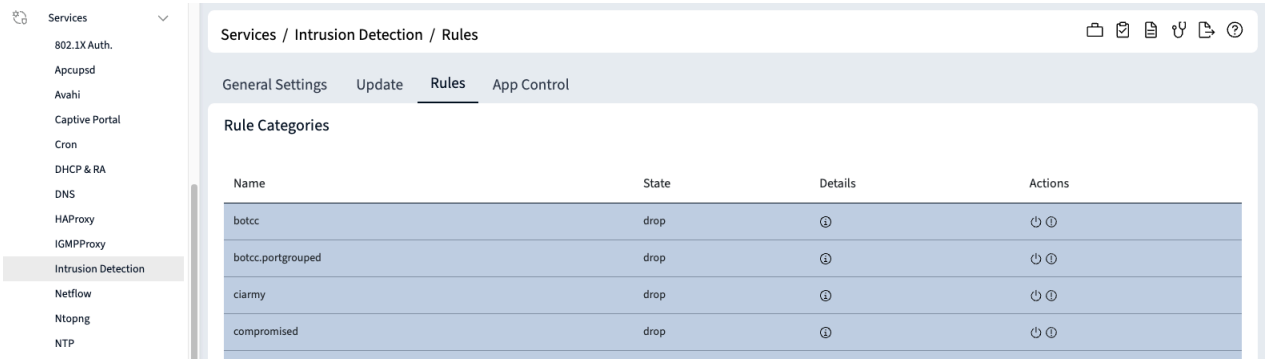


Figure 207: Rule categories

TBF displays installed rule files. Each file typically represents a single category. To view specific rule definitions, select the **Details** icon for each rule. In the **Actions** column, set each rule file to one of 3 states:

- disabled
- alert
- drop

Perform the following steps:

1. Navigate to **Services > Intrusion Detection > Rules**.
2. Review the list of installed rule files.
   Each rule file typically represents a single category of security rules.
3. To set up individual rules, select the appropriate action in the **Actions** column.
   - To deactivate the rule completely, select **disabled**.
   - To record matches in logs without blocking traffic, select **alert**.
   - To block traffic that matches the rule, select **drop**.
4. To apply changes to all rules simultaneously, use the bulk action buttons:
   - **Drop All**: Block all matching traffic
   - **Alert All**: Log all matches without blocking
   - **Disable All**: Deactivate all rules
5. To disable specific rules, enter multiple **SIDs** (signature IDs).
6. To create **Custom Rules**, use the section at the bottom of the page.

Example Suricata configuration

This topic describes rule packages and recommended configuration practices for intrusion detection.

Note: This section contains screenshots from the former VT Air GUI. These will be updated in a future version.

In TBF, select from 2 rule packages for intrusion detection. Each rule package contains different categories that help you specify which types of network traffic to block. Not every category is relevant for every environment.

Suricata includes its own default rule set. View these rules in the *official Suricata repository*.

When you enable the **ET Open Rules**, TBF displays them with the prefix `emerging-XXX` under **Services > Intrusion Detection > Rules**. For detailed descriptions of each category, refer to the *official Emerging Threats documentation*.

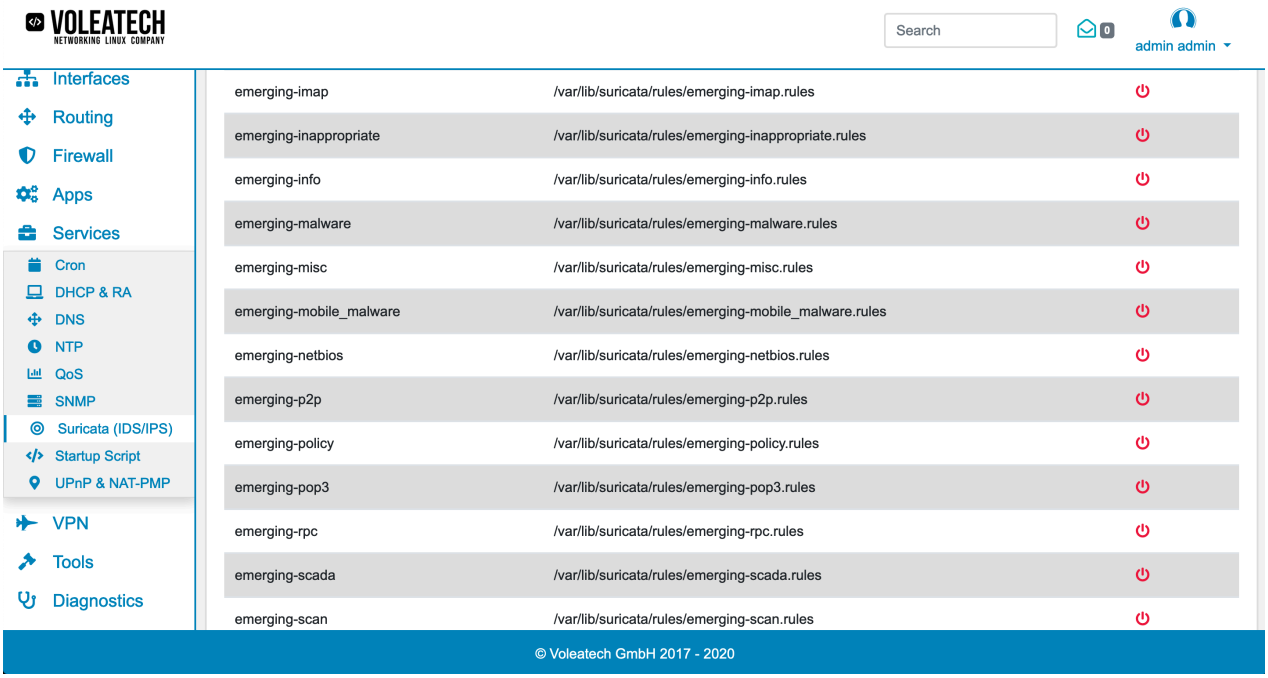


Figure 208: Emerging threats open rules

Recommended configuration

Set **Interfaces** to **any** and create a firewall rule to filter only the traffic you need to inspect. To implement this approach, create a *global firewall rule (forward and input)* on page 139 with the following settings:

- **Input Interface**: **Any**
- **Output Interface**: **Any**
- **Action**: **Match**
- **Source**: **Private Networks**
- **Destination**: **Private Networks**
- Under **Advanced Settings**: Enable **Bypass IDS/App Control**

Place this rule as the first user-created rule in the global rules list.

Netflow

Netflow is a network monitoring protocol that collects network traffic data and sends it as flow records to a designated collector for analysis.

Find the Netflow settings under **Services > Netflow**.

Netflow is a network monitoring protocol that captures information about network traffic patterns. It collects data about active network flows and forwards this information as structured records to a collector system where you analyze network performance, security, and usage patterns.

Netflow topics

The following topics contain information about implementing and using Netflow:

- *Install Netflow* on page 291: Follow the complete installation process with detailed step-by-step instructions
- *Netflow settings* on page 291: Explore configuration options, parameters, and the user interface reference

Install Netflow

This topic describes how to install the Netflow service to enable network traffic monitoring on the system.

Netflow is a network monitoring service that tracks and analyzes network traffic patterns. Install this service before monitoring network activity on the system.

Perform the following steps:

1. Navigate to **System > Addons**.
2. Find the Netflow addon in the list of available addons.
3. To begin the installation process, select the **Install** button for the Netflow addon.

Once the installation completes successfully, to manage and set up Netflow settings, navigate to **Services > Netflow**.

Netflow settings

To send network traffic data to different collectors, set up Netflow instances using the Netflow settings interface.

Overview

This page lets you create, modify, enable, or disable Netflow instances. Netflow is a network protocol that collects information about network traffic flowing through the device and sends this data to external collectors for analysis.

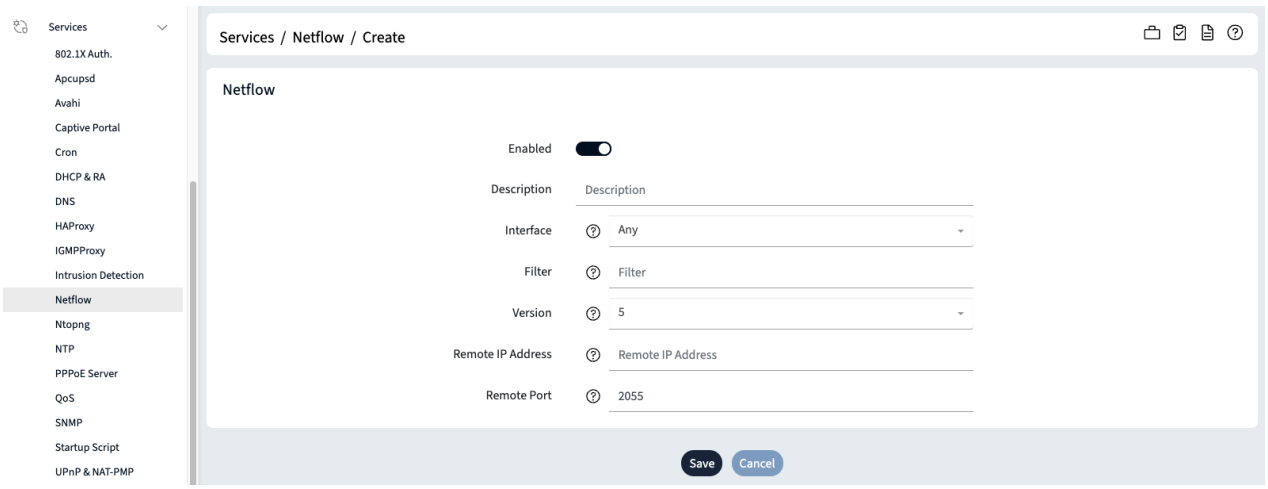


Figure 209: Netflow

Settings

Interface

To collect traffic data, select the network interface that Netflow monitors. This determines which traffic of the network connection the system analyzes.

Filter

To define which network packets to capture, enter a TCPDump-style filter expression. This allows you to focus on specific types of traffic. To capture the traffic on the selected interface, leave this field empty.

Version

To transmit data, select the Netflow protocol version. Version **5** is the default and most commonly supported version.

Remote IP Address

Enter the IP address of the remote Netflow collector server that receives and processes the flow data.

Remote Port

Specify the port number on the remote Netflow collector where the system sends data. The default is `2055`.

Ntopng

Ntopng is a network traffic monitoring tool that analyzes and tracks network usage patterns in real-time.

This tool helps you understand how network resources operate and identifies potential issues or bottlenecks.

Find the ntopng settings under **Services > Ntopng**.

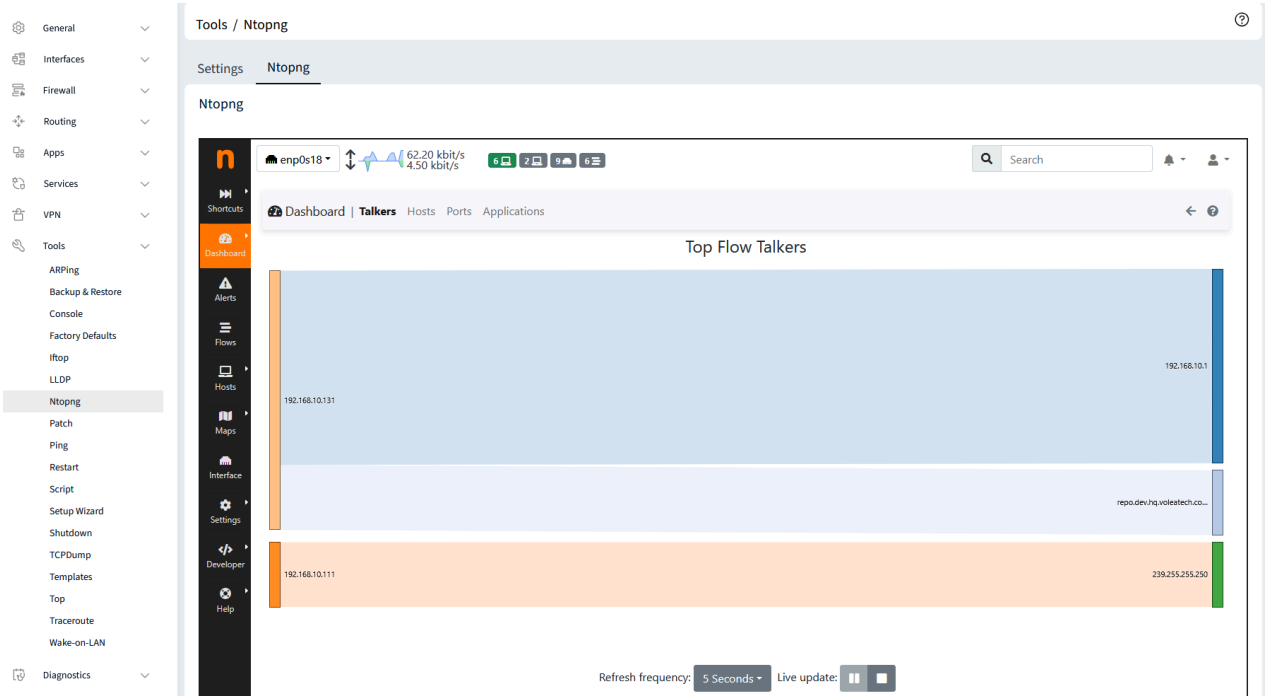


Figure 210: ntopng

Before you use ntopng, install it on the system. To install ntopng, navigate to **System > Addons** and follow the installation process.

Related topics

- *Ntopng settings* on page 292: Learn about the configuration parameters available for ntopng
- *Install ntopng* on page 294: Follow step-by-step instructions to install ntopng on the system

Ntopng settings

This topic covers configuration parameters for the ntopng network traffic monitoring tool.

General settings

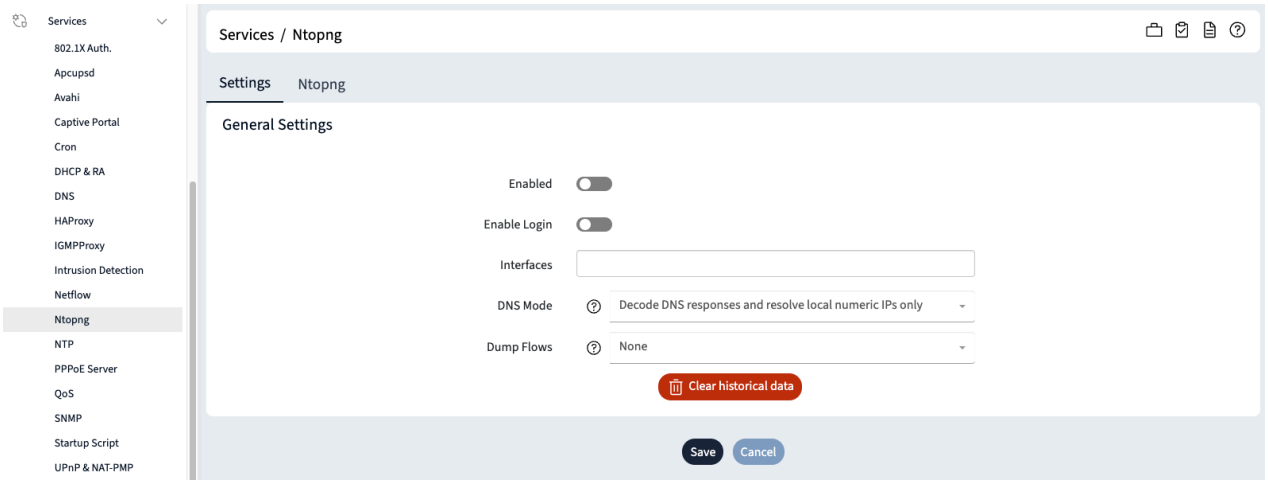


Figure 211: General

To start or stop network traffic monitoring, enable or disable ntopng.

Settings

Enable Login

When you enable this option, TBF displays the **Username** and **Password** fields. Users must provide valid credentials to access the ntopng interface.

Username

TBF uses a fixed username `admin` that you cannot change.

Password

Enter the password that users need to access the ntopng web interface.

Interfaces

Select the network interfaces where you want to monitor traffic. TBF only collects data from interfaces that you enable.

Clear historical data

To permanently delete the stored network traffic data from TBF, use this option.

Important:

You cannot undo this action.

DNS Mode options

This setting controls how the system handles DNS address resolution. Select an available mode:

- Decode DNS responses and resolve local numeric IPs only
- Decode DNS responses and resolve the numeric IPs
- Decode DNS responses and do not resolve numeric IPs
- Do not decode DNS responses and do not resolve numeric IPs

Dump Flows options

This option allows you to export network flow data to an external backend system for further analysis or storage. TBF supports these backend options:

- MySQL
- ElasticSearch
- Logstash

Install ntopng

Install the ntopng network traffic monitoring tool through the TBF addons.

To use ntopng for network monitoring, install it first. The ntopng addon provides real-time network traffic monitoring capabilities that help you analyze and track network usage patterns on the system.

Perform the following steps:

1. Navigate to **System > Addons**.
2. Find the ntopng addon in the list of available addons.
3. To install the ntopng addon, select the button.

Once the installation completes, to access ntopng settings, navigate to **Services > Ntopng**.

NTP

Network Time Protocol (NTP) synchronizes system time across the network to help provide accurate timekeeping for the device.

Find the NTP settings under **Services > NTP**.

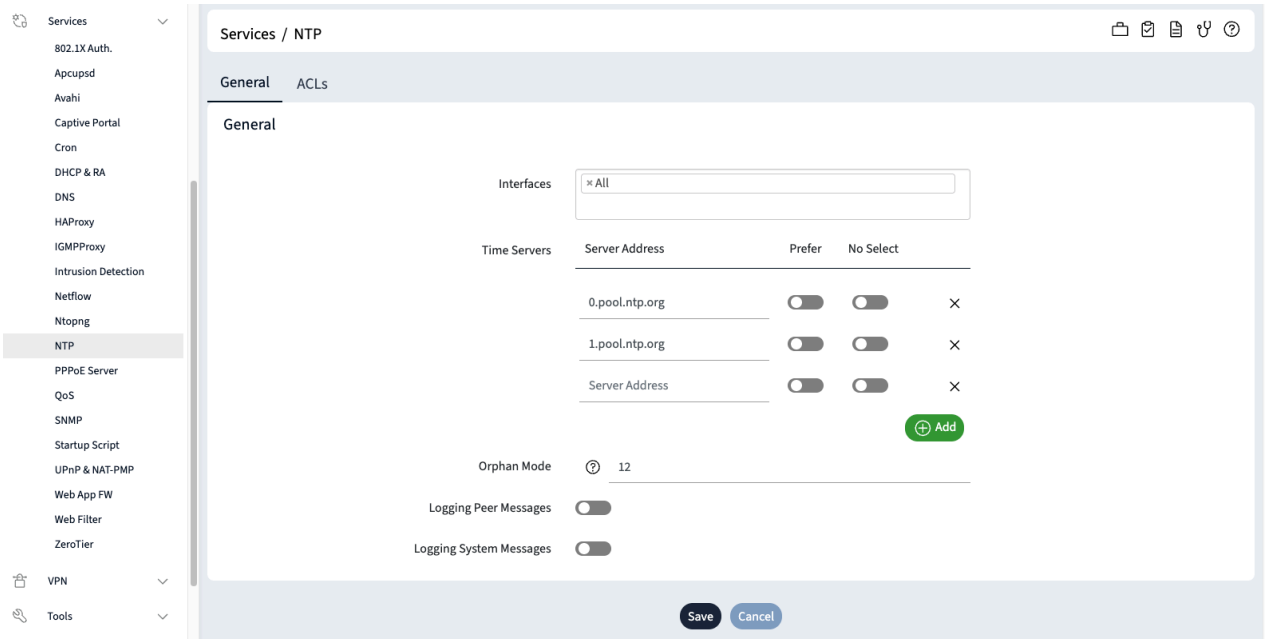


Figure 212: NTP

NTP synchronizes the system time across the network. This helps provide that devices maintain accurate and consistent timekeeping, which is essential for logging, security, and network operations.

Enable or disable the NTP service completely. Specify whether NTP uses every network interface or only selected ones. The **Orphan Mode** defaults to `12`, which allows a group of `ntpd` processes to automatically select a leader when external time sources become unreachable.

To capture additional peer log information, use **Logging Peer Messages**. To capture additional system log information for troubleshooting purposes, use **Logging System Messages**.

Add multiple time servers, each with a specific **Server Address**. When you select **Prefer**, the system gives priority to this time server over others when other settings are equal. The **No Select** option marks the server as inactive for synchronization purposes, though it remains visible for monitoring.

Related topics

For detailed access control configuration, see:

- *NTP access restrictions* on page 295
- *Custom access restrictions* on page 296

NTP access restrictions

To control client access and service availability for the NTP service, set up access control lists (ACLs) with specific restrictions.

Access control settings

To set up access control options that determine how clients interact with the NTP service, use the **ACLs** tab. These settings help you establish security for the time server and control which operations clients perform.

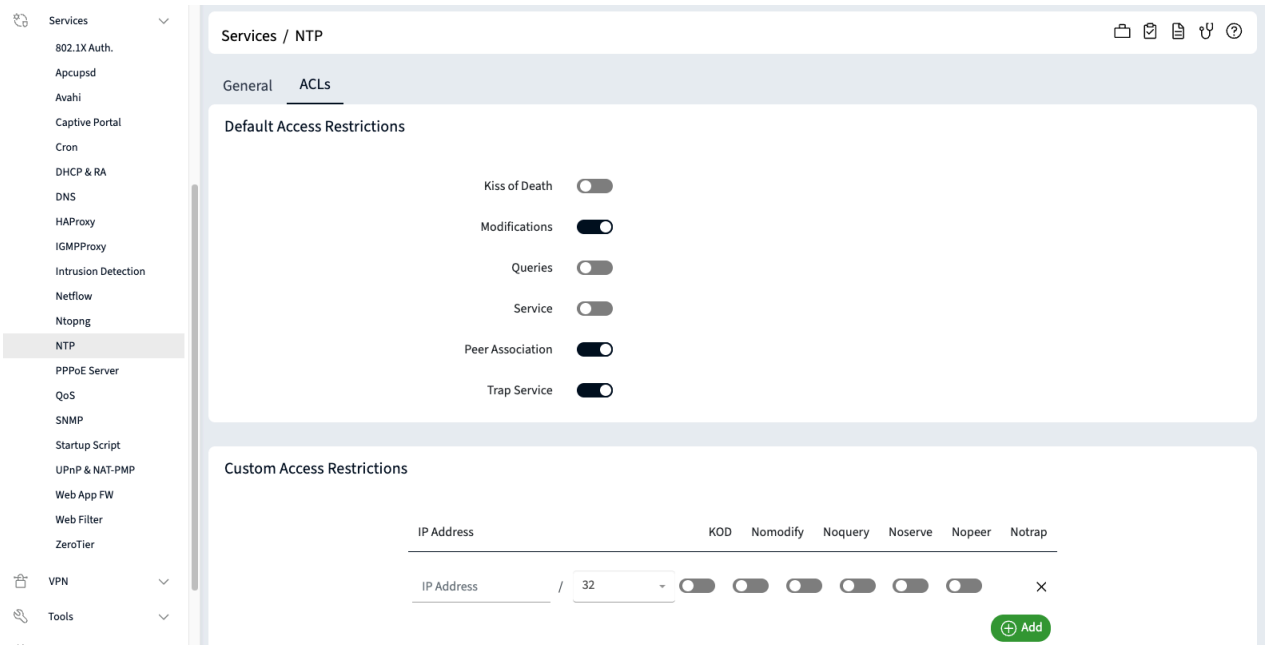


Figure 213: Access restrictions

Settings

Kiss of Death (`kod limited`)

This setting sends special packets to rate-limit aggressive clients that make too many requests.

Modifications (`nomodify`)

This setting helps prevent clients from changing server settings with `ntpq` and `ntpdc` commands.

Queries (`noquery`)

This setting blocks status and information requests from `ntpq` and `ntpdc` tools.

Service (`noserve`)

This setting stops time synchronization services but still allows status queries from `ntpq` and `ntpdc`.

Peer Association (`nopeer`)

This setting rejects requests from other servers that try to establish a peer relationship.

Trap Service (`notrap`)

This setting disables the trap service that sends unsolicited status messages to monitoring systems.

Custom access restrictions

To apply specific access restrictions for individual networks, use NTP access control options.

To apply specific access restrictions for individual networks, use the options described in *NTP access restrictions* on page 295.

This feature allows you to create detailed control policies that apply different restriction levels to different network segments or client groups. For example, allow full access to internal networks and restrict external networks to time queries only.

QoS

Quality of service (QoS) enables bandwidth management and traffic prioritization on a per-interface basis.

Find the QoS settings under **Services > QoS**.

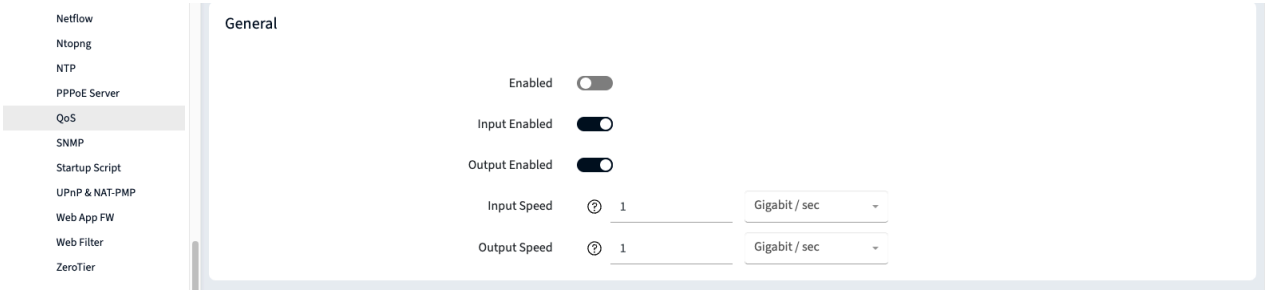


Figure 214: QOS

Interface-based QoS setup

Enable and set up QoS on each base interface individually. This means VLANs require traffic shaping on their underlying base interface. Without this setup, the system counts bandwidth twice for each VLAN that runs on top of the base interface.

To manage VLANs properly:

- Create different classes on the base interface
- Enable QoS for input and output traffic separately

Traffic-shaping mechanisms

QoS typically shapes only traffic that leaves an interface. To shape traffic that enters the interface, TBF creates a virtual interface named `INTNAME-ifb`. Traffic entering the interface passes through this virtual interface, which allows TBF to shape incoming traffic as well.

Bridge member limitation: For bridge members, set only the output speed limit.

Traffic direction and configuration

Note: The meaning of input and output depends on the interface type:

- **WAN interfaces**: Output is traffic leaving to the ISP (upload), while input is traffic coming from the ISP (download)
- **LAN interfaces**: Input is traffic from clients (upload), while output is traffic going to clients (download)

Because TBF shapes traffic as it either arrives on an interface (**INPUT**) or leaves an interface (**OUTPUT**), create different configurations. To shape WAN traffic effectively, enable QoS on the WAN interface only, with **INPUT** for download shaping and **OUTPUT** for upload shaping.

QoS topics

The following topics contain detailed information about QoS configuration:

- *QoS classes* on page 297: Class parameters and bandwidth settings
- *QoS assignment* on page 298: How to assign QoS in firewall rules
- *Limiter* on page 298: Alternative approach using firewall rule limiters

QoS classes

Quality of service (QoS) uses classes to organize traffic management, with each class having customizable names and priority levels that you can enable or disable individually.

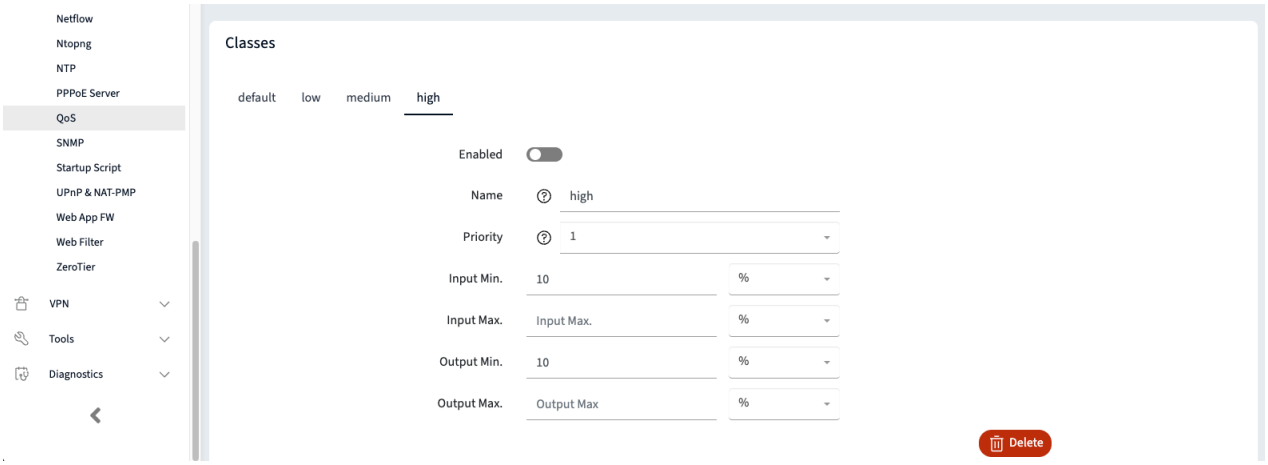


Figure 215: QOS classes

QoS uses classes to organize traffic management. Each class has a **Name** that you can customize, and you can enable or disable each class individually. All interfaces share the same classes. When you create a class on one interface, it automatically appears on all other interfaces. TBF supports up to 30 classes. You can set different priority and speed settings for each class on each interface.

Each class has a **Priority** level. When bandwidth is available, TBF gives preference to classes with higher priority values. TBF enforces priority strictly, so it is usually better to assign the same priority to all classes and control traffic using minimum and maximum bandwidth values. This approach helps prevent high-priority classes from consuming available bandwidth and starving lower-priority classes.

Spare/shared Bandwidth refers to unused network capacity. Each class can reserve a minimum amount of bandwidth. When a class does not use its reserved bandwidth, TBF makes it available to other classes as spare bandwidth. Any bandwidth not assigned to specific classes also becomes spare bandwidth. Classes access this shared bandwidth based on their priority order. When a class needs its minimum bandwidth, TBF immediately allocates it and removes that amount from the shared pool.

TBF serves higher priority classes before lower priority classes.

Class settings

Input Min.

This setting sets the guaranteed minimum bandwidth for incoming traffic. When this class does not use its minimum allocation, TBF makes it available as spare bandwidth for other classes.

Input Max.

This setting sets the maximum bandwidth limit for incoming traffic. The class does not exceed this limit, even when additional spare bandwidth is available.

Output Min.

This setting sets the guaranteed minimum bandwidth for outgoing traffic. When this class does not use its minimum allocation, TBF makes it available as spare bandwidth for other classes.

Output Max.

This setting sets the maximum bandwidth limit for outgoing traffic. The class does not exceed this limit, even when additional spare bandwidth is available.

QoS assignment

To control traffic priority on enabled quality of service (QoS) interfaces, assign QoS classes to firewall rules.

To control traffic priority through firewall rules, assign QoS classes. TBF supports QoS assignment in both global and local *firewall rules (forward and input)* on page 131. For each rule, specify separate QoS classes for incoming and outgoing traffic. TBF automatically applies these settings to the interfaces where you enable QoS.

Perform the following steps:

1. Navigate to the firewall rules settings section.
2. Select the specific firewall rule where you want to assign QoS classes.
3. Select the appropriate QoS class for **input traffic** and another class for **output traffic**.
4. Save and apply the configuration changes.

TBF now applies the selected QoS classes to the traffic that matches this firewall rule on any interface where you enable QoS.

Tip: Remember to enable QoS on the network interfaces where you want these traffic prioritization rules to take effect.

Limiter

Traffic limiters connect to firewall rules and apply bandwidth restrictions to traffic that matches specific rule criteria.

When you need to set traffic limiters, TBF connects them to firewall rules. Find these limiters in *firewall rules (forward and input)* on page 131, under the **Advanced** settings of each firewall rule. TBF applies the limiter settings to traffic that matches the rule criteria.

Important: Firewall rule limiters work in 1 direction only. To limit both upload and download speeds, you need 2 separate firewall rules. Shape download traffic on the **WAN** interface and upload traffic on the **LAN** interface, where the traffic first enters the firewall.

Global firewall rules (forward and input) on page 139 with parameters that match work better for flexible limiter configurations. With match rules, TBF only performs traffic shaping. You still need a separate allow rule to permit the traffic.

SNMP

The Simple Network Management Protocol (SNMP) allows you to collect and organize information about the system from other network devices.

Find the SNMP settings under **Services > SNMP**.

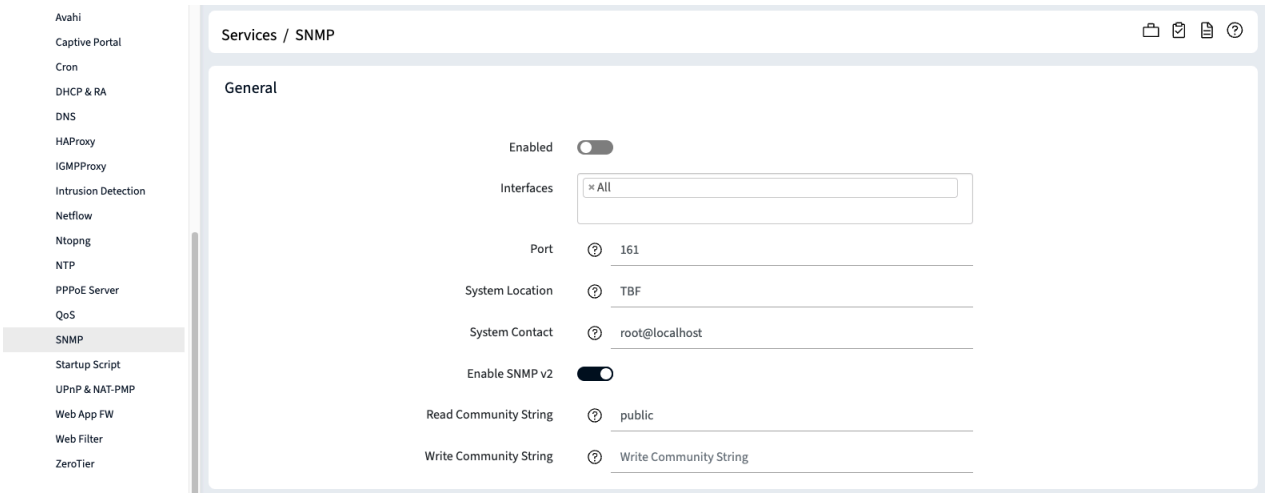


Figure 216: SNMP

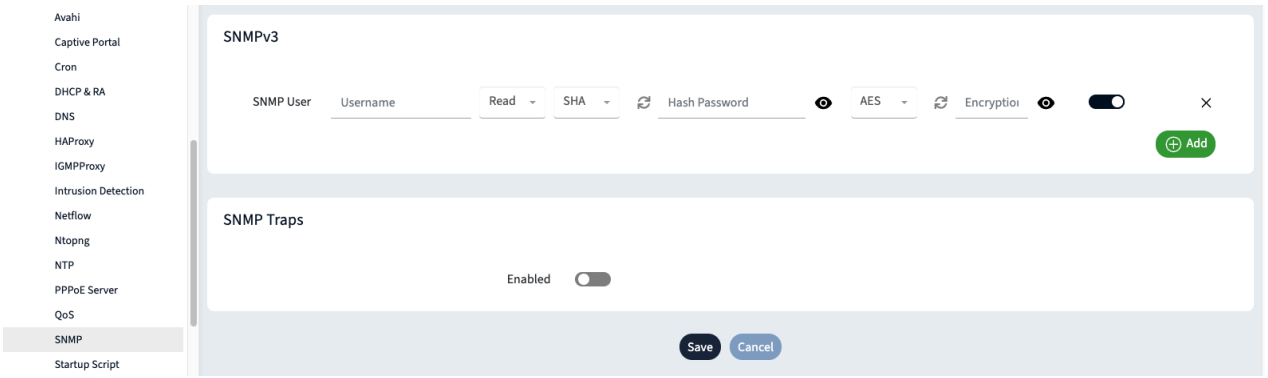


Figure 217: SNMP user

SNMP enables you to gather and organize system information from TBF using other network devices. This protocol offers a standardized way to monitor and manage network equipment remotely.

Enable or disable the entire SNMP service and specify which network interface it operates on. The service uses port `161` by default. The system location and system contact fields include placeholder text that you can customize to match the environment. The `read` community string defaults to `public`. To help prevent read access, leave this field empty.

TBF disables the `write` community string by default for security reasons. When you enable it, you can only write to TBF object identifiers (OIDs). TBF does not support writing to general system OIDs.

For SNMPv3, to create multiple users, set the following settings:

- Username
- Access rights
- Password
- Hash algorithm
    - MD5
    - SHA
    - SHA-224
    - SHA-256
    - SHA-384
    - SHA-512
- Encryption algorithm
    - AES
    - DES

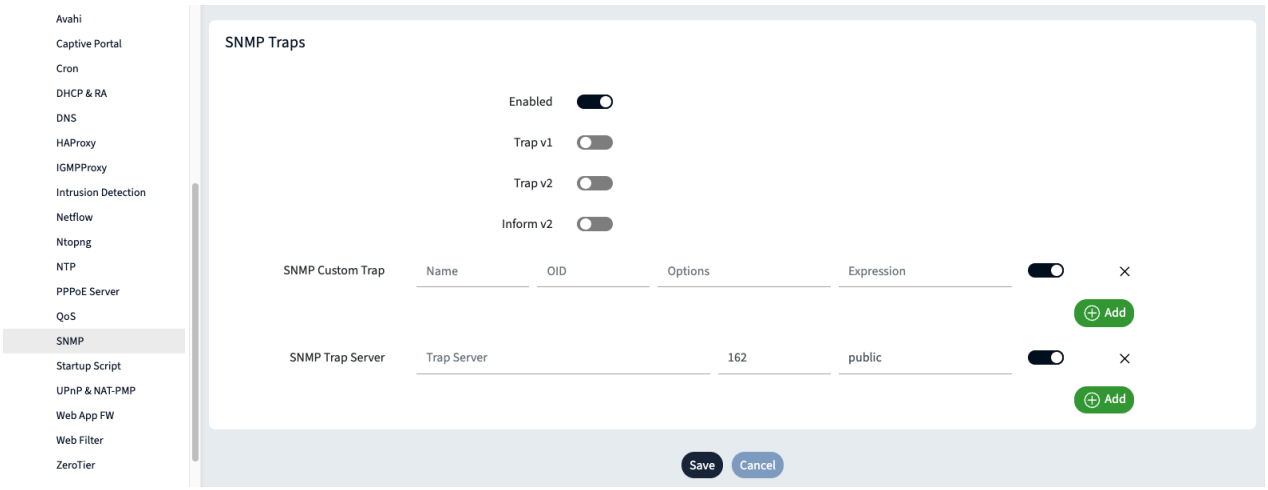


Figure 218: SNMP traps

To receive automatic notifications when specific events occur, enable SNMP traps. Specify whether the trap supports version 1, version 2, or both, and specify whether to send information requests along with the traps.

To create multiple custom traps, define a name, OID, options, and expression for each trap. To set up multiple trap servers, specify an IP address, port number, and community string for each destination server.

Related topics

For detailed MIB information, see:

- *SNMP MIB* on page 300

SNMP MIB

The Management Information Base (MIB) provides access to custom system information through specific object identifiers (OIDs).

Custom OIDs

TBF provides custom system information through its MIB using the following OIDs:

- `1.3.6.1.4.1.53277.1.1.1.1`: SHDSL connection information
- `1.3.6.1.4.1.53277.1.2.1.1`: VDSL connection information
- `1.3.6.1.4.1.53277.2.1.1.1`: Software version and status
- `1.3.6.1.4.1.53277.3.1.1`: Network interface details
- `1.3.6.1.4.1.53277.4.1.1.1`: System temperature readings
- `1.3.6.1.4.1.53277.5.1.1.1`: General system information
- `1.3.6.1.4.1.53277.6.1.1.1`: IPsec tunnel status
- `1.3.6.1.4.1.53277.7.1.1.1`: OpenVPN server status
- `1.3.6.1.4.1.53277.8.1.1.1`: OpenVPN client status
- `1.3.6.1.4.1.53277.9.1.1.1`: WireGuard VPN status
- `1.3.6.1.4.1.53277.10.1.1.1`: Cellular connection information

Startup scripts

Startup scripts contain commands that run automatically when the system starts up.

Find the startup script settings under **Services > Startup Script**.

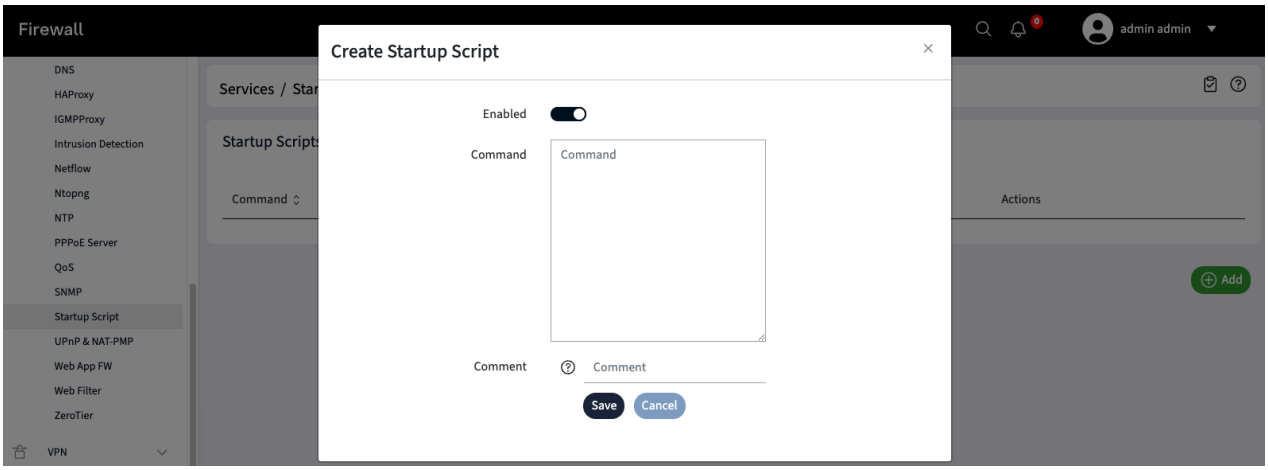


Figure 219: Create startup script

This section displays the commands that TBF runs automatically when it starts up.

To add custom startup entries, specify a custom **Command**.

Note: If the command needs a file, such as when running a script, first place the file in the correct location manually.

Web filter

The web filter uses SquidProxy for the caching proxy functionality for HTTP, HTTPS, and additional protocols.

Find the web filter settings under **Services > Web Filter**.

The web filter is built on SquidProxy, a powerful caching proxy server that handles web traffic for HTTP, HTTPS, and other internet protocols. This proxy server improves network performance when it caches frequently requested content and offers additional security and access control features.

Configuration topics

- *General settings* on page 302
- *HTTP settings* on page 302
- *HTTPS settings* on page 302
- *Cache settings* on page 303
- *ClamAV anti-virus settings* on page 304
- *WPAD autoconfigure settings* on page 305
- *Advanced settings* on page 305

Access control topics

- *Access control lists* on page 306
- *HTTP access rules* on page 307
- *Authentication* on page 308
- *LDAP authentication* on page 309
- *Active Directory authentication* on page 310
- *Blocklist* on page 310

Setup procedures

- *Set up the web filter as proxy* on page 311
- *Set up a transparent proxy* on page 311
- *Set up a proxy with anti-virus* on page 312
- *Set up blocklist filtering* on page 312

General settings

This topic covers basic web filter configuration options including enable/disable and proxy interfaces.

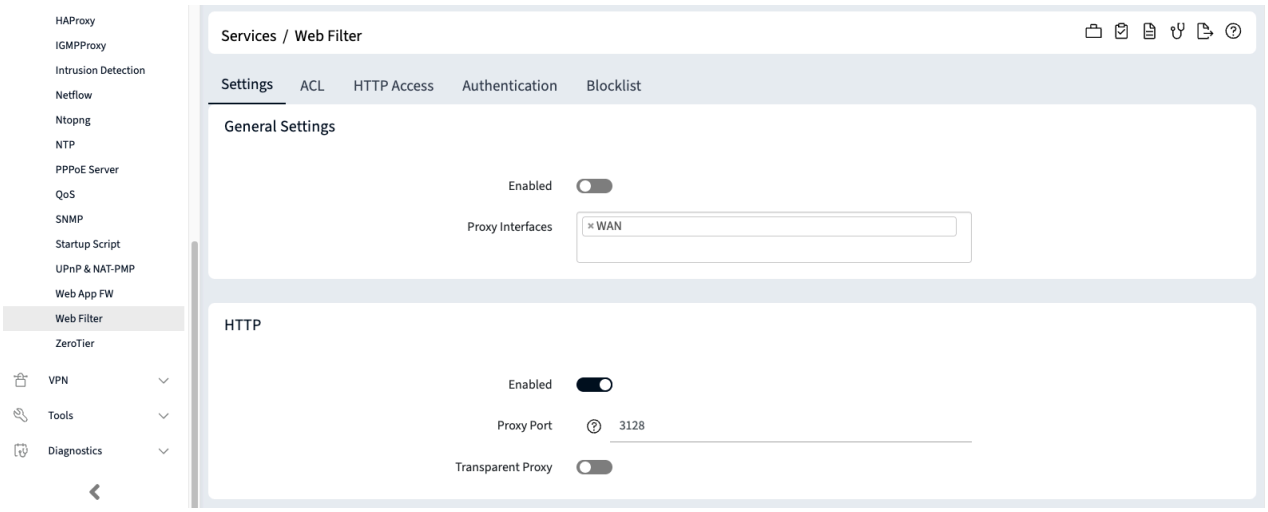


Figure 220: Web filter settings

Settings

Enabled

This setting enables or disables the web filter. The web filter is disabled by default when you first access this page.

Proxy Interfaces

This setting selects the network interfaces and IP addresses where the proxy service operates. The proxy listens for web traffic on these designated interfaces.

Export settings

To export the current settings as an Excel spreadsheet, use the export option in the upper-right corner of the interface.

HTTP settings

Set HTTP proxy configuration options that include port settings and transparent proxy functionality.

Settings

Enabled

This setting enables or disables HTTP functionality. This setting is enabled by default, which allows the proxy to process HTTP traffic.

Proxy Port

This field sets the port number where the HTTP proxy server accepts incoming connections. TBF uses port `3128` as the default listening port.

Transparent Proxy

This setting enables transparent proxy mode, which allows the proxy to intercept traffic and does not require client configuration. This feature is disabled by default. When enabled, the transparent HTTP proxy operates on the local address `127.0.0.1` and uses port `3129`. To use this feature, set Destination Network Address Translation (DNAT) rules to redirect incoming traffic to this specific address and port.

HTTPS settings

This topic covers HTTPS proxy options including SSL/TLS handling and certificate management.

Settings

Enabled

This setting enables or disables the HTTPS functionality. TBF disables this setting by default.

Proxy Port

This field sets the port where the HTTPS proxy server listens for connections. The default is `3129`.

Transparent Proxy

This setting enables the proxy to act as a transparent proxy. TBF disables this setting by default. The transparent HTTPS proxy runs on `fd00:7371:7569:6470:726f:7879::1` and port `3129`. Set up DNAT to redirect traffic to this address. The traffic is encrypted and the transparent proxy of TBF does not perform a man-in-the-middle attack. Therefore, the system gathers necessary information by examining the connection start and extracting IP addresses and the SNI field. This method allows clients to connect without security warnings while helping provide sufficient information to evaluate HTTP access rules.

HTTPS Proxy Mode

This setting specifies the proxy mode when you enable **Transparent Proxy**. Select from:

- **SNI Scan**: This option examines the SNI field during the TLS handshake.
- **Man-in-the-Middle**: This option intercepts the connection and presents the client with its own **Certificate**. The client must trust the **Certificate**, or the browser displays a security warning.

Certificate Authority

This setting lets you select which certificate authority to use for SSL/TLS operations.

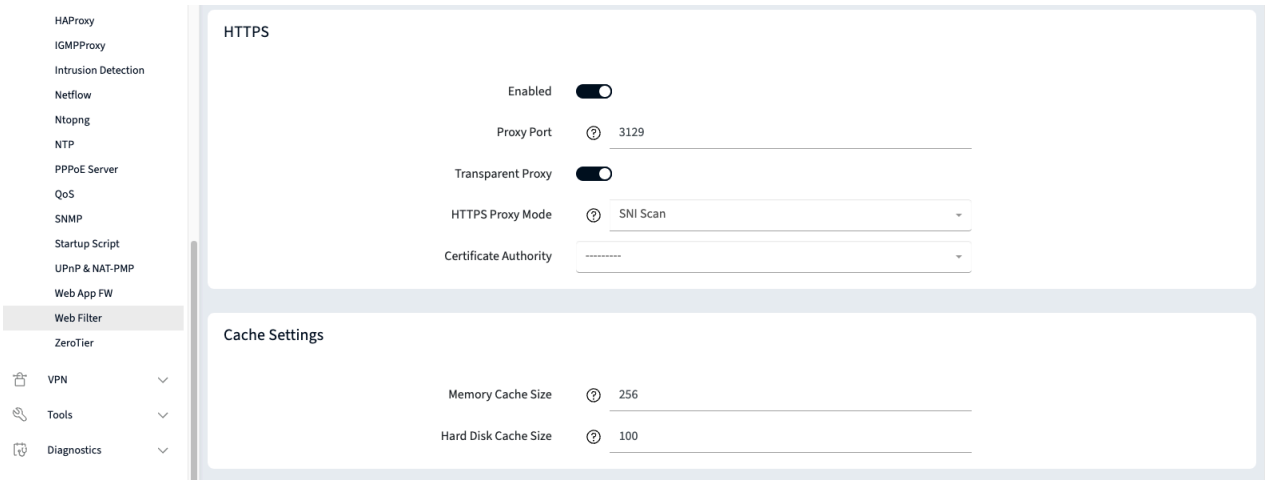


Figure 221: HTTPS and cache settings

Cache settings

This topic covers web filter caching settings.

Settings

Memory Cache Size

This field sets the target amount of system memory allocated for caching web filter data. This value is measured in megabytes (MB). The default is `256` MB.

Note: Higher values improve performance but use more system resources.

Hard Disk Cache Size

This field sets the maximum disk space reserved for storing cached web filter data.

The default is `100` MB. This value is measured in MB.

Note: Larger cache sizes reduce network requests but consume more storage space.

ClamAV anti-virus settings

Set up the anti-virus scanning functionality for web traffic that uses the ClamAV engine and includes configuration options and limitations.

ClamAV scans web traffic through the proxy to detect viruses. This feature works only when the system accesses traffic in unencrypted form. It cannot scan encrypted HTTPS traffic in transparent proxy scenarios.

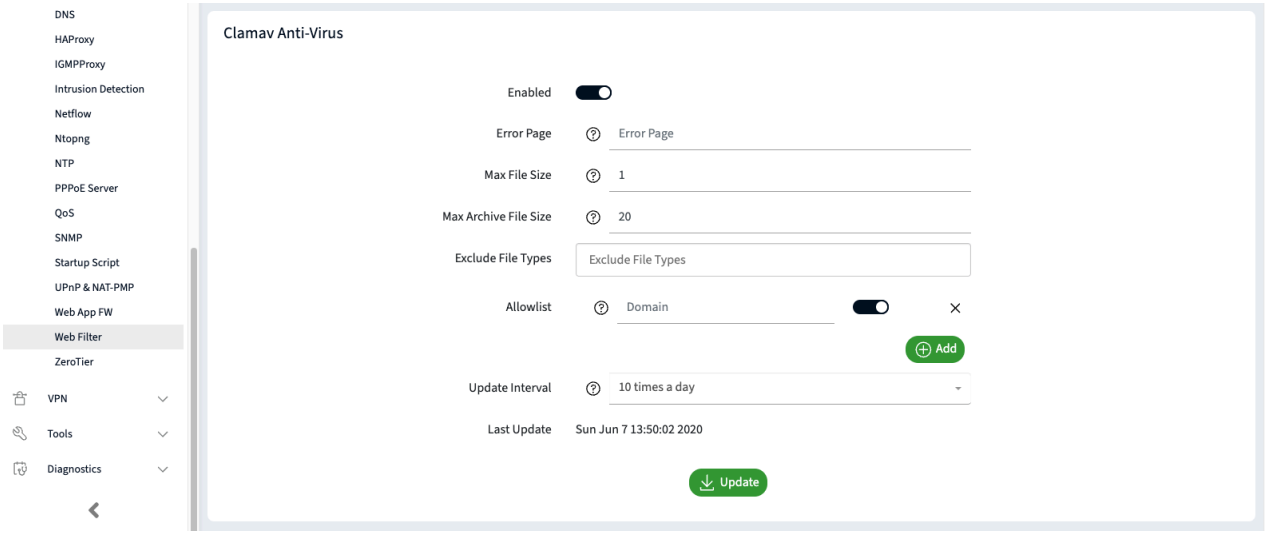


Figure 222: Clamav anti-virus

Settings

Enabled

This setting enables or disables virus scanning.

Error Page

This field lets you enter the complete URL of a custom error page. TBF redirects users to this page when it detects a virus.

Max File Size

This field sets the maximum file size to scan in MB. TBF does not scan files larger than this limit.

Max Archive File Size

This field sets the maximum archive file size to scan in MB. TBF does not scan archive files larger than this limit.

Exclude File Types

This field specifies file types to skip during virus scanning. To improve performance and exclude low-threat file types, use this field.

Allowlist

To bypass virus scanning, add trusted **Domains**. TBF does not scan traffic from these domains, which improves performance for known trusted sources.

Update Interval

This setting sets how many times per day TBF updates virus definitions. More frequent updates help provide better protection against new threats.

WPAD autoconfigure settings

The Web Proxy Auto-Discovery (WPAD) functionality allows clients to automatically discover and set proxy settings.

WPAD enables clients to enable the automatic proxy setting and automatically discover the TBF web filter. This functionality helps eliminate the need for manual proxy configuration on each client device.

Settings

WPAD Enabled

This setting enables or disables the WPAD functionality.

Proxy

To match the network security requirements, specify the HTTP or HTTPS proxy port for client connections.

WPAD Interface

This setting specifies the interface address that communicates with clients. Use only 1 IP address. Verify that the firewall allows connections on this interface.

How WPAD works

TBF creates a DNS record named *WPAD* that helps clients locate the proxy configuration settings automatically.

The web server delivers the WPAD configuration file to clients over port 80 with the standard HTTP protocol.

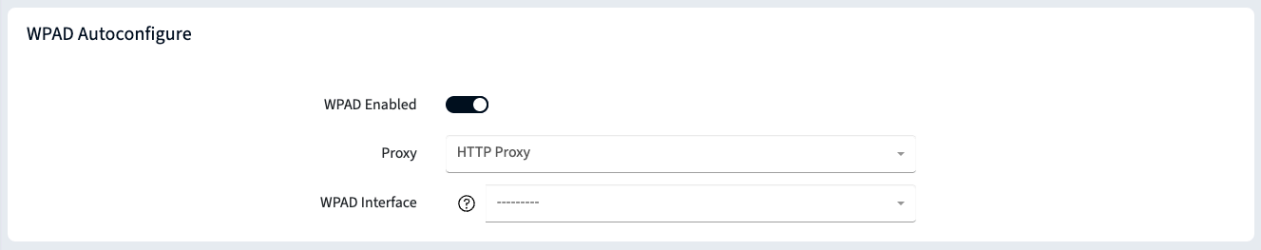


Figure 223: Wpad autoconfigure

Advanced settings

Set advanced web filter settings including upstream proxy, DNS, and custom parameters.

Settings

Upstream Proxy URL

This field lets you enter the complete URL for the upstream proxy server. To include authentication credentials, use the format `http://user:password@proxy.server:port/`.

Visible Hostname

This field sets the hostname that users see in proxy server error messages. When you leave this blank, TBF uses `localhost` as the default value.

Administrator's Email

This field lets you provide an email address that appears in error messages displayed to users. This helps users contact support when issues occur. When you do not specify an address, TBF uses `admin@localhost`.

DNS Server 1 and DNS Server 2

This field sets the primary and secondary DNS servers for domain name resolution. These servers handle DNS queries for the web filter system.

Number of CPUs

This field controls how many CPU cores the web filter uses for processing. The default setting uses the available CPUs minus 1, which reserves 1 CPU core for other system operations. Increasing CPU allocation improves filtering performance for high-traffic environments.

Pre ACL Custom Options

This field adds custom parameters that TBF applies before it processes access control list (ACL) rules. These settings take effect before the main filtering rules run.

Post ACL Custom Options

This field adds custom parameters that TBF applies after it processes ACL rules. These settings take effect after the main filtering rules complete.

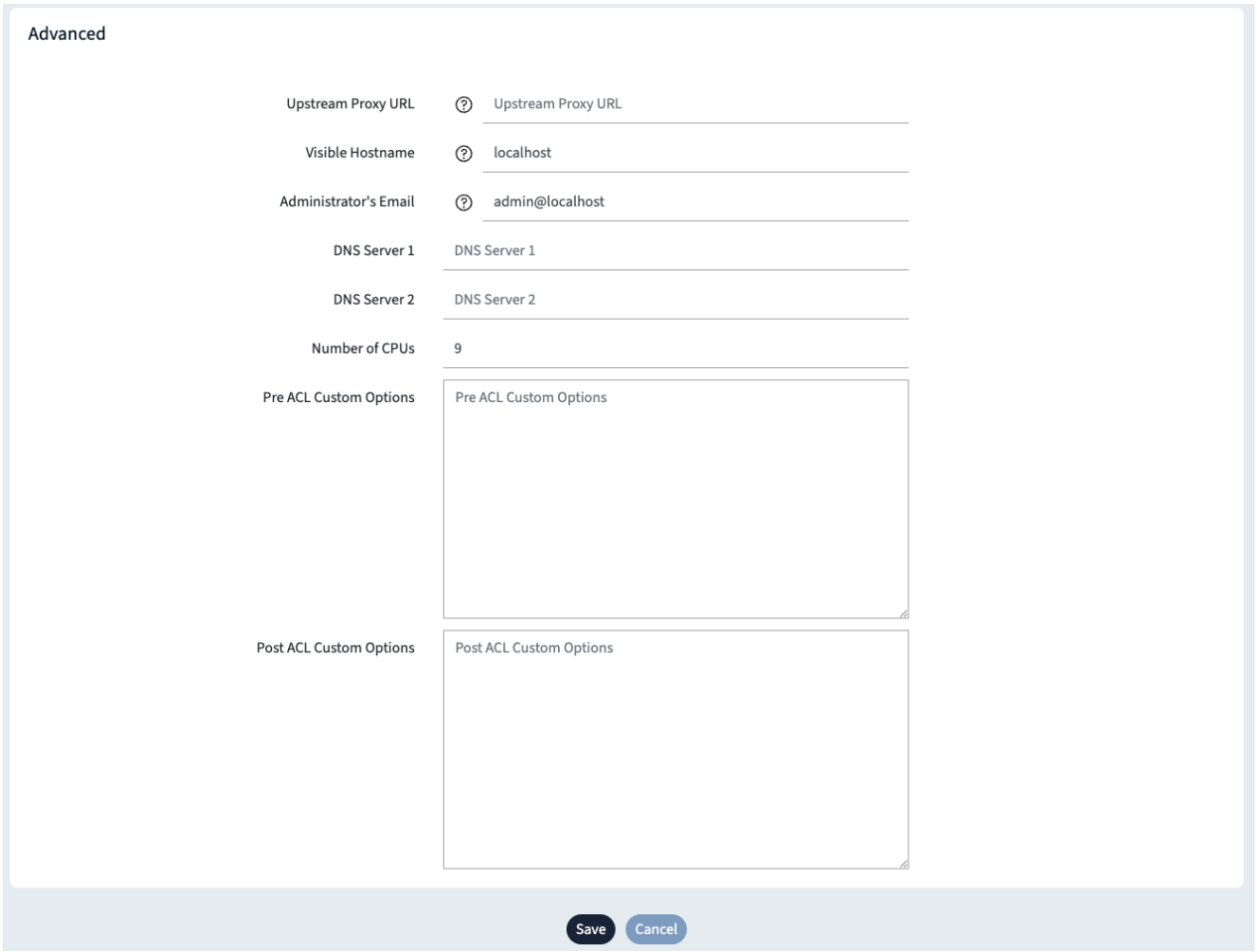


Figure 224: Advanced

Access control lists

Access control lists (ACL) define rules for controlling web traffic based on criteria such as source, destination, and protocol.

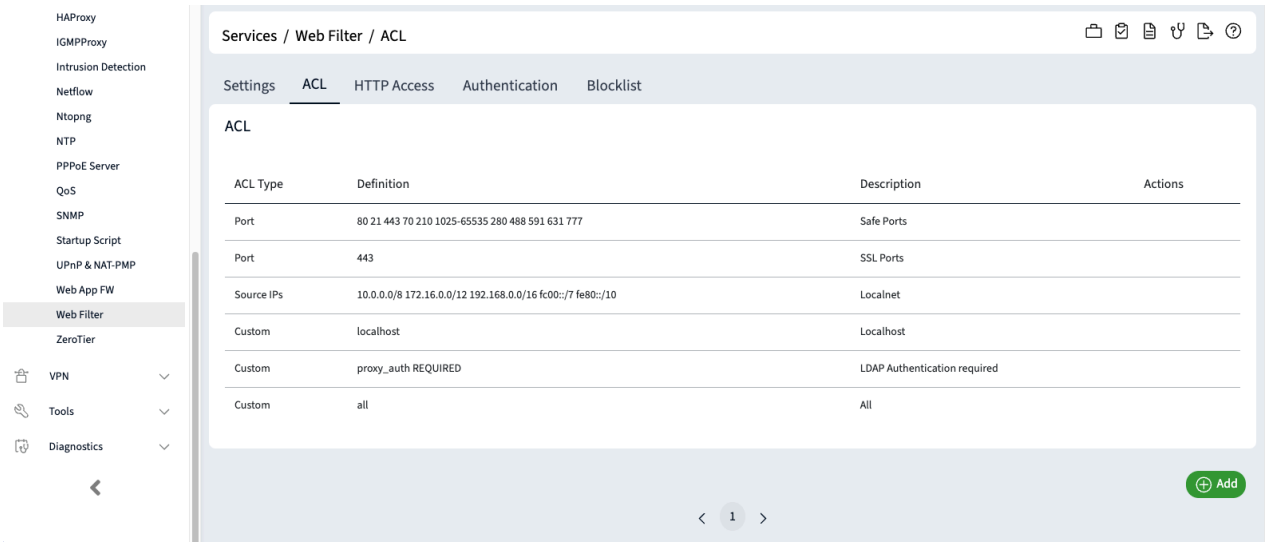


Figure 225: ACL

ACL types

An ACL is one of several types:

- Source
- Destination domain
- Destination regex
- Port
- Protocol
- Custom

The **Custom** type allows you to select an ACL type from the Squid manual (*Squid ACL*).

ACL entries

An ACL entry contains one or more entries. Enter one entry per line. For example, a source ACL contains:

- `192.168.100.1`
- `192.168.101.0/24`

Settings

Log Full Traffic

This option logs the complete data packet that matches this ACL. For HTTPS traffic, TBF bumps the traffic first so it reads the unencrypted content.

SSL Bump

This feature performs SSL bumping and examines SSL traffic. Select from:

- **Splice**: This option reads the SNI field and the certificate but does not decrypt the traffic.
- **Bump**: This option creates a man-in-the-middle connection with the Squid certificate and encrypts and decrypts traffic. This option allows you to either read or encrypt matching traffic.

Important: Run Squid in HTTPS mode and select a **Certificate**, or this option has no effect.

HTTP access rules

HTTP access rules control web traffic and allow or deny access based on access control lists (ACLs) that use logical operators.

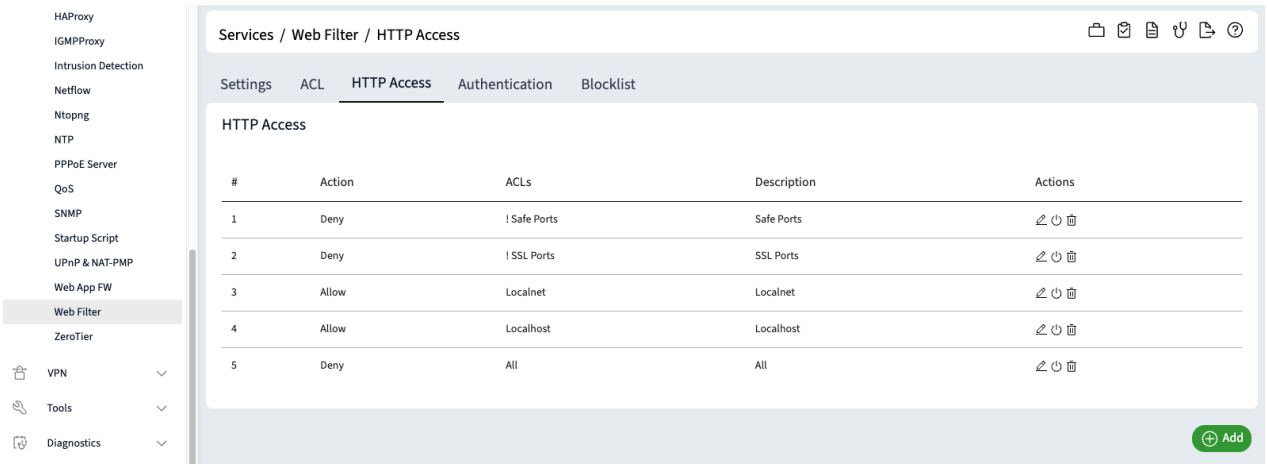


Figure 226: HTTP access

Access control functionality

You can allow or deny access based on access control lists. HTTP access lists combine ACLs with either `AND` or `OR` logic. To negate an ACL, use `NOT`.

This functionality allows you to grant or deny access based on ACLs.

Example configuration

For example, to deny access to `www.google.com`, create an ACL of the type **destination domain**. Then use that ACL in an HTTP access rule of the type **deny**.

Rule ordering

Note: The order of HTTP access rules matters. You can arrange entries in the list in the desired order by dragging and dropping them.

Authentication

Set up web filter authentication methods that include the Lightweight Directory Access Protocol (LDAP) and Active Directory integration for user access control.

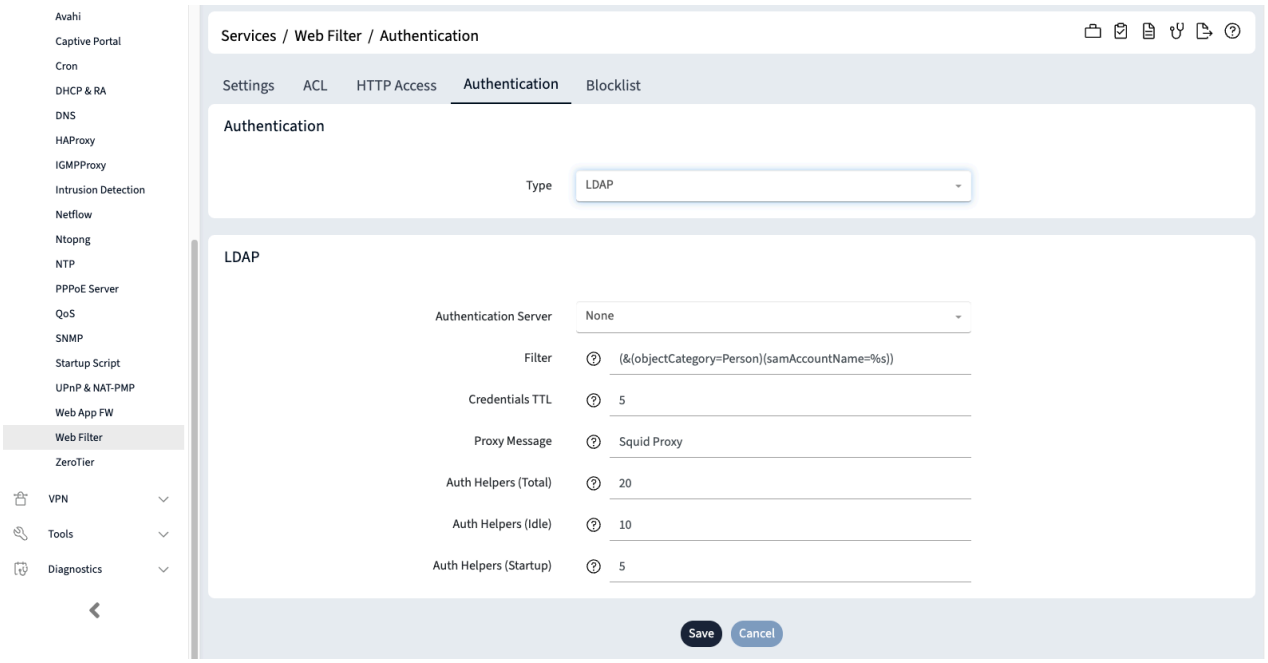


Figure 227: Authentication

To control user access to web resources, set up an additional authentication method. TBF supports 2 authentication methods:

- **LDAP**
- **Active Directory**

Both methods allow you to integrate with existing directory services to authenticate users before they access filtered web content.

LDAP authentication

Set LDAP authentication settings for web filter user authentication including server settings and helper processes.

Settings

Authentication Server

This setting lets you select from the authentication servers you created in TBF.

Filter

This field lets you enter the LDAP search filter used to locate the user DN. This filter is required when users are in a hierarchy below the base DN, or when the login name differs from what builds the user-specific part of the user DN. The default is `(&(objectCategory=Person)(samAccountName=%s))`.

Credentials TTL

This field sets the time in minutes after which the system rechecks credentials. The default is `5`.

Proxy Message

This field sets the proxy authentication informational message displayed to proxy users. The default is `Squid Proxy HASync Test`.

Auth Helpers (Total)

This field sets the total number of authentication helpers to run. Set this *equal* to the number of network proxy users. The default is `20`.

Auth Helpers (Idle)

This field sets the number of idle authentication helpers to run. Set this to *half* the total number of network users. The default is `10`.

Auth Helpers (Startup)

This field sets the number of authentication helpers to run at startup. Set this to *1 quarter* of the total network users. The default is `5`.

Enforce authentication

To enforce authentication, create an HTTP access rule that includes the ACL `proxy_auth REQUIRED`.

Active Directory authentication

The Active Directory authentication options for web filter user authentication include domain settings and process management.

Settings

Domain name short

This field specifies the short form of the domain name.

Domain name long

This field specifies the fully qualified domain name.

Authentication Processes

This field defines the number of authentication processes. Set the number of authentication processes to *25* % of the total number of users in the network. The default is `10`.

Blocklist

Set up the blocklist functionality that uses UT1 categories to block access to unwanted websites and content types.

To block unwanted websites and content, use the UT1 blocklist and select from several categories.

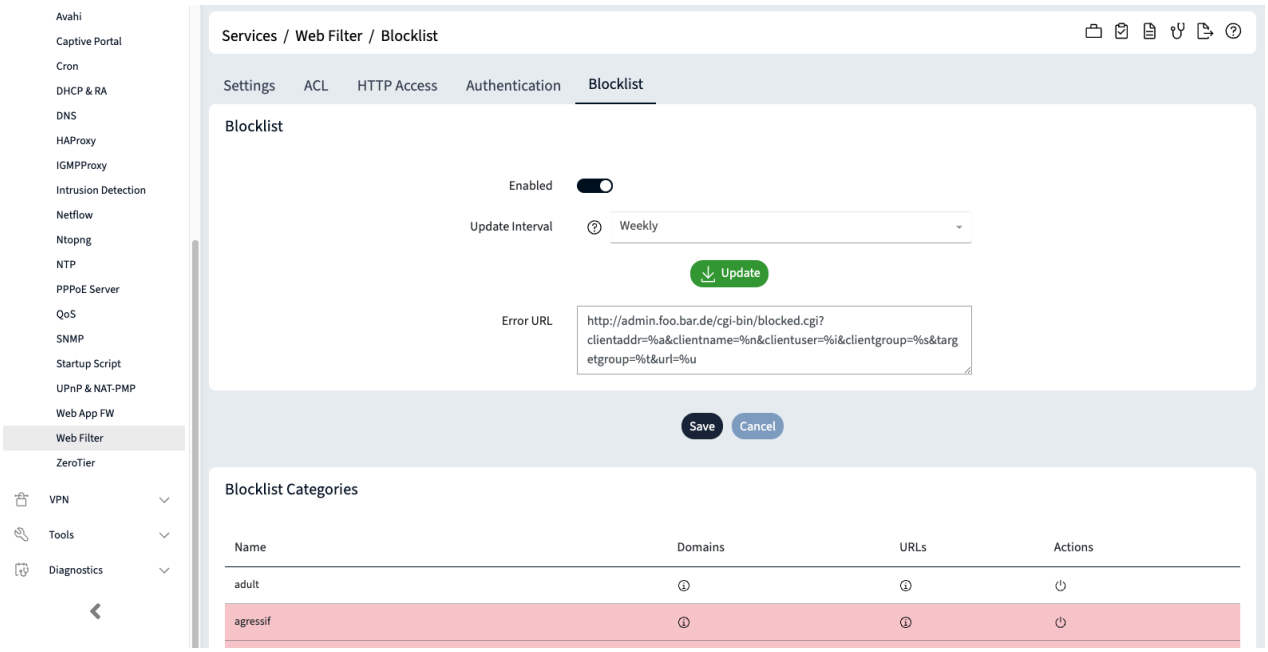


Figure 228: Blocklist

Settings

Enabled

This setting enables or disables the blocklist functionality.

Update Interval

This setting sets how often the blocklist definitions update.

Error URL

This field defines a custom error URL. The default is

```
http://admin.foo.bar.de/cgi-bin/blocked.cgi?clientaddr=%a&clientname=%n&clientuser=%i&client
group=%s&targetgroup=%t&url=%u
```

Define an error URL, or no blocking occurs.

Blocklist Categories

This setting enables or disables these categories as needed. By default, TBF enables the following categories:

- adult
- cryptojacking
- ddos
- malware

To view the **Domains** and **URLs** of each category, select the **Information** button.

Whitelist configuration



Figure 229: Whitelist

Add multiple whitelist entries with their IP address and netmask.

Set up the web filter as proxy

To analyze traffic and cache web content, you can set up the web filter as a proxy in the network.

You can set up the web filter as a proxy in the network. This setup allows TBF to analyze traffic from inside the network before it reaches the internet, and cache web content that enters the network. This setup helps block access to specific unwanted services. Blocking malicious services is a task for *intrusion detection* on page 284. The proxy also reduces bandwidth usage when multiple clients in the local network browse the internet.

Perform the following steps:

1. Navigate to **Services > Web Filter**.
2. Enable the package.
3. Select the local network interface.
   You can select the local network, for example, **LAN**, as the **Interface**.

Set up a transparent proxy

Set up the web filter as a transparent proxy that caches web traffic for the local network.

Set up the web filter as a transparent proxy that caches web traffic for the local network.

Perform the following steps:

1. Enable **HTTP** and the **Transparent Proxy** setting.
2. Enable **HTTPS** and the **Transparent Proxy** setting.

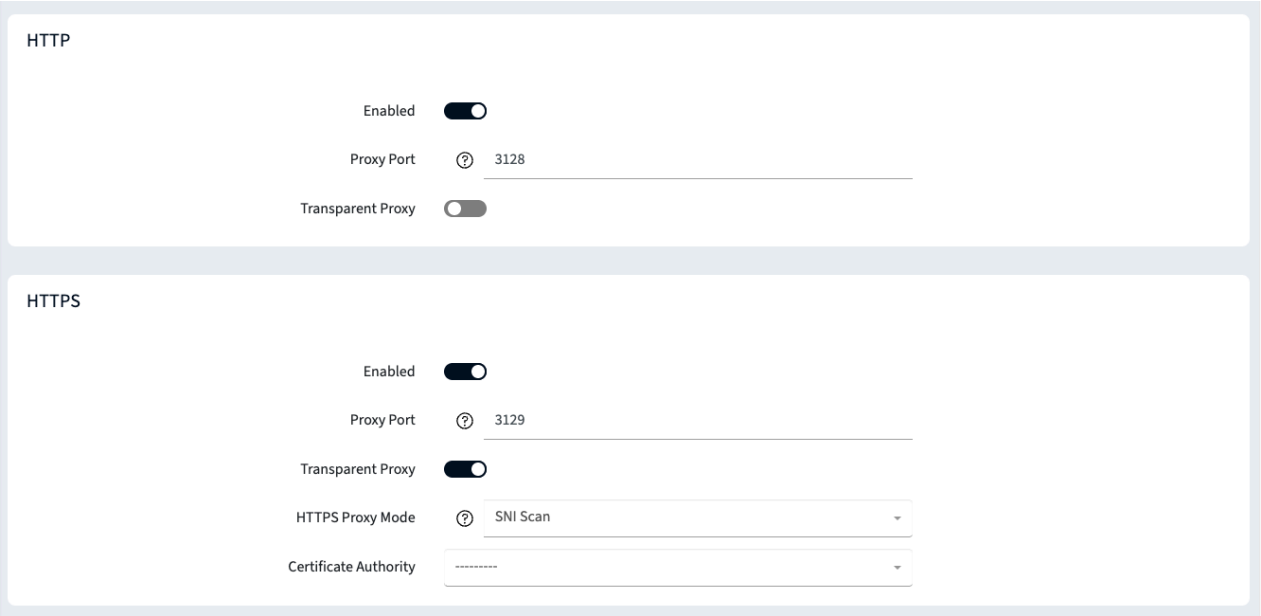


Figure 230: Http(s) configuration

Note: The **HTTPS** transparent proxy cannot inspect encrypted traffic content. Only the certificate and hostname are visible for filtering.

Set up a proxy with anti-virus

Enable ClamAV anti-virus functionality for the web filter proxy to scan unencrypted traffic.

Enable ClamAV Anti-Virus to scan web traffic for viruses. The anti-virus functionality can only inspect unencrypted traffic.

Perform the following steps:

1. Enable **ClamAV Anti-Virus**.
2. Specify an update interval for virus definitions.

The **HTTPS** transparent proxy does not support virus scanning.

Set up blocklist filtering

To block unwanted web content, you can enable and set up blocklist filtering using predefined categories.

To download and use predefined blocklists for filtering web content, you can enable the blocklist functionality.

Perform the following steps:

1. Enable **Blocklist**.
2. To download the predefined blocklists, specify an update interval.
3. To view available blocklist categories, navigate to **Services > Web Filter > ACL**.
4. To create the filtering rules, navigate to **Services > Web Filter > HTTP Access**.
5. To create or edit a blocklist rule, select **Add** or **Edit**.
6. Set the following options:
    a. Set **Action** to **Deny**.
    b. Set **Logical Operator** to **OR**.
    c. Add the blocklist categories that you want to block under **ACL**.

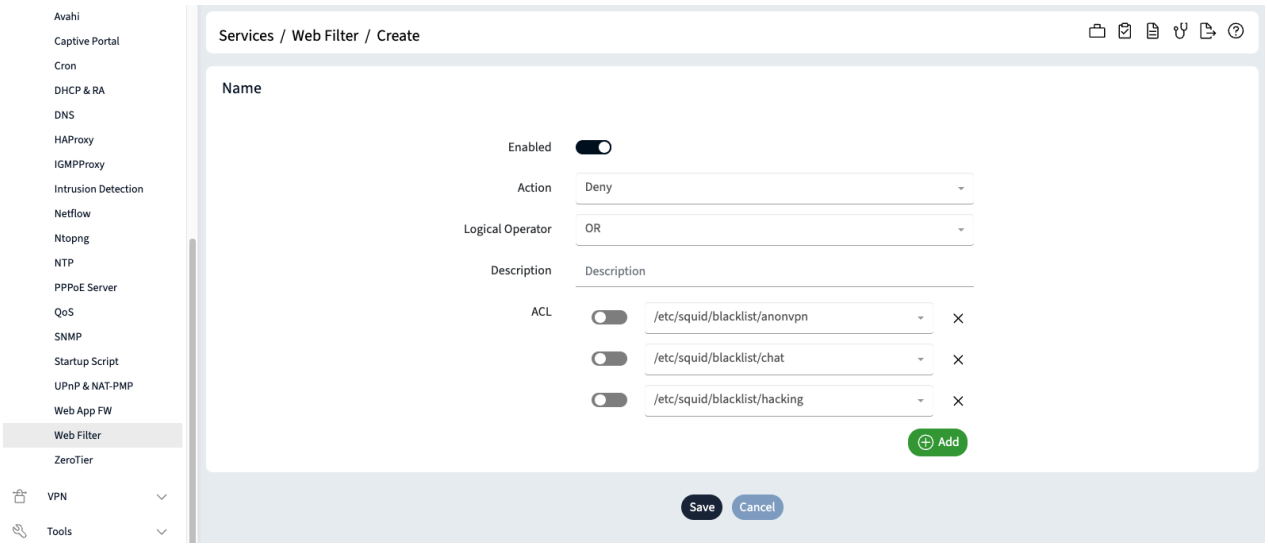


Figure 231: Blocklist configuration

UPnP and NAT-PMP

Universal Plug and Play (UPnP) and NAT Port Mapping Protocol (NAT-PMP) are protocols that enable automatic network configuration and port mapping for network devices.

Access

Find the UPnP and NAT-PMP settings under **Services > UPnP & NAT-PMP**.

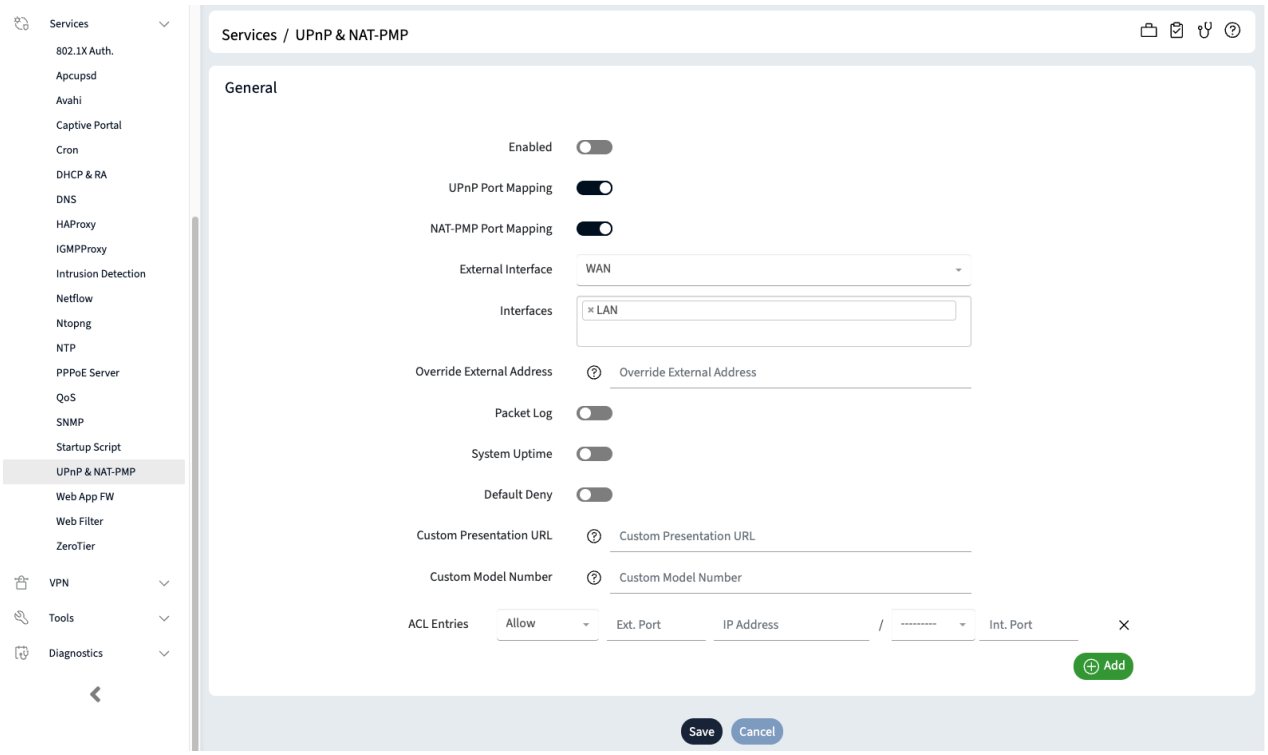


Figure 232: UPnP and NAT-PMP settings

UPnP allows network devices to find each other automatically and to establish connections without manual setup. The NAT-PMP handles the automatic setup of address translation and port forwarding rules.

Both protocols automatically create the necessary firewall and NAT table entries for network communication.

Note: Without proper access control list (ACL) entries in place, any device on the network can automatically create NAT and firewall rules. This automatic rule creation can introduce security vulnerabilities to the network.

UPnP and NAT-PMP settings

Set the settings for UPnP and NAT-PMP services including interface, access control, and logging options.

Settings

Enabled

This setting enables or disables the service. TBF disables it by default.

UPnP Port Mapping

This setting enables or disables the UPnP port mapping functionality.

NAT-PMP Port Mapping

This setting enables or disables the NAT-PMP port mapping functionality.

External Interface

This setting specifies which network interface connects to the external network, typically the **WAN** interface.

Interfaces

This setting sets up a set of network interfaces where the UPnP and NAT-PMP services are available.

Override External Address

This field sets up an alternative external address for inbound connections. This is useful when you use IP aliases or CARP virtual IP addresses.

Packet Log

This setting enables or disables packet logging.

System Uptime

This setting reports the total system uptime instead of the UPnP daemon uptime when devices query the service.

Default Deny

This setting denies access by default.

Custom Presentation URL

This field defines a custom web interface URL that UPnP devices can access. When you leave this field empty, TBF uses the default TBF URL.

Custom Model Number

This field sets an optional custom model number. When you leave this field empty, TBF uses the current TBF firmware version.

ACL Entries

This setting sets access control rules to manage which devices and ports can use UPnP and NAT-PMP services. Each entry specifies whether to allow or deny access, defines **External and Internal Port** ranges, specifies an **IP Address**, and optionally includes a **Netmask** for network ranges.

Web application firewall

The web application firewall (WAF) helps protect web applications by filtering, monitoring, and blocking HTTP traffic between web services and users. Before use, install the WAF through **System > Addons**. The WAF requires HAProxy configuration to analyze unencrypted traffic properly.

Find the web application firewall settings under **Services > Web Application FW**.

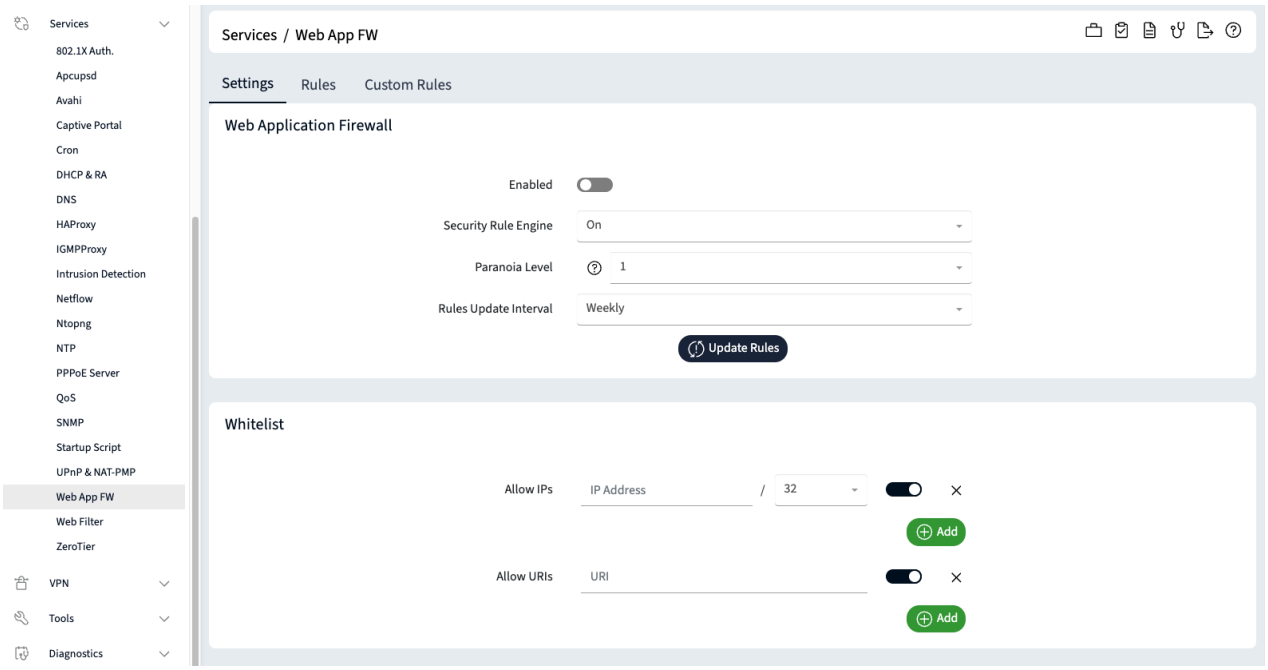


Figure 233: Web application firewall

A WAF is a security tool that helps protect web applications by filtering, monitoring, and blocking HTTP traffic between web services and users. The WAF acts as a barrier that examines the web traffic before the traffic reaches the application.

Before you use the WAF, install the WAF. To install the WAF, go to **System > Addons**.

To export the current WAF settings as an Excel spreadsheet, use the export option in the upper-right corner of the interface.

The WAF requires HAProxy configuration to work properly because the WAF needs to analyze unencrypted traffic. If the web traffic uses SSL encryption, then set up SSL offloading to decrypt the traffic before the WAF inspects the traffic.

OWASP CRS

The WAF uses the OWASP Core Rule Set (CRS) to calculate threat scores and take actions based on thresholds.

The WAF uses the OWASP CRS to help protect applications from security threats.

The OWASP CRS works as a scoring system that evaluates incoming requests and takes actions when necessary. Here is how it works:

- When an incoming request matches a security rule, the system adds points to a threat score based on the severity level of the rule.
- Each rule has a specific point value that reflects how severe the detected threat is.
- The system compares the final threat score against the set threshold.
- When the score exceeds the threshold, the WAF automatically executes the action defined in the final rule.

This approach allows the WAF to distinguish between minor security concerns and serious threats. This helps provide appropriate responses to different severity levels.

WAF settings

The options for the WAF include the security rule engine, paranoia level, and rules update interval.

To control how TBF helps protect against web-based attacks, enable or disable the WAF functionality.

Security Rule Engine

This setting controls the operational mode of the WAF. Select from:

- **On**: This option actively blocks threats.
- **Off**: This option disables the feature.
- **DetectionOnly**: This option monitors and logs threats without blocking the threats.

Paranoia Level

This setting determines the strictness of security rules. The values range from 1..4. Level **1** helps provide basic protection with minimal false positives. The default is **1**. Level **4** assists in establishing a high level of security but blocks legitimate traffic more frequently. To enhance security, select a higher level. To reduce the chance of blocking valid requests, select a lower level.

Rules Update Interval

This setting sets up an automatic schedule to update security rules. TBF uses a cron job to download and apply the latest rule definitions at specified intervals. This helps assure that the WAF stays current with emerging threats.

Set up the WAF allowlist

You can add IP addresses and URIs to the WAF allowlist to exclude them from inspection.

You can add multiple **Allow IPs** entries with their corresponding IP addresses and ports. The WAF does not inspect traffic from these trusted IP addresses. This configuration allows the traffic to bypass security checks.

You can add multiple **Allow URIs** entries with their corresponding URIs. The WAF does not inspect requests to these specific URIs, which is useful to exclude legitimate resources or endpoints from security scanning.

WAF plugins

The WAF plugins provide specialized rules and exceptions for specific applications.

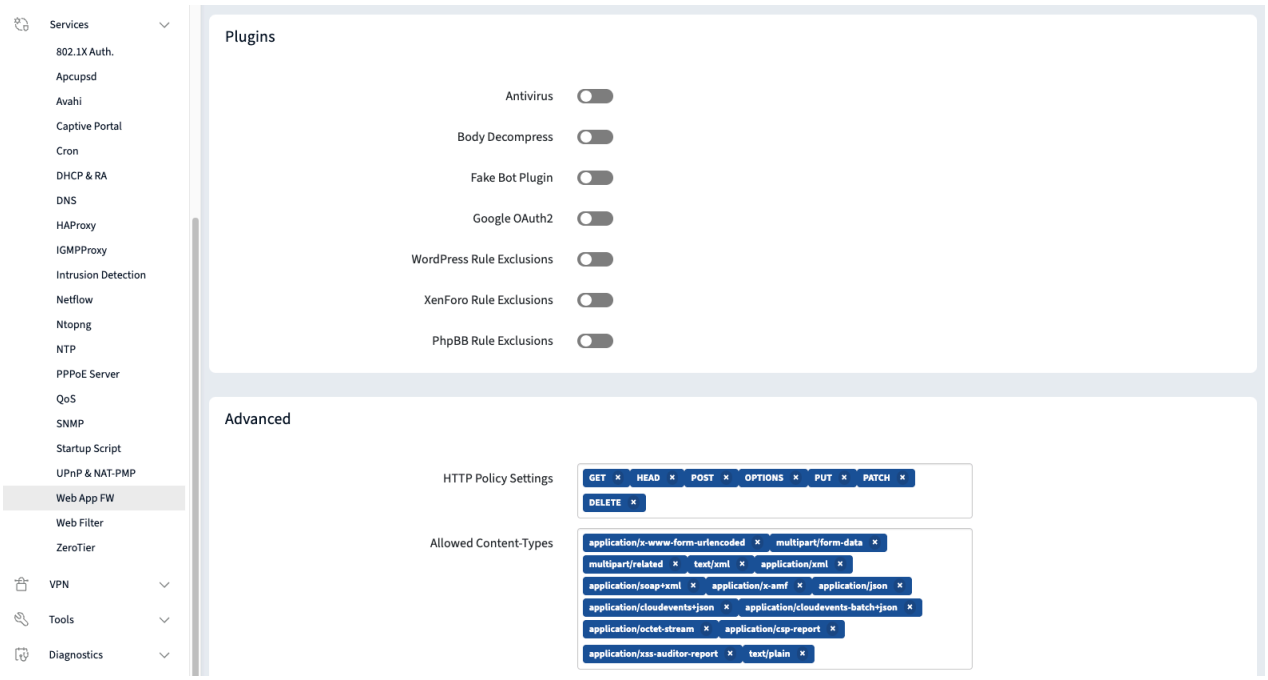


Figure 234: WAF plugins

To help protect the web application, enable the following WAF plugins:

- **Antivirus**: This plugin scans uploaded files for malicious content.
- **Body Decompress**: This plugin handles compressed request bodies for inspection.
- **Fake Bot Plugin**: This plugin identifies and blocks malicious bots that pretend to be legitimate crawlers.
- **Google OAuth2**: This plugin provides authentication integration with Google services.
- **WordPress Rule Exclusions**: This plugin optimizes WAF rules specifically for WordPress sites.
- **XenForo Rule Exclusions**: This plugin customizes rules to help protect XenForo forum software.
- **PhpBB Rule Exclusions**: This plugin adapts security rules for PhpBB bulletin board systems.

Note: When you use any of these applications, enable the corresponding plugin to help protect the applications. These plugins contain specialized rules and exceptions for each application type. This approach helps provide better security coverage and minimizes false positive alerts that disrupt normal application functionality.

Advanced WAF settings

This topic describes advanced HTTP policy settings and custom rules for the web application firewall.

HTTP Policy Settings

This setting controls which HTTP methods the web application accepts from client requests. By default, TBF allows these HTTP methods:

- GET
- HEAD
- POST
- OPTIONS
- PUT
- PATCH
- DELETE

Allowed Content-Types

This setting determines which content types clients send in their HTTP requests. TBF accepts these content types by default:

- application/x-www-form-urlencoded
- multipart/form-data
- multipart/related
- text/xml
- application/xml
- application/soap+xml
- application/x-amf
- application/json
- application/cloudevents+json
- application/cloudevents-batch+json
- application/octet-stream
- application/csp-report
- application/xss-auditor-report
- text/plain

Allowed HTTP versions

This setting controls which HTTP protocol versions TBF accepts from client connections. TBF allows the following versions by default:

- HTTP/1.0
- HTTP/1.1
- HTTP/2
- HTTP/2.0

Blocked file extensions

This setting blocks access to files with potentially harmful or sensitive extensions. TBF blocks these file extensions by default:

- asa
- asax
- ascx
- axd
- backup
- bak
- bat
- cdx
- cer
- cfg
- cmd
- com
- config
- conf
- cs
- csproj
- csr
- dat
- db
- dbf
- dll
- dos
- htr
- htw
- ida
- idc
- idq
- inc
- ini
- key
- licx
- lnk
- log
- mdb
- old
- pass
- pdb
- pol
- printer
- pwd
- rdb
- resources
- resx
- sql
- swp
- sys
- vb
- vbs
- vbproj
- vsdisco
- webinfo
- xsd
- xsx

Blocked request headers

This setting blocks specific HTTP request headers that pose security threats. TBF blocks these headers by default:

- proxy
- lock-token
- content-range
- if
- user-agent

File extensions considered static files

This setting identifies which file types TBF treats as static content that does not require server-side processing. TBF classifies the following extensions as static files by default:

- jpg
- jpeg
- png
- gif
- js
- css
- ico
- svg
- webp

Allowed Charsets

This setting specifies which character encodings clients declare in their content type headers. TBF accepts these character sets by default:

- utf-8
- iso-8859-1
- iso-8859-15
- windows-1252

Custom Rules

This field lets you create additional security rules that go beyond the standard HTTP policy settings. To address specific security requirements for the web application, use custom rules.

Manage WAF rules

View, enable, disable, and remove specific WAF rule files to customize the level of security.

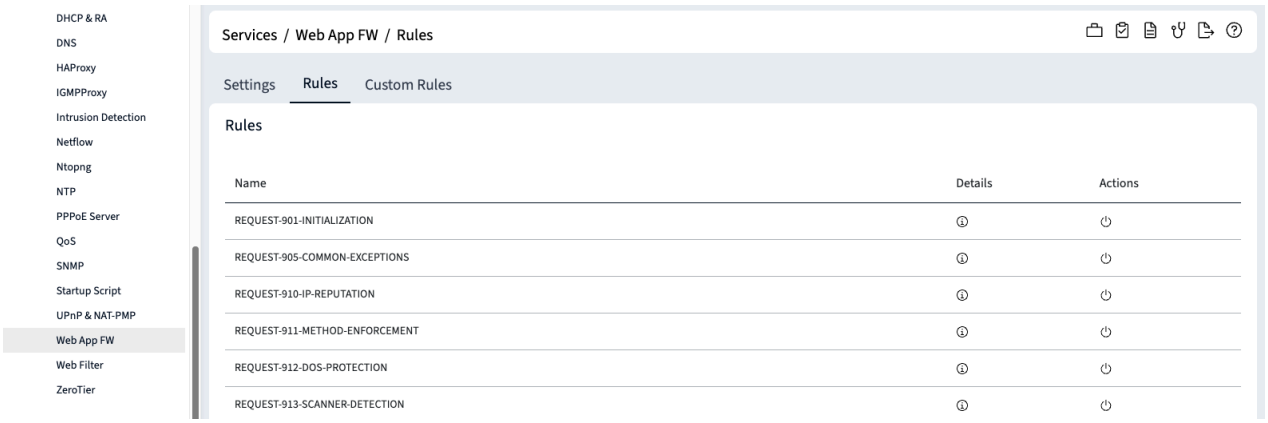


Figure 235: WAF rules

The **Rules** tab displays the installed and active rule files in the system. Each rule file contains security policies that help protect web applications from various threats. To quickly enable or disable individual rule files based on security requirements, use the **Actions** column. To understand what each rule does, select the **Details** icon next to any rule to view its specific definition and purpose.

To disable specific rules without affecting entire rule files, add multiple **Remove Rules** entries. Enter the rule IDs of the rules you want to disable. This approach helps provide granular control over the WAF configuration and maintains overall security coverage.

ZeroTier

ZeroTier creates networks that connect on-premise, cloud, desktop, and mobile devices.

ZeroTier is a network virtualization service that creates encrypted connections between devices across different locations. ZeroTier works by establishing a spoke connection model, where each device connects to a central network controller.

Find the ZeroTier settings under **Services > ZeroTier**.

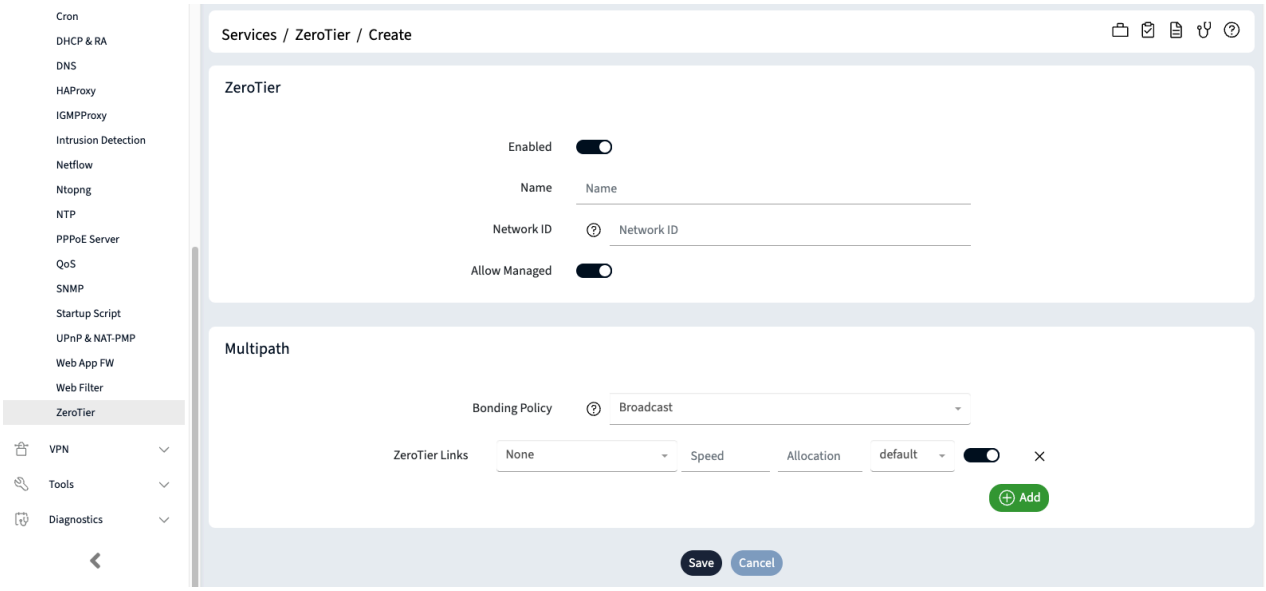


Figure 236: ZeroTier

To get started with ZeroTier, complete these steps:

1. To access the service, sign up for a *ZeroTier* account.
2. To install ZeroTier on the system, navigate to **System > Addons**.
3. Create and manage the ZeroTier networks through the interface.

Once you set up ZeroTier, view the networks you have created and enable or disable them as needed.

ZeroTier configuration

- *ZeroTier settings* on page 320: Configuration parameters for ZeroTier networks
- *ZeroTier multipath* on page 321: Multipath configuration for multiple interfaces

ZeroTier settings

You can set the settings for ZeroTier network connections.

Settings

Name

This field specifies a descriptive name to help you identify this ZeroTier connection. This name appears in the connection list for easy reference.

Network ID

The field specifies the 16-character network identifier for the ZeroTier network. You can find this ID in the ZeroTier Central dashboard or you can obtain it from a network administrator.

Allow Managed

When you enable this setting, ZeroTier automatically assigns and manages IP addresses for this connection.

Note: If you set up a bridge configuration, then you disable this setting. You manually set the IP address on the bridge interface instead.

ZeroTier multipath

Multipath configuration allows you to use multiple network interfaces at the same time in either fallback mode or bonding mode.

Bonding configuration

Bonding Policy Port

Set this setting to one of the following bonding policies:

- **None**
- **Active Backup**
- **Broadcast**
- **Balance RR**
- **Balance XOR**
- **Balance Aware**

ZeroTier links

Add multiple **ZeroTier Links**. Assign each link to a specific **Interface**.

Each link has the following options:

Speed

This field sets the relative speed of this link with arbitrary units. Enter a number between 1..1000000.

Allocation

This field sets how much traffic this link handles relative to other links. Enter a number between 1..255.

Mode

This setting controls when this link is active. Select one of these modes:

- **undefined**
- **primary**
- **spare**

Note: To distribute traffic across multiple links, set only either **Speed** or **Allocation**.

17. VPN

Virtual private network (VPN) technologies help provide connections over public networks.

A VPN creates an encrypted connection between a device and a remote server. This connection helps provide access to resources on a private network as if you were directly connected to it, even when you use public internet infrastructure.

TBF supports multiple VPN technologies, each with different strengths and use cases:

VPN technologies

***IPsec* on page 322**

This is an industry-standard protocol suite for IP communications. It is suitable for site-to-site connections and enterprise deployments.

***OpenVPN* on page 338**

This is an SSL/TLS-based VPN solution. It is suitable for remote access and operates through firewalls and NAT.

***WireGuard* on page 352**

This is a VPN protocol with cryptography and high performance. It is straightforward to set up and maintain.

***WebVPN* on page 358**

This is a browser-based remote access solution that uses Apache Guacamole. No client software is required.

Choose a VPN technology

Select the appropriate VPN technology based on the requirements:

- **Site-to-site connections**: IPsec or WireGuard
- **Remote user access**: OpenVPN or WireGuard
- **Browser-based access**: WebVPN
- **High performance requirements**: WireGuard or IPsec with hardware offload
- **Legacy system compatibility**: IPsec or OpenVPN

IPsec

Internet Protocol Security (IPsec) assists in establishing security for Internet Protocol (IP) communications through authentication and encryption.

IPsec is a suite of protocols that assists in establishing security for IP communications by authenticating and encrypting each IP packet in a communication session. IPsec operates at the network layer and helps protect data flows between hosts, between security gateways, or between hosts and security gateways.

IPsec offers the following security services:

- **Authentication**: Verifies the identity of the sender
- **Integrity**: Helps provide assurance that data has not been modified in transit
- **Confidentiality**: Encrypts data to help prevent unauthorized access
- **Anti-replay protection**: Helps prevent replay attacks

IPsec topics

The following topics contain detailed information about IPsec configuration and usage:

- *IPsec general settings* on page 323: Global IPsec configuration options
- *IPsec phase 1* on page 324: IKE negotiation and authentication
- *IPsec phase 2* on page 330: ESP/AH security associations
- *IPsec client settings* on page 333: Mobile client configuration
- *IPsec route based (VTI/XFRM)* on page 334: Route-based VPN configuration
- *GRE over IPsec* on page 335: GRE tunneling over IPsec
- *IPsec troubleshooting* on page 337: Diagnostic and troubleshooting information

IPsec general settings

Set up global IPsec settings including Cisco extensions, network exclusions, replay protection, and hardware offload options.

Find the IPsec settings under **VPN > IPSec > Global Settings**.

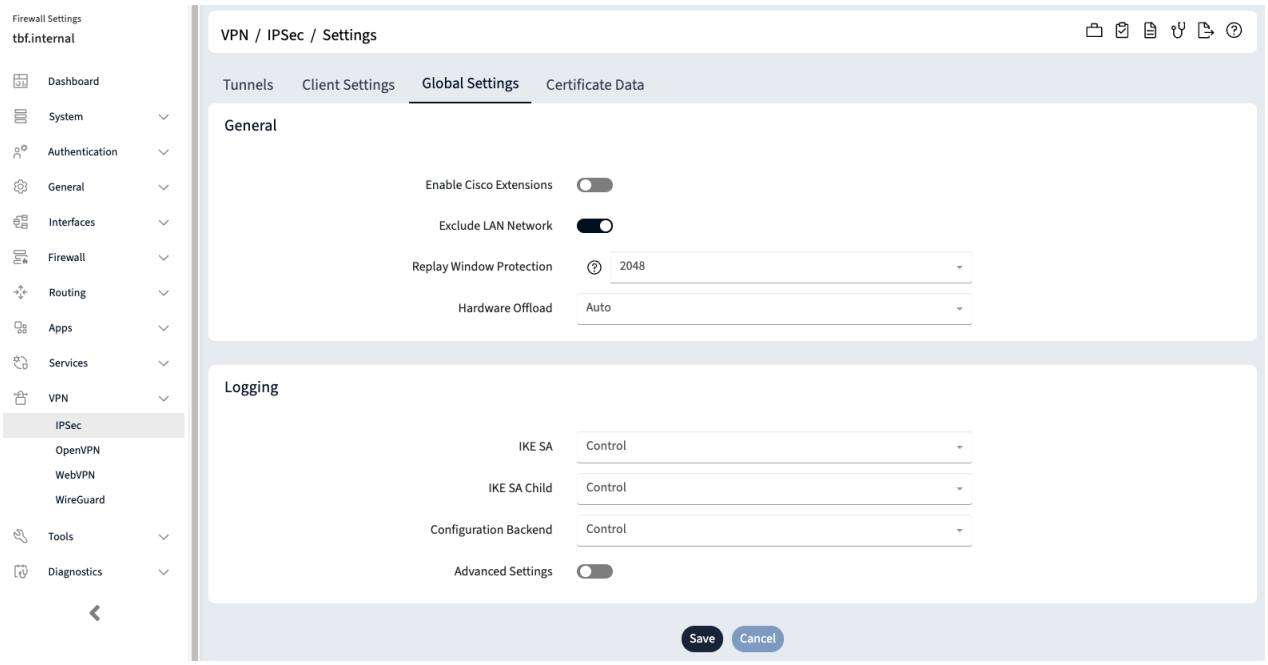


Figure 237: IPSEC global settings

Settings

Enable Cisco Extensions

This setting enables or disables Cisco extensions. If you need to create multiple *IPsec phase 2* on page 330 entries in IKEv1 and the remote device requires this feature, then enable this setting. This extension helps provide tunnel configurations with Cisco equipment.

Exclude LAN Network

This setting enables or disables the exclude LAN network functionality. This setting helps prevent traffic between the local network (LAN subnet) and the LAN IP address from going through IPsec processing. To help avoid routing conflicts and verify that local network traffic stays within the internal network, enable this setting.

Replay Window Protection

This setting determines the size of the AH/ESP replay window, which helps protect against replay attacks by tracking received packets. The window size is measured in packets, with a default of **2048** packets.

Note: Larger window sizes assist with protection but reduce system performance.

Hardware Offload

This setting determines the hardware offload mode. Select between:

- **Auto**: This option lets the system automatically identify compatible Mellanox or Intel network cards that support IPsec offload and uses them to handle encryption tasks at the hardware level to increase VPN performance. This is the default option.
- **No**: This option disables the hardware offload.

IPsec logging

Set logging settings to monitor IPsec operations and troubleshoot configuration issues.

You can define various logging settings to monitor IPsec operations. The 3 primary logging categories are:

- IKE SA
- IKE SA child
- Configuration backend

If you encounter configuration issues with IPsec, then change the logging level to `DEBUG` for these categories to get detailed troubleshooting information.

Under **Advanced Settings**, you can define additional logging options for more specific monitoring requirements.

IPsec phase 1

Internet Protocol Security (IPsec) defines the architecture for security services in IP network traffic.

IPsec defines the architecture for security services in IP network traffic. IPsec helps secure traffic between 2 entities through either full encryption or authentication only.

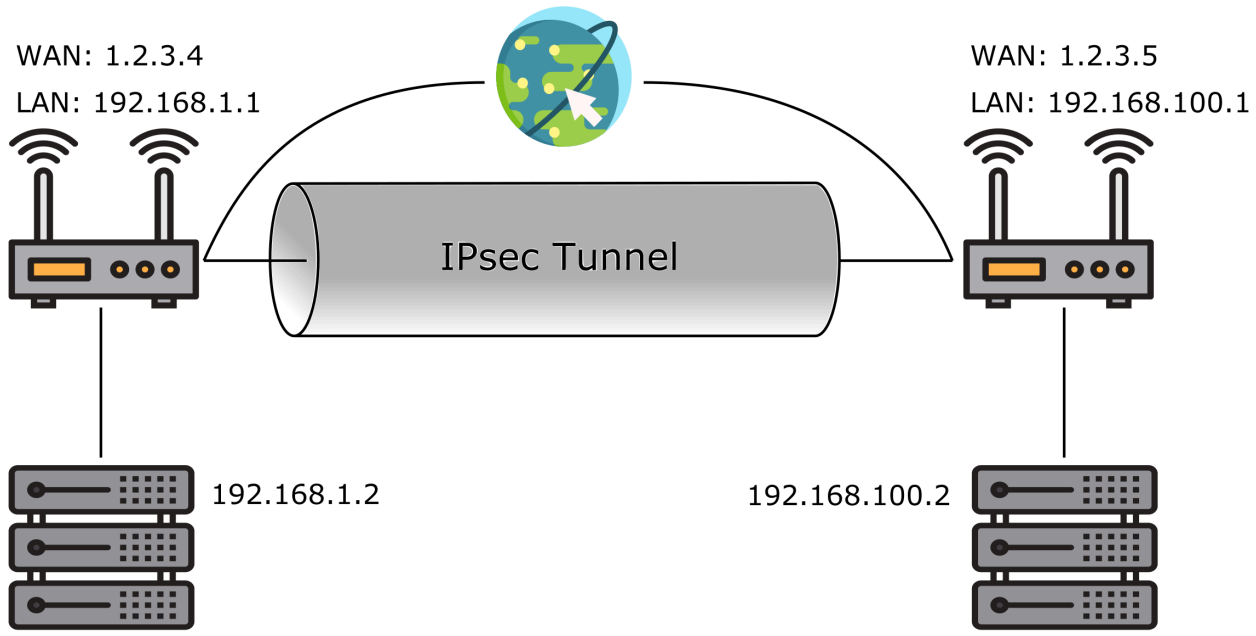


Figure 238: IPSEC tunneling

This image uses icons by *srip* and *Freepik* from *Flaticon*.

Find the IPsec settings under **VPN > IPSec**.

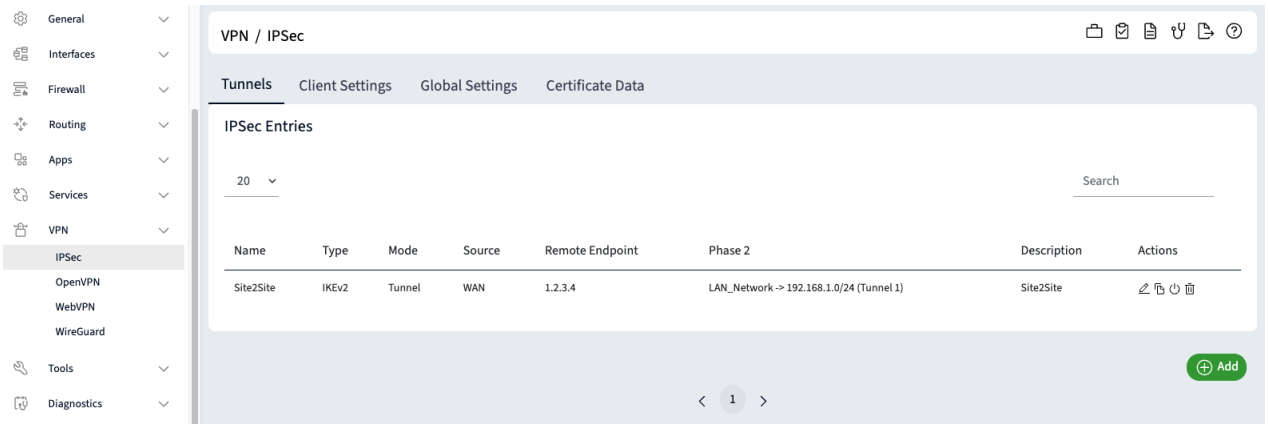


Figure 239: IPSEC tunnel overview

Phase 1 topics

- *Phase 1 configuration* on page 325: Configuration parameters for IPsec phase 1
- *Phase 1 authentication* on page 326: Authentication methods and settings
- *Phase 1 settings* on page 327: Advanced configuration options
- *IPsec action example* on page 329: Best practices and configuration examples
- *IPsec multi-WAN setup* on page 329: Multi-WAN configuration with automatic failover
- *IPsec with failover* on page 330: Failover configuration setup
- *Performance* on page 330: Performance optimization tips
- *IPsec firewall rules* on page 330: Required firewall configuration

Phase 1 configuration

IPsec phase 1 negotiates encryption and setting parameters. TBF has various options to set up a phase 1.

Interfaces are sender interfaces that also set the source IP address of the IPsec tunnel on the TBF end. If the interface is a VRRP virtual IP address in an HA setup, then only the master that holds the IP address activates the tunnel. The slave remains on standby for the IPsec connection. You can select multiple interfaces.

Important: TBF uses the default route when the system initializes the connection. Therefore, TBF uses only the IP address of the interface associated with the default route.

If you select **Any**, then TBF selects the IP address based on the default route and active interfaces. This selection is useful for backup connections when you have multiple gateways, as the system selects the current interfaces more quickly.

IPSec Interface

This setting lets you select an existing IPsec interface. This setting works only if the phase 2 networks do not overlap.

IP Type

This setting specifies whether the connection uses IPv4 or IPv6.

IKE Type

This setting specifies the IKE type. Select from:

- **IKEv1**
- **IKEv2**: We recommend IKEv2 because IKEv2 is more robust, and easier to set up.

Connection Type

This setting specifies the connection type. Select from:

- **Tunnel**: This option creates a tunnel between the two tunnel endpoints. TBF automatically encrypts traffic when the source and destination match the *IPsec phase 2* on page 330 settings.
- **Transport**: This option encrypts only the payload of an IP packet. The protocol then encloses the encrypted payload in a normal IP packet. Traffic sent in transport mode is less secure than traffic sent in tunnel mode because the protocol does not encrypt the IP header in each packet. This mode encrypts traffic between the 2 entities and uses only a single *IPsec phase 2* on page 330 to define the transport parameters.

Encryption Type

This setting specifies the encryption type. Select from:

- **ESP**
- **AH**

ESP

Encapsulating Security Payloads (ESP) helps provide confidentiality, data-origin authentication, connectionless integrity, an anti-replay service (a form of partial sequence integrity), and limited traffic-flow confidentiality.

AH

Authentication headers (AH) help provide connectionless data integrity and data origin authentication for IP datagrams and helps protect against replay attacks.

Init Type

This setting specifies the init type. Select from:

- **Start/On Demand**: This option attempts to connect immediately and installs a policy to initiate a connection when the first packet flows. The policy confirms that the packets travel to the endpoint encrypted and also starts a reconnection when the tunnel is inactive.
- **Start**
- **Nothing**: This option puts phase 1 in responder-only mode.

Interface

This field creates an XFRM interface in the system for this IPsec tunnel. You can use the interface like any other interface. Manually create *routes* on page 209 from phase 2 to enter the interface. You can also set an IP address on the *interface settings* on page 92. The interface automatically sets to active or inactive when phase 1 connects or disconnects. The interface allows custom routes and metrics that the system removes and adds when the interface is active or inactive. You can also specify an existing XFRM interface as long as the phase 2 IP pairs do not overlap. This selection is useful for managing multiple tunnels in a single interface.

Remote Endpoints

This field accepts one or more IP addresses of the remote end. The system also supports hostnames, but requires a working DNS to resolve the hostnames. There is no DNS caching, so when the system reestablishes a connection, the system looks up the name again. This behavior allows DynDNS hostnames. A remote endpoint of `0.0.0.0` creates an `any` entry in case the remote IP address is dynamic. The system performs identification via identifier and PSK/certificate in that case.

Note: As an *initiator*, the system uses the first IP address to initiate the connection. As a *responder*, the system requires the initiator source address to match at least one of the specified addresses, subnets, or ranges. If you assign FQDNs, then the system resolves the FQDNs every time it performs a configuration lookup.

Phase 1 authentication

Phase 1 authentication provides different options to authenticate the IPsec connection.

Phase 1 Authentication provides different options to authenticate the connection.

Authentication methods

Authentication Method can be:

- **Pre Shared Key**
- **Certificate**
- **EAP TLS (Certificate)**
- **EAP MD5**
- **EAP MSCHAPv2**

Enter either a pre-shared key that is required to match on both ends or a certificate and the corresponding certificate authority (CA). Use **EAP MD5** and **EAP MSCHAPv2** for client authentication with a username and password.

Identifier settings

Local Identifier is the identifier of TBF and can be one of:

- **My IP Address**
- **IP Address (Custom IP)**
- **Other**

Remote Identifier is the identifier of the remote side and can be one of:

- **Any**
- **Peer IP Address**
- **IP Address (Custom IP)**
- **Other**

Two-factor authentication

Use **Use Two Authentications** to define a secondary authentication that has the same options as above.

Phase 1 settings

Advanced settings for IPsec phase 1 connections.

Phase 1 Algorithms

You can mix the algorithms. Add as many combinations as you like.

Lifetime

This setting matches the lifetime of the remote side.

Advanced Options

This section offers additional control over the IPsec connection.

Associate Firewall Rule

This option generates a firewall rule for you and keeps the rule up to date with any changes you make.

Rekey

If the connection is nearing expiration, then this setting allows TBF to start renegotiation. If you disable this setting, then only the remote side starts the negotiation.

Responder Only

If you do not want TBF to start the connection but to wait for the remote side to start the connection, then enable this setting.

Close Action

This setting specifies the close action. Select from:

- **Connect**: This option tries to connect immediately.
- **On Demand**: This option only initiates a connection when the first packet flows.
- **Nothing**: This option only closes the connection. This option is the default option.

NAT-T

This setting defaults to automatic so that IPsec determines the correct remote IP address.

MOBIKE

This setting is for the IKEv2 MOBIKE protocol. See *Strongswan Mobike* for a more detailed explanation.

IPComp

This setting enables or disables the IP payload compression.

Split Connection

For IKEv2, this option creates a new connection for each *IPSec phase 2* on page 330 entry. Typically, the system groups and sends the entries through a single connection. Depending on the endpoint, this grouping is unsupported, and you must enable the split connection. This setting is typically required for Cisco ASA.

Dead Peer Detection

This feature allows IPsec to check whether the remote endpoint is still alive.

DPD Delay

This setting defines the time between checks. The default is `10`.

DPD Timeout

If the DPD check was unsuccessful, then this setting defines the time before the system restarts a connection. The default is `30`.

DPD Action

This setting specifies the DPD action. Select from:

- **Restart**: This option attempts to restart immediately.
- **On Demand**: This option only initiates a new connection when the first packet flows.
- **Nothing**: This option only closes the connection without further action.

Traffic Selector

This setting specifies the traffic selector. You can only select this option in transport mode. Select from:

- **Any**
- **GRE**
- **VXLAN**

In cases besides **Any**, the system encrypts only the selected protocol or traffic between the 2 endpoints.

Fallback IPSec

This setting allows another IPsec to act as a fallback for this phase 1. The system selects each IPsec only once in a fallback scenario.

Ping Check

This setting enables or disables the ping check. If you select **Fallback IPSec**, then you can use this setting and the following ping settings. The ping check determines when the system uses the fallback IPsec.

Ping IP Address

This setting defines the ping IP address for the fallback check.

Ping Interval

This setting defines the interval between the checks in minutes. The default is `5`.

Ping Retries

This setting defines the number of retries before the system terminates and initiates the connection when the ping check is unsuccessful. The default is `3`. A value of `0` means disabled.

Ping Fallback

This setting defines whether there is a fallback to the primary tunnel when ping works again. This setting depends on whether you ping an outer or an inner IP address. If you ping an inner IP address, then the inner IP address is available on the fallback as well. If you ping the primary IP endpoint, then this setting checks the tunnel endpoint, and enabling the setting is advisable.

Ping Cron

This setting allows you to select a cron object.

Restart Gateway Change

This setting restarts the tunnel when the main routing table gateway changes. This setting is useful in multi-WAN setups because IPsec might not change the gateway automatically.

IPsec action example

This concept describes recommended practices for IPsec connection management and configuration examples.

IPsec has 3 different parameters to manage the start and reconnect behavior:

- **Init Type**
- **Close action**
- **DPD Action**

If you do not set these settings correctly, a combination of them can lead to duplicate tunnel connections.

Recommended configurations

2 sets of configurations make sense:

1. Start the connection immediately and restart on detected inoperability:

- **Init Type**: `Connect`
- **DPD Action**: `Restart`
- **Close Action**: `Nothing`

Note: If you do not use DPD, set **Close Action** to `Restart`.

2. Start the connection when traffic flows, which also confirms that no unencrypted traffic flows:

- **Init Type**: `On Demand`
- **DPD Action**: `Nothing`
- **Close Action**: `Nothing`

Encryption recommendations

Tip: For **Encryption**, use `AES-256 SHA 256 DH-16` as a minimum for **phase 1**. Do not use `AES-GCM` for phase 1, but use it for **phase 2** encryption as it helps provide a high speed for IPsec.

Related topics

- *IPsec multi-WAN setup* on page 329: Multi-WAN configuration with automatic failover

IPsec multi-WAN setup

This task describes how to create a multi-WAN setup with automatic failover to a backup WAN connection.

Creating a multi-WAN setup with automatic failover to a backup WAN requires **Restart Gateway Change**. The IPsec tunnel does not follow the gateway change to a different WAN. The restart disconnects and reconnects the tunnel to force the tunnel to the new active WAN connection.

Perform the following steps:

1. Navigate to the IPsec phase 1 configuration.
2. Enable **Restart Gateway Change** in the advanced settings.
3. Set up the multi-WAN setup with appropriate gateway priorities.

The IPsec tunnel automatically restarts when the **Main** routing table gateway changes, helping provide failover between WAN connections.

IPsec with failover

This task describes how to set up IPsec with high availability (HA) failover support.

Prerequisites:

Before you begin, verify that you have set up the HA setup as described in *high availability* on page 374.

To set up the IPsec with failover support, set up the server and clients as described in *IPsec* on page 322.

Perform the following steps:

1. Set up the HA setup as described in *high availability* on page 374.
2. Enable the configuration sync option.
3. Note the virtual IP address of the interface that the IPsec server runs on.
4. On the remote device, navigate to **VPN > IPsec**.
5. Set up the VPN connection.
6. Enter the virtual IP address that you noted earlier as **Remote Endpoint** instead of the real address of the device. This way, the IP address of the remote device does not change when the secondary router takes control.

The IPsec connection now supports HA failover with switching between primary and secondary devices.

Performance

This topic describes performance optimization tips for IPsec connections.

CPU utilization optimization

Tip: TBF assigns each SA to a single CPU. When you have multiple phase 2 entries, the **Split Connection** option helps improve performance. The option creates 1 SA for each phase 2 and therefore utilizes more CPUs. The option requires you to enable it on both ends of the connection.

IPsec firewall rules

This topic covers the required firewall configuration for IPsec connections.

Firewall rule requirements

Important: Create firewall rules on the **IPsec** tab to allow network traffic from the other end of the IPsec. If you chose to create an interface, then these rules belong in the newly created `XFRM_XXX` interface. A detailed explanation on how to create firewall rules for an interface is available in the *firewall rules (forward and input)* on page 131.

IPsec phase 2

This reference describes how to set up IPsec phase 2 including network addresses, encryption algorithms, and ping check settings for tunnel monitoring.

Overview

Create phase 2 entries under the current phase 1 entry. The IPsec phase 2 setup involves 3 main areas:

- *Network setup* on page 331: Define local and remote network mappings
- *Algorithm settings* on page 331: Set up encryption ciphers and lifetime parameters
- *Ping check settings* on page 332: Set up tunnel monitoring and automatic reconnection

IPsec phase 2 network setup

This topic describes how local and remote network definitions work in IPsec phase 2 setups to establish tunnel mappings.

IPsec phase 2 requires a separate entry for each pair of local and remote networks you want to connect. TBF creates an automatic mapping between these network pairs. This mapping allows traffic to flow through the IPsec tunnel.

Local Network

This setting specifies the network or the specific address on TBF that remote users reach through the tunnel.

Remote Network

This setting specifies the network or the specific address that TBF reaches on the other end of the tunnel.

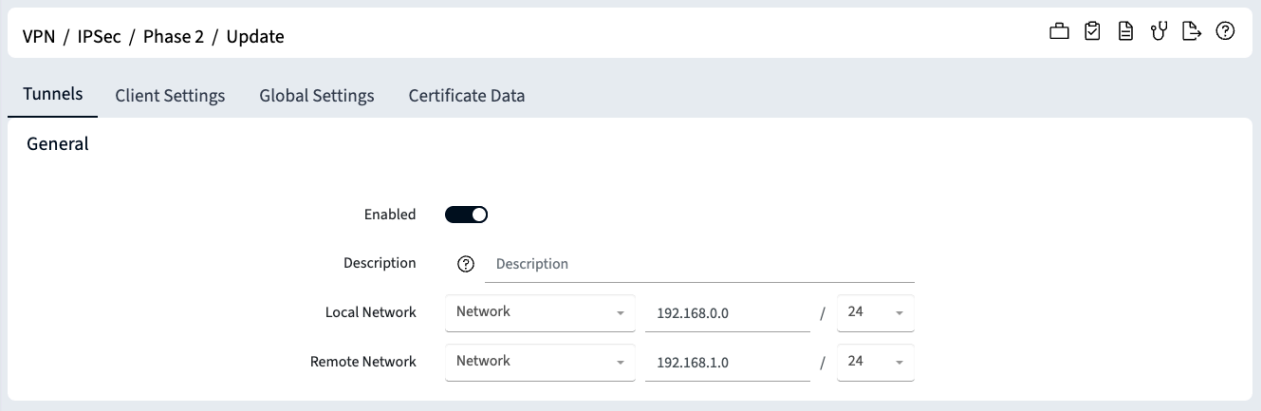


Figure 240: IPSEC phase 2 addresses

IPsec phase 2 algorithm settings

Set up encryption ciphers and lifetime settings for IPsec phase 2 security associations.

Algorithm settings

Ciphers

This setting sets 1 or more encryption algorithms to assist in establishing security for the data transmission. Combine different cipher types to meet the security requirements. Add multiple cipher combinations as needed for the network setup.

Lifetime

This field sets the duration for how long the security association remains active. This value matches the lifetime setting on the remote endpoint to help provide proper connection establishment and maintenance.

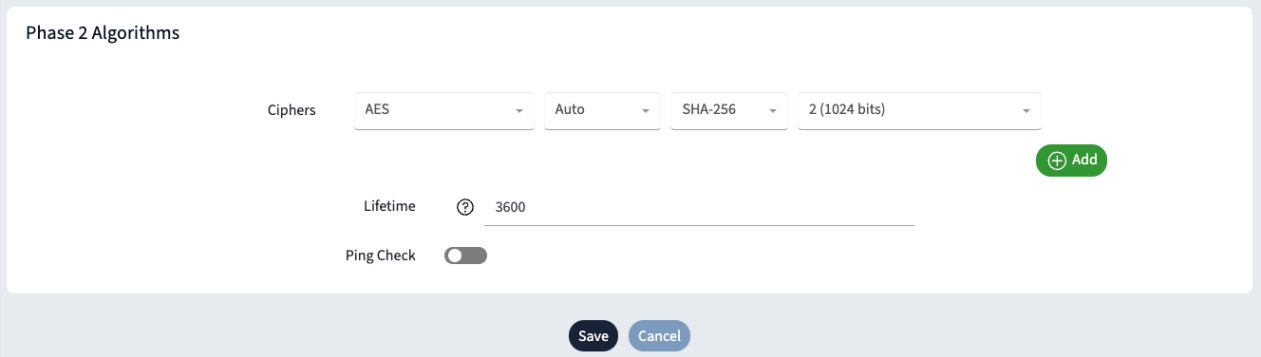


Figure 241: IPSEC phase 2 algorithms

IPsec phase 2 ping check settings

Set ping monitoring settings to verify tunnel connectivity and enable automatic reconnection when connectivity issues occur.

Settings

Ping Check

This setting enables or disables a ping check against an IP address on the remote side of the tunnel. Verify that the phase 2 network definition includes at least 1 IP address of TBF.

Ping IP Address

This field specifies the remote IP address to ping. The address is within the remote network range.

Ping Interval

This field determines how many seconds pass between ping checks.

Ping Retries

This field specifies how many ping retries occur before TBF disconnects and reconnects phase 2. If you set this value to `0`, then TBF does not disconnect or reconnect when ping errors occur.

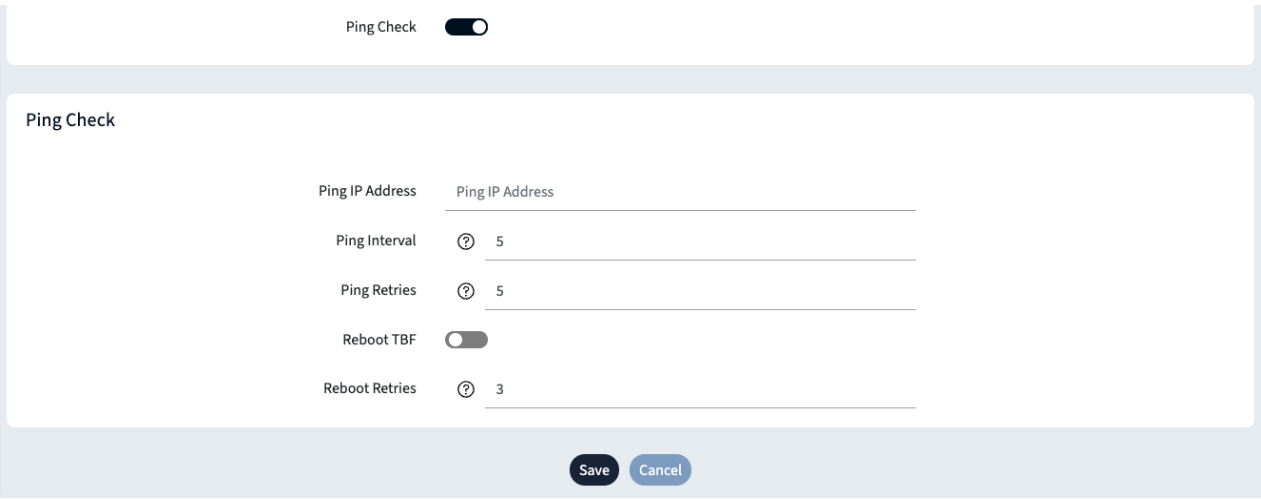


Figure 242: IPSEC phase 2 ping check

Custom script integration

Note: If you need to create custom behavior for the ping check, then add a custom script to the location `/usr/local/bin/check_ipsec_custom`. The script receives 2 environment variables: `PHASE2` with the phase 2 name and `RESULT` with the ping result. `0` indicates success, and `1` indicates a detected failure.

IPsec client settings

This topic describes how to set IPsec client settings including EAP RADIUS servers, authentication secrets, and IP address pools.

Access client settings

Find the IPsec client settings under **VPN > IPSec > Client Settings**.

Configuration areas

General

Set EAP RADIUS server settings for IPsec client authentication. See the *EAP RADIUS server settings* on page 333.

Secrets

Define authentication credentials for client connections. See the *Authentication secrets* on page 333.

Pools

Set IP address pools for client assignment. See the *IP address pools* on page 333.

EAP RADIUS server settings

Set EAP RADIUS server settings for IPsec client authentication.

In the **General** section, set an EAP RADIUS server from the available authentication servers. This server handles RADIUS authentication for IPsec tunnel connections.

When you set up an EAP RADIUS server for IPsec clients, TBF uses this server to authenticate client connections during the IPsec tunnel establishment process. Properly set up the authentication server and make it accessible for successful client authentication.

Authentication secrets

Define usernames and passwords for IPsec client authentication.

In the **Secrets** section, define usernames and passwords for client authentication. Either pre-shared keys or EAP MD5/MSCHAPv2 authentication methods use these credentials. The identifier matches the remote identifier for successful authentication.

Note: Secrets apply to any defined phase 1 configuration that matches the criteria.

When you set up authentication secrets, verify that the credentials match the credentials that the remote IPsec peer expects. TBF uses these credentials during the authentication phase of tunnel establishment.

IP address pools

Set up IP address pools for IPsec client assignment.

Pool configuration

The **Pools** section defines IP address pools for specific phase 1 configurations. TBF uses the DHCP protocol to assign these IP addresses to clients during connection establishment.

Address assignment process

When an IPsec client connects, TBF automatically assigns an IP address from the pool according to the phase 1 configuration match. The DHCP protocol handles the address assignment and makes proper network connectivity available for the client.

IPsec route based (VTI/XFRM)

Route-based VPNs offer flexible IPsec processing through routing packets to specific IPsec interfaces rather than relying solely on negotiated policies.

In TBF, you can set up route-based VPNs. With this approach, IPsec processing is not solely dependent on negotiated policies. Instead, you control the processing by routing packets to a specific IPsec interface.

Route-based IPsec involves several key concepts:

- *VTI and XFRM interfaces* on page 334: Virtual tunnel interfaces that offer automatic route management
- *Route-based IPsec configuration* on page 334: Setup procedures for failover and routing
- *Route-based IPsec interface capabilities* on page 334: Network features and MTU configuration

VTI and XFRM interfaces

Virtual Tunnel Interface (VTI) and XFRM interfaces provide automatic route management for IPsec tunnels.

You can create an interface from the IPsec tunnel, commonly known as VTI or the newer XFRM interface. This interface provides a key advantage: TBF automatically removes static routes defined on the interface when the interface goes down. The *up* and *down* status of the interface is directly linked to whether phase 1 is active or inactive.

The interface does not modify the IPsec traffic itself. This behavior means the remote end of the tunnel does not require any knowledge of the interface configuration.

Note: You can set up route-based IPsec on 1 end of the connection only.

Route-based IPsec configuration

Route-based IPsec configuration enables failover setups and flexible routing through manual route creation.

Route-based IPsec configuration enables failover configurations where multiple IPsec tunnels carry identical routes but with different metrics. You can enable the interface in the phase 1 settings. The phase 2 configuration can include any network or combination of networks.

Manually create *routes* on page 209 from phase 2 to direct traffic into the interface. You can also assign an IP address in the *interface settings* on page 92.

Route-based IPsec interface capabilities

Route-based IPsec interfaces support standard network interface features including firewall rules, NAT operations, and MTU configuration.

Network interface features

The interface functions like any other network interface and supports firewall rules, NAT rules, and services. This capability supports DNAT/SNAT operations before traffic enters the tunnel.

MTU configuration

You can also specify the maximum transmission unit (MTU) for the IPsec interface. With this setting, the interface fragments packets to the appropriate size before it tunnels the packets.

Set up GRE over IPsec

This task describes how to set up GRE/GRETAP over IPsec to create encrypted site-to-site connections with enhanced assistance in establishing security.

This task describes how to set up GRE/GRETAP over IPsec in TBF. In many cases, you do not necessarily need a GRE/GRETAP tunnel within the IPsec tunnel because you can add the remote subnets directly to the phase 2 configuration.

Perform the following steps:

1. Create a GRE interface as described in *tunnel* on page 107.
2. Confirm that you enable **Assign to a new Interface**.
3. Set up the *interface* on page 92.
4. Set the GRE endpoints and the internal IP addresses (local and remote tunnel IP addresses).
   For example, use `10.10.10.1` (local) and `10.10.10.2` (remote).
5. Create a new *IPsec phase 1* on page 324.
6. Select **Connection TypeTransport**.
7. For **Interfaces**, select the WAN interface that you use to send the GRE packets to the endpoint.
8. Set **Remote Endpoints** to the remote tunnel IP address (the external tunnel IP address of the remote end).
   For example, `1.2.3.4`.
9. To encrypt only the GRE traffic, enable **GRE over IPSec**.
   Set the other settings according to the IPsec endpoint requirements.
10. Create an *IPsec phase 2* on page 330 with matching encryption settings.

When the IPsec tunnel is established, the tunnel encrypts the GRE packets to the remote endpoint.

To direct additional traffic through the GRE tunnel, add *routes settings* on page 209. To allow the traffic, add *firewall rules (forward and input)* on page 131.

Related information
Site-to-site encryption on page 335
Set up GRE over IPsec failover on page 336

Site-to-site encryption

Understand how to combine multiple physical locations into 1 large company network with IPsec GRE/GRETAP tunnels.

Follow the steps in *set up GRE over IPsec* on page 335. See the *bridging scenarios* on page 99. This setup creates a GRE/GRETAP tunnel that an IPsec tunnel encapsulates and encrypts. The networks then appear as if the networks were 1 physical network.

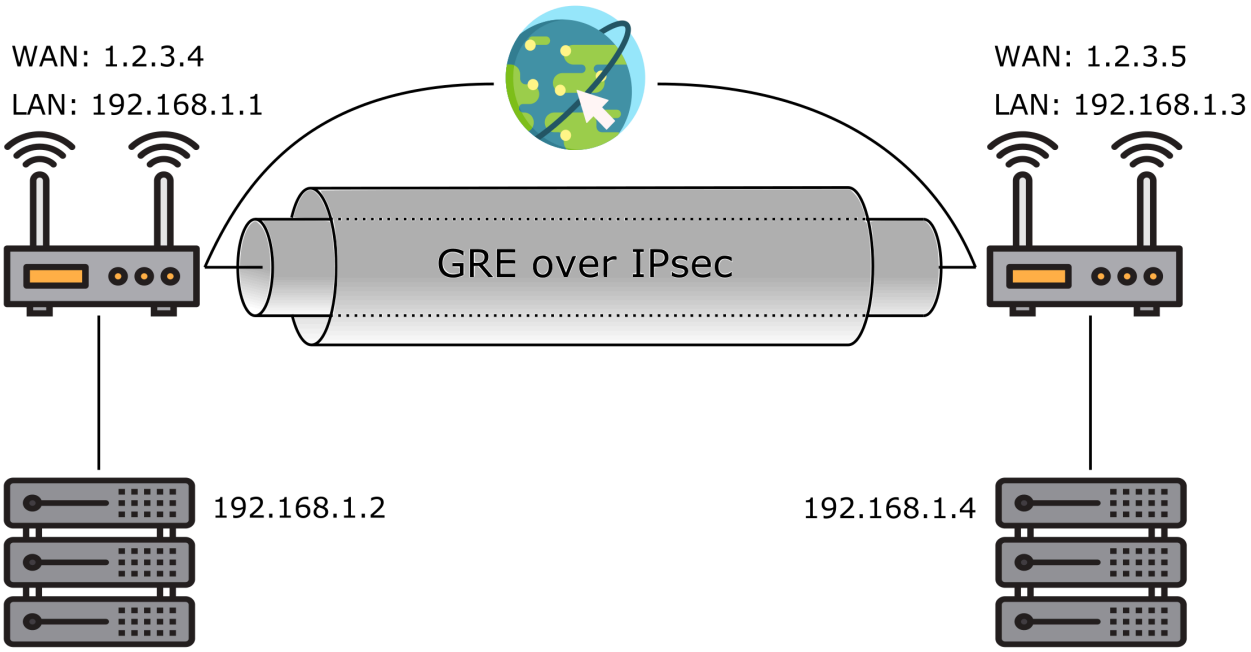


Figure 243: GRE over IPSEC tunneling

This image uses icons by *srip* and *Freepik* from *Flaticon*.

Alternative approach

As an alternative to GRE over IPsec, use a standalone IPsec tunnel. Add the remote and local networks to connect as a phase 2 entry in the IPsec tunnel. In the example below, the local LAN connects to the remote network with the public IP address `1.2.3.4` and the private IP address range `192.168.1.0`.

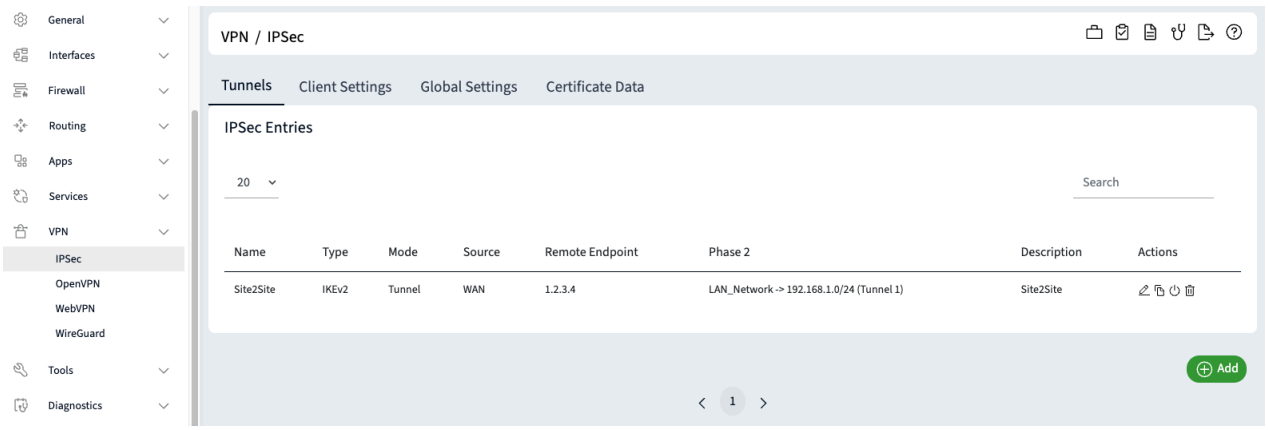


Figure 244: IPSEC tunnel overview

Set up GRE over IPsec failover

Set up high availability (HA) for GRE over IPsec tunnels to maintain connectivity during router failover.

Prerequisites:

Before you begin, verify that you have:

- A working GRE over IPsec tunnel as described in *set up GRE over IPsec* on page 335
- An HA setup as described in *high availability* on page 374

In an HA setup, include the GRE over IPsec tunnel to keep the tunnel operational even when the secondary router takes control.

Perform the following steps:

1. Set up the tunnel without failover support.
2. Set up the HA setup as described in *high availability* on page 374.
3. Note the virtual IP address that you assigned for the interface the tunnel runs on.
4. Add the virtual IP address to both tunnels, as described in *IPsec with failover* on page 324 and *GRE with failover* on page 107.

The tunnel now works across the TBF devices that the HA setup includes.

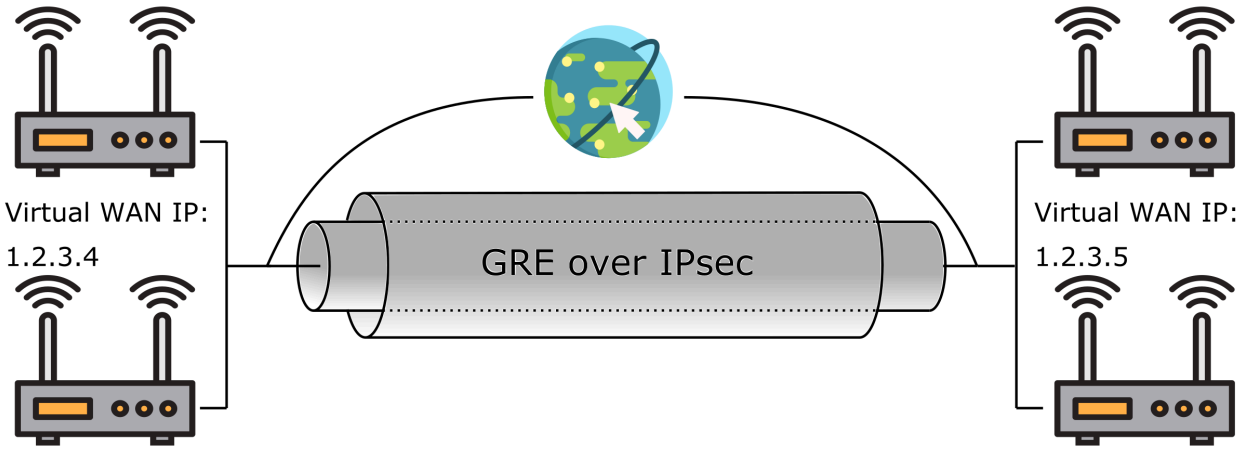


Figure 245: GRE over IPSEC tunneling with failover

IPsec troubleshooting

This topic provides diagnostic and troubleshooting information for resolving IPsec tunnel connection issues.

You are experiencing detected problems with IPsec tunnel connections, such as detected reconnection failures or detected connection errors.

Use the following troubleshooting resources to diagnose and resolve IPsec issues:

- *IPsec troubleshooting steps* on page 337: Follow systematic troubleshooting procedures
- *NO_PROPOSAL_CHOSEN error* on page 338: Resolve phase 1 parameter mismatch issues
- *HASH_V1 payload error* on page 338: Fix pre-shared key mismatch detected problems

IPsec troubleshooting steps

Follow these troubleshooting steps to resolve detected reconnection problems with the IPsec tunnel.

Following these troubleshooting steps often resolves detected reconnection problems with the IPsec tunnel. Try each step in order and test the connection after completing each one.

Perform the following steps:

1. Verify that the settings on both tunnel endpoints match.
2. Change the **Init Type** setting from **Connect** to **On Demand**.
3. Set up a cron job under **Services > Cron** that sends a ping to an IP address on the remote side of the tunnel every 5 minutes.

   This cron job helps maintain an active connection. Use the command `ping -c 3 IP_ADDRESS > /dev/null` for this purpose.

NO_PROPOSAL_CHOSEN error

This topic describes how to resolve the NO_PROPOSAL_CHOSEN error that occurs when phase 1 encryption settings do not match between tunnel endpoints.

You receive the following error message:

```
parsed INFORMATIONAL_V1 request 0 [ N(NO_PROP) ] received NO_PROPOSAL_CHOSEN error notify
```

This error message typically indicates that the phase 1 encryption settings do not match between the 2 tunnel endpoints. This mismatch causes the remote side to reject the connection proposal.

- Verify that both sides have identical phase 1 settings.
- Check that encryption algorithms, hash algorithms, and DH groups match on both endpoints.
- Verify that authentication methods are consistent across both tunnel endpoints.

HASH_V1 payload error

This topic describes how to resolve the HASH_V1 payload error that occurs when pre-shared keys do not match between tunnel endpoints.

You receive the following error message:

```
invalid HASH_V1 payload length, decryption failed?
```

This detected error occurs when one of the tunnel endpoints has an incorrect pre-shared key (PSK). Because the system uses the PSK to generate the key material that encrypts IKEv1 packets, mismatched PSKs help prevent proper decryption of the communication.

- Verify that both tunnel endpoints have identical PSKs set up.
- Check for any extra spaces or hidden characters in the PSK configuration.
- Verify that you entered the PSK correctly on both sides, observing case sensitivity.
- Consider regenerating the PSK if you cannot identify the mismatch.

OpenVPN

OpenVPN implements virtual private network (VPN) techniques using SSL/TLS to assist in establishing security for point-to-point or site-to-site connections.

OpenVPN is an open-source VPN solution that uses SSL/TLS protocols for key exchange and encryption. OpenVPN offers a flexible way to assist in establishing security when creating encrypted tunnels over public networks.

OpenVPN has several characteristics:

- **Flexibility**: Works over UDP or TCP and can traverse NAT and firewalls
- **Assistance in establishing security**: Uses SSL/TLS with support for various encryption algorithms
- **Cross-platform**: Available on multiple operating systems
- **Scalability**: Supports both site-to-site and remote access scenarios

OpenVPN can operate in two modes:

Server mode

Acts as a VPN server accepting connections from multiple clients

Client mode

Connects to a remote OpenVPN server as a client

OpenVPN topics

The following topics contain detailed information about OpenVPN configuration and usage:

- *OpenVPN general* on page 339: Overview and basic concepts
- *OpenVPN settings* on page 339: Configuration parameters and options
- *OpenVPN export* on page 350: Export client configurations
- *OpenVPN override* on page 351: Client-specific configuration overrides

OpenVPN general

OpenVPN implements virtual private network (VPN) techniques to assist in establishing security for point-to-point or site-to-site connections using server and client configurations.

Regardless of the connection type, you need to run an OpenVPN server and an OpenVPN client. However, the device they run on can vary. For a site-to-site connection, you typically run the server on a local TBF device and the client on a remote TBF device or other router that supports OpenVPN. The VPN tunnel then forwards traffic from the remote network to the local network.

Alternatively, create *point-to-point* connections. For example, if you want to use a phone while traveling and still have access to the local network, open a VPN tunnel to the local network from any device that supports OpenVPN. This approach is also useful when you want the traffic to be routed through the local network for security reasons or to help bypass certain restrictions when browsing from specific locations.

TBF helps provide you create:

- OpenVPN **Servers**
- OpenVPN **Router Clients**

To assist in establishing security for a site-to-site connection, 1 site is required to have an OpenVPN server set up, while the other site is required to have an OpenVPN router client set up.

Note: To establish a working connection, both sites are required to use the same device mode, authentication mode, encryption algorithms, digest algorithms, and tunnel network configuration.

OpenVPN settings

This topic covers settings for OpenVPN servers and clients.

Find the OpenVPN settings under **VPN > OpenVPN**.

Prefill from File

This setting automatically populates the configuration fields using an existing OpenVPN configuration file.

Authentication Mode

This setting determines how users authenticate with the VPN. Select from:

- **Certificate**: With this mode, the OpenVPN server requires a server certificate, and the OpenVPN client requires a client certificate. Both certificates are issued by the same certificate authority (CA). You can also enable **Additional User Auth.** to implement two-factor authentication using both a certificate and user credentials. The same requirements apply to the additional user authentication as described for user-only authentication below.
- **User**: With this mode, both the server and client require certificates from the same CA. Additionally, the client provides a **Username** and **Password** under **User Authentication**. This user account exists as a system user on the TBF device where the OpenVPN server runs. The user also requires the **User permission** *open vpn - OpenVPN User Login*, which you can assign under **System > User**. Alternatively, you can add the user to the **OpenVPN Access** group, which automatically grants the same user permission.
- **Shared Key**: With this mode, both the server and client use the same shared key. When you create the OpenVPN server, TBF automatically generates a shared key. You copy this shared key to the OpenVPN client configuration.

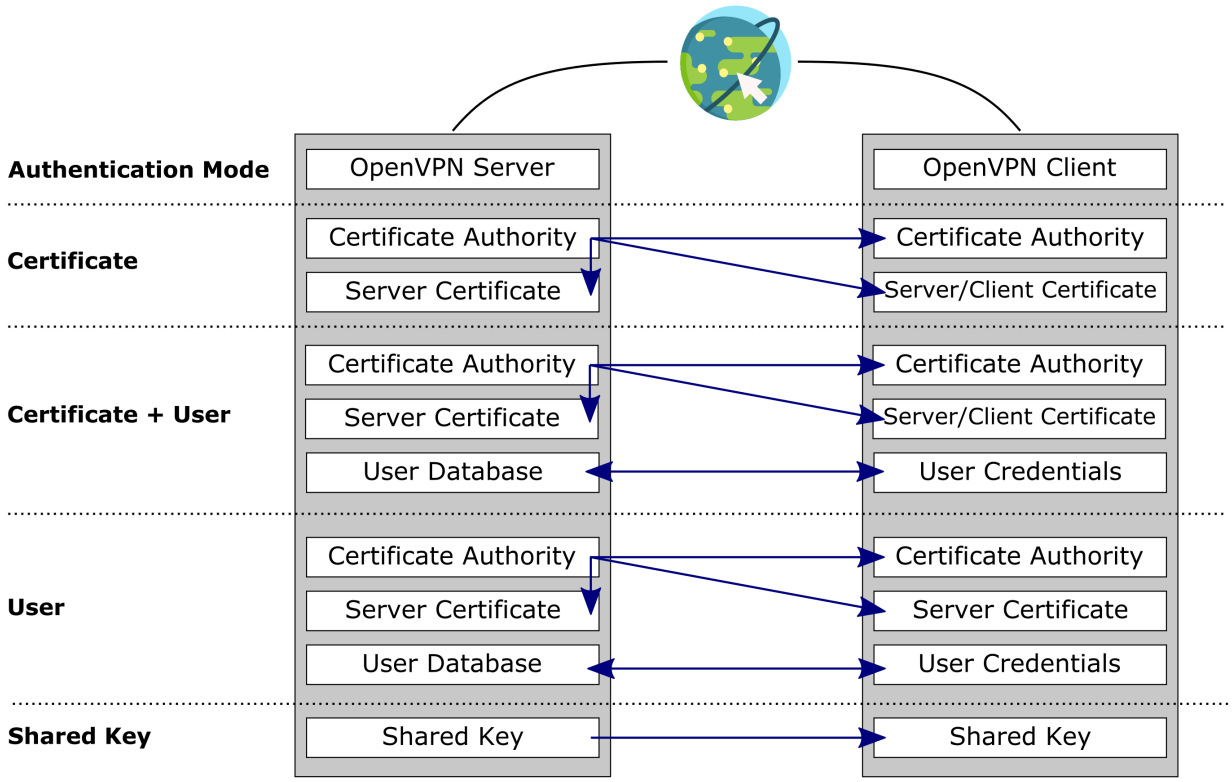


Figure 246: OpenVPN authentication modes

This image uses icons by *Freepik* from *Flaticon*.

Protocol

This setting determines the network protocol (UDP or TCP) and IP version support. You can set it for IPv4 only, IPv6 only, or both.

Note: Both IPv4 and IPv6 options are multihome only. This means that you cannot specify a particular interface and you can only select **All**.

Device Mode

This setting determines the VPN tunnel type: tun (layer 3 routing) or tap (layer 2 bridging).

Interface

This setting specifies which network interface the OpenVPN server uses, or you can set it to **All**. If the interface uses a VRRP virtual IP address in an HA setup, then only the master device that holds the IP address has OpenVPN active. The slave device remains on standby for OpenVPN.

Local Port

This setting defines the port number for the OpenVPN service. If you leave it blank, then TBF automatically assigns a random port.

Use TLS Key

This setting enables Transport Layer Security authentication and is only available in **Certificate** or **User** authentication modes. Both the server and the client use the same TLS key. TBF automatically generates this key when you create an OpenVPN server.

Encryption Algorithm

This setting determines the encryption method used to assist in establishing security for the VPN tunnel. You can select from various algorithms or select **none**. Both the server and client use the same algorithm.

Enable NCP

This setting enables Negotiable Cryptographic Parameters, which allow the server and client to negotiate encryption algorithms. With **NCP Algorithm(s)**, you can select multiple algorithms. OpenVPN uses the algorithms in the order you specify.

Note: This feature is only available for OpenVPN 2.4 and later versions.

Auth. Digest Algorithm

This setting specifies the authentication digest algorithm. Both the server and the client use the same algorithm.

Note: If you select **none**, then you cannot use TLS authentication.

IPv4 Tunnel Network

This setting defines the IPv4 tunnel network for the connection.

IPv6 Tunnel Network

This setting defines the IPv6 tunnel network for the connection.

Remote Network(s)

This setting specifies multiple remote networks that clients can access through the VPN tunnel.

Compression

This setting compresses tunnel packets to reduce bandwidth usage. Select from:

- Omit Preference (Use OpenVPN Default)
- LZ4 Compression [compress lz4]
- LZ4 Compression v2 [compress lz4-v2]
- LZO Compression [compress lzo, equivalent to comp-lzo yes for compatibility]
- Enable Compression (stub) [compress]
- Omit Preference, + Disable Adaptive LZO Compression [Legacy style, comp-noadapt]
- Adaptive LZO Compression [Legacy style, comp-lzo adaptive]
- LZO Compression [Legacy style, comp-lzo yes]
- No LZO Compression [Legacy style, comp-lzo no]

Type-of-Service

This setting sets the TOS IP header value of tunnel packets to match the encapsulated packet value.

Topology

This setting determines how the system assigns IP addresses to the virtual network adapter.

Custom Options

This field defines custom parameters for the configuration.

Gateways

This setting helps you specify which gateways the system creates. You can use these gateways to create additional routes or routing tables.

Send/Receive Buffer

This setting determines the buffer size for sending and receiving data through OpenVPN. The default buffer size may be insufficient depending on the hardware capabilities and network connection speeds.

Log Level

This setting controls the amount of detail in OpenVPN logs. Level 0 is silent except for fatal detected errors, level 4 is suitable for general use, levels 5 and 6 help debug connection detected errors, and level 9 has extremely detailed logging.

Keepalive Interval

This setting determines how often in seconds the system sends keepalive packets to maintain the connection. The default is `10`.

Keepalive Timeout

This setting determines how long in seconds the system waits for a keepalive response before considering the connection lost. The default is `60`.

Renegotiation

This setting specifies how often in seconds the system renegotiates the data channel encryption key to assist in establishing security. The default is `3600` (1 hour). A value of `0` disables automatic renegotiation.

You can export the settings as an Excel spreadsheet with the export option in the upper-right corner.

Related topics

For additional configuration options, see:

- *OpenVPN server-only settings* on page 342
- *OpenVPN client-only settings* on page 343
- *Set up an OpenVPN site-to-site connection* on page 344
- *OpenVPN firewall rules* on page 350

OpenVPN server-only settings

This topic covers settings that are available only for OpenVPN servers.

Authentication Server

Use this setting when you have user authentication enabled. Use a predefined *authentication server* on page 80 for user verification. You do not need to create user accounts in TBF, as the system relies solely on the response of the authentication server.

Radius Attributes

If the authentication server is a RADIUS server, you can use the RADIUS attributes to set client override settings. TBF supports the following attributes:

- `Framed-IP-Address`
- `Framed-Route`

Use CRL

This setting enables or disables the use of the certificate revocation list (CRL) from the selected peer certificate authority, which checks for revoked certificates.

Two Factor Authentication

Enable this setting when using user authentication. The system uses Time-based One-Time Password (TOTP) authentication only, not traditional one-time passwords. Verify that all relevant users have two-factor authentication enabled, as they cannot log in otherwise.

Note: This feature works without enabling two-factor authentication for the GUI.

Certificate Depth

This setting defines the depth to which the system accepts certificate-based client logins.

DH Parameter Length

This setting specifies the Diffie-Hellman parameter length used to assist in establishing security for key exchange between the server and clients.

ECDH Curve

This setting specifies the elliptic curve to use for key exchange.

Redirect Gateway

This setting forces all client traffic to route through the VPN tunnel, making the VPN server the default gateway for connected clients.

Concurrent connections

This setting limits the maximum number of clients that simultaneously connect to this OpenVPN server.

Push Compression

This setting forces the server to push the selected compression settings to all connecting clients.

Inter-client communication

With this setting, clients that connect to the same server communicate directly with each other.

Note: This communication requires appropriate firewall rules.

Duplicate Connection

This setting supports multiple simultaneous connections from clients using the same certificate common name, which is useful in certain deployment scenarios.

Associate Firewall Rule

This setting automatically creates and maintains firewall rules associated with this OpenVPN server.

Dynamic IP

With this setting, connected clients maintain their VPN connection even when their public IP address changes, which is useful for clients with dynamic internet connections.

DNS Settings

This setting defines advanced DNS options. Specify up to 4 DNS servers.

DNS Default Domain

This setting specifies a default domain name that the system provides to connected clients.

Block Outside DNS

This setting forces Windows 10 clients to block access to DNS servers outside the VPN tunnel, helping provide that they only use the DNS servers provided by the VPN.

Register DNS

This setting helps Windows clients properly recognize and use the DNS servers pushed by the VPN server.

NTP Settings

Use this setting to set up up to 2 Time Protocol (NTP) servers that the system pushes to connected clients for time synchronization.

OpenVPN client-only settings

This topic covers settings that are available only for OpenVPN clients.

Server host or address

This field specifies the IP address or hostname of the OpenVPN server.

Server Port

This field specifies the port number that the OpenVPN server is listening on for incoming connections.

Limit outgoing bandwidth

This field defines the maximum outgoing bandwidth for this VPN tunnel. Leave this field empty for unlimited bandwidth. The value is between 100 bytes/sec and 100 Mbytes/sec.

Do not pull routes

This setting prevents the server from automatically adding routes to the routing table of the client.

Do not add/remove routes

This setting prevents the automatic addition or removal of routes.

Proxy host or address

This field specifies the address of an HTTP proxy server that this client uses to connect to the remote OpenVPN server.

Proxy Port

This field specifies the port number of the proxy server. The default is `1080`.

Proxy Authentication

This setting enables authentication with the proxy server using a username and password.

Set up an OpenVPN site-to-site connection

Establish a stable OpenVPN connection between 2 or more physically separated sites by running an OpenVPN server and the associated clients.

Site-to-site connections

Run an OpenVPN server and the associated clients to establish a stable OpenVPN connection between 2 or more physically separated sites. This section shows you how to set up a basic OpenVPN site-to-site connection on the TBF device.

Perform the following steps:

1. If you have not yet created a certificate authority (CA), then create one.

   Navigate to **General > Certificates > CAs**.

   Select **Add**.

2. Fill in the required information and select the cryptographic parameters.

   The **Common Name** typically specifies the identifier of the organization. Select a **Key Size** and an **Algorithm**. Verify that the client devices support the selected key lengths and algorithms, as some devices do not support larger values. This limitation does not apply when connecting between two TBF devices.

3. To create the CA, select **Save**.

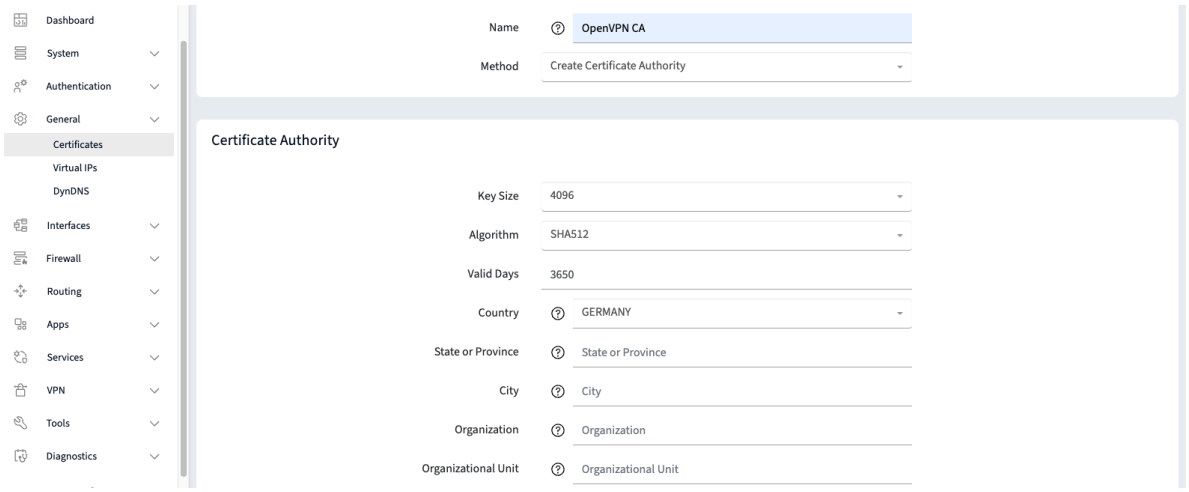


Figure 247: OpenVPN CA creation

4. Create a server certificate for the OpenVPN server.

   Navigate to **General > Certificates > Certificates**.

   Select **Add**. Select the newly created CA and select **Server** as the **Type**.

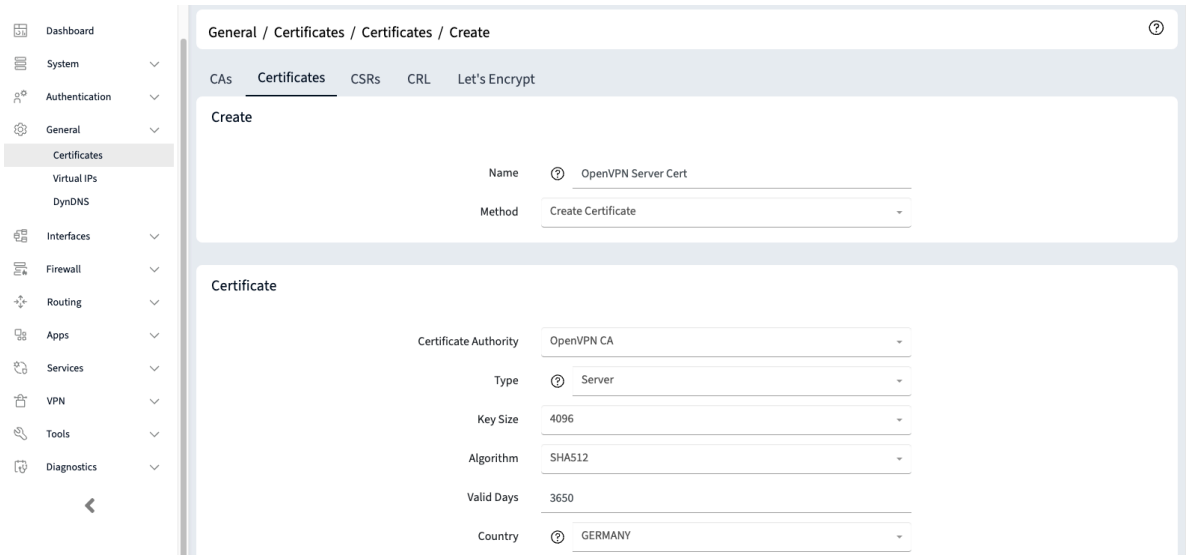


Figure 248: OpenVPN server certificate creation

5. Create the OpenVPN server configuration.

   Navigate to **VPN > OpenVPN**.

6. Select **Add**

7. Set the basic server settings.

   Define a descriptive **Name** and select an **Authentication Mode** (refer to the descriptions above). This example uses certificate + user authentication mode. For some authentication modes, you can set up the connection as a peer-to-peer connection that allows only 1 client. In a site-to-site connection with only 1 remote location, you enable this setting.

Under **Protocol**, you can select the network protocol. OpenVPN typically uses UDP because UDP is faster and more reliable than TCP. If you need IPv6 connectivity, then select IPv6 connectivity here. In most cases, IPv4 is sufficient.

Under **Device Mode**, select from **tun** and **tap**. **tun** mode is more widely supported and compatible with mobile clients such as iOS and Android devices, and **tun** mode is generally more stable than **tap** mode.

The **Interface** for the server is typically WAN or multihome in a site-to-site setup that communicates over the internet. You can leave the **Port** at its default value unless that port is already in use on the network.

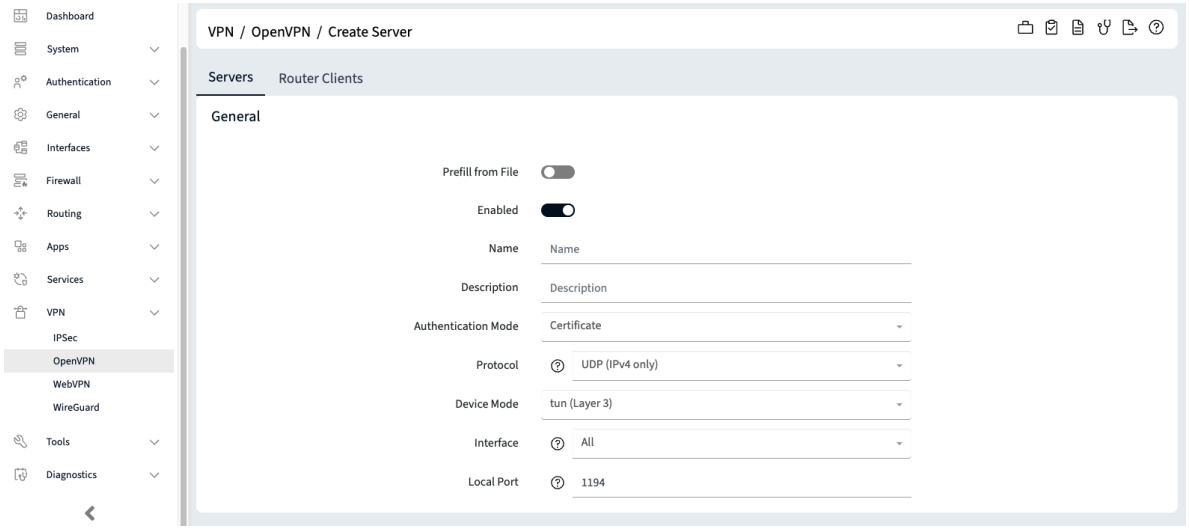


Figure 249: OpenVPN server general

8. Set the certificate settings.

Under **Certificate**, select the CA and the newly created OpenVPN server certificate. Leave **Certificate Depth** at **One** for a basic setup like this. If you want the server to recognize revoked certificates in the future, then enable **Use CRL**. This helps maintain VPN security, even if a certificate is compromised and you replace the certificate. You can revoke certificates in the Certificate Manager.

Because this setup uses certificate + user authentication mode, enable **User Authentication**.

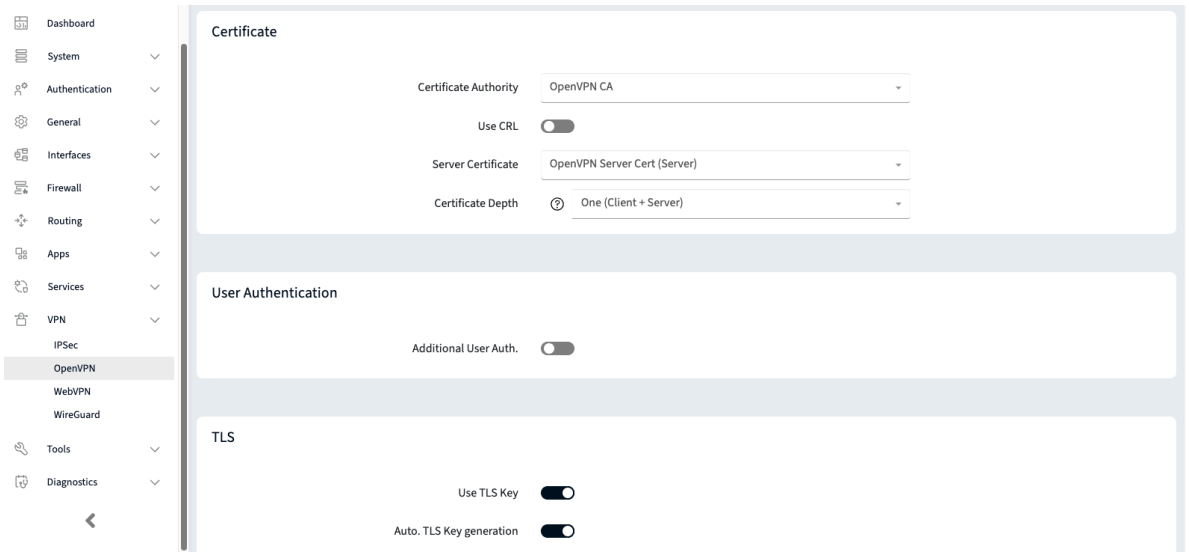


Figure 250: OpenVPN server certificate

9. Set the cryptography settings.

   Take special note of the client devices' capabilities. Some devices do not support key lengths or cryptographic algorithms. If you **Enable NCP**, then specify that the OpenVPN server negotiates alternative algorithms with the client if the client does not support the primary algorithm.

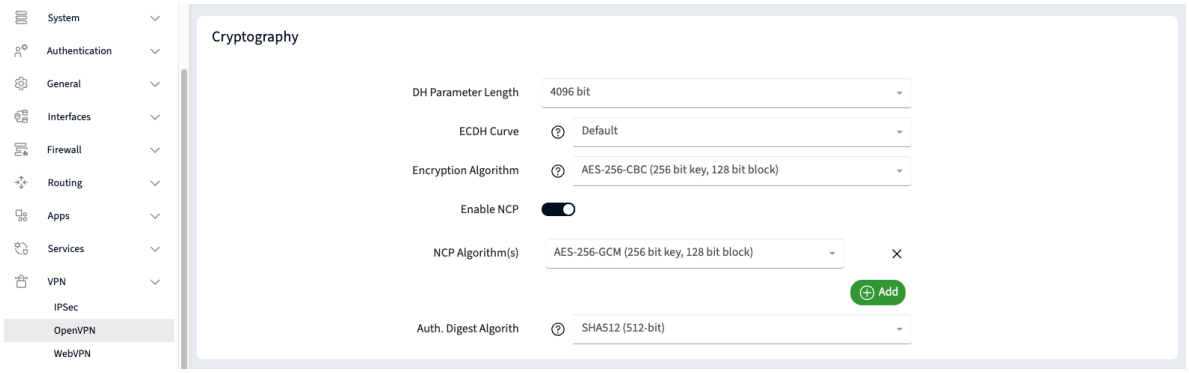


Figure 251: OpenVPN server cryptography

10. Set the tunnel network settings.

    The address range you define under **Tunnel Network** determines the IP addresses the system assigns to VPN clients. Specify an address range that does not conflict with the existing local network. If you have only a single client connecting to the VPN, then select single IP address (range 30). The system automatically sets the maximum number of concurrent connections to `1`.

    If you want to force client traffic through the VPN tunnel, then enable **Redirect Gateway**. This setting routes traffic through the tunnel regardless of its destination. If you leave this disabled, then the system only forwards traffic destined for networks on the other side of the tunnel through the VPN.

    Under **Remote Network**, you can specify the local networks that clients can access via the VPN connection. To create a multi-site network that functions as 1 unified local network, enter the local network addresses here.

    If you enable **Compression** for the tunnel, then you save bandwidth at the cost of increased processing power usage. For high-speed connections, this setting introduces additional latency. Refer to the compression mode descriptions above for details. With **Push Compression**, you can force clients to use the compression settings you specify instead of their own configuration.

    Inter-Client communication remains disabled in this example because the system limits the number of clients to 1. Therefore, there are no other clients to communicate with.

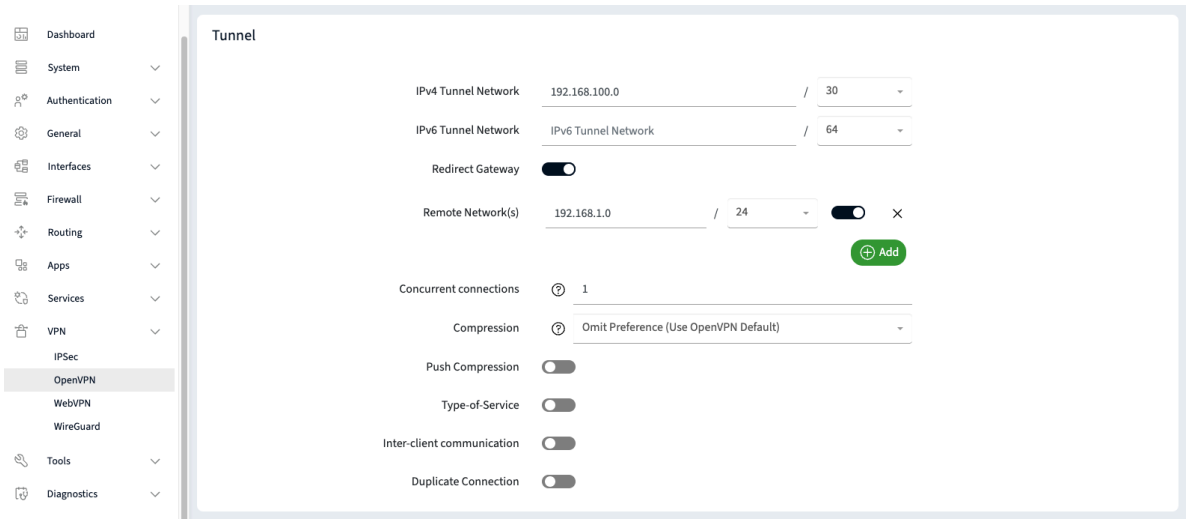


Figure 252: OpenVPN server tunnel

11. Set the advanced settings.

    If the clients operate on dynamic IP addresses, then enable **Dynamic IP** to maintain the connection even when a client's IP address changes. Otherwise, the client reconnects every time its IP address changes.

    Under **Topology**, you can specify whether the server creates a separate subnet with only 1 IP address per client or a common subnet for clients. Because this configuration has only 1 client, this setting is not relevant. Under **Gateways**, you can define whether to create gateways for both IPv4 and IPv6 or only 1 of them. In this IPv4-only configuration, set this to **IPv4 Only**.

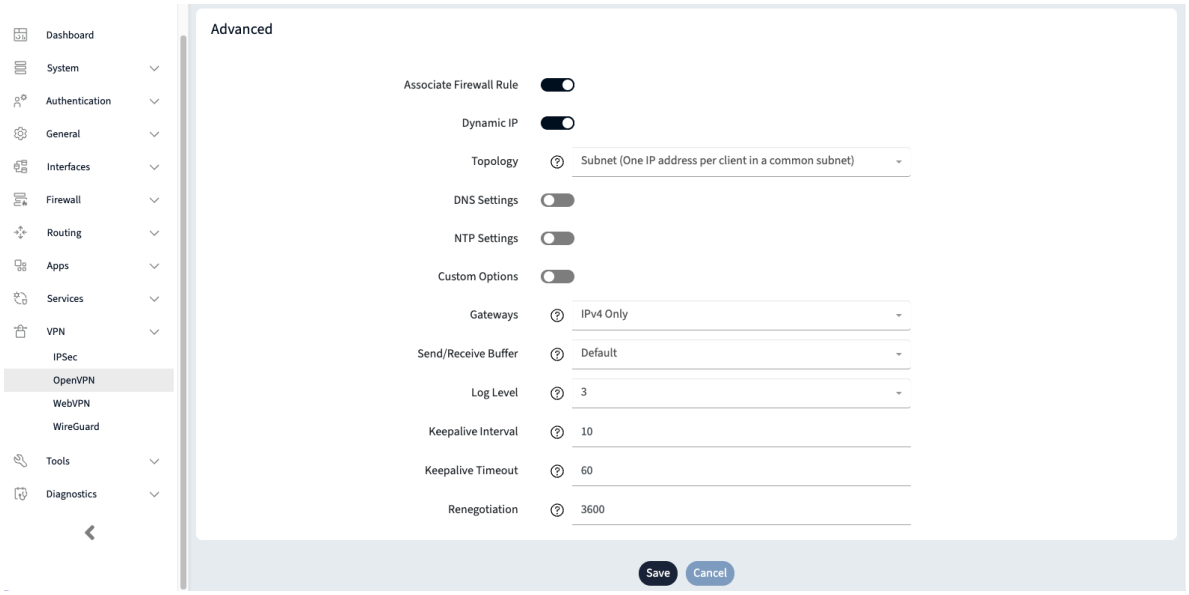


Figure 253: OpenVPN server advanced

12. To create the OpenVPN server, select **Save**.

13. Set the client device(s).

On the client device(s), you can navigate to **VPN > OpenVPN > Router Clients** and select **Add** to manually enter the server's settings.

14. Export the server settings for client configuration.

    To export the settings on the server device, navigate to **VPN > OpenVPN > Servers**. Select the **Edit** button of the server you want to connect the client to and scroll to the bottom. Under **Export**, you see **Hostname/IP address**, where you enter the address of the OpenVPN server and the port you defined for the OpenVPN server. Select **Export** and save the file. Extract the downloaded zip file.

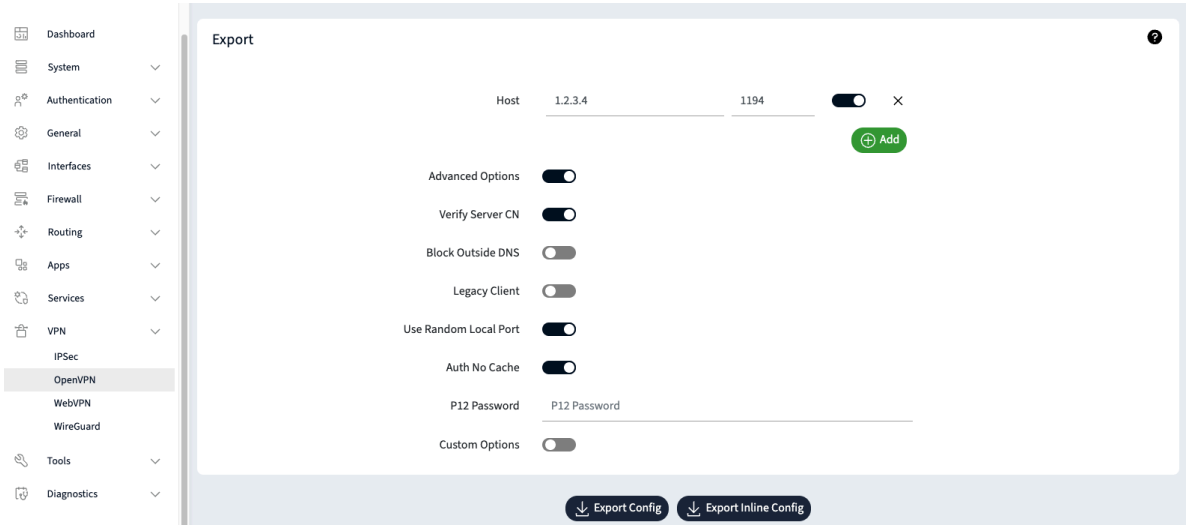


Figure 254: OpenVPN server export

You have successfully created an OpenVPN site-to-site connection. The server is now ready to accept client connections with the exported configuration files.

If you select **Advanced Options**, then TBF provides the following additional options:

Verify Server CN

This setting verifies the server certificate common name (CN) when the client connects.

Block Outside DNS

This setting helps prevent access to DNS servers outside the VPN tunnel while connected, which forces clients to use only the VPN-provided DNS servers.

Legacy Client

This setting excludes OpenVPN 2.5-specific settings from the client configuration, which helps provide compatibility with older OpenVPN clients.

Use Random Local Port

This setting uses a random local source port for client traffic. Without this setting, multiple clients from the same network do not run simultaneously.

Auth No Cache

This setting makes the system immediately forget the username and password credentials after the system uses the credentials.

P12 Password

This field specifies the password required to access the P12 certificate file.

Custom Options

This field specifies custom OpenVPN parameters for the configuration.

OpenVPN firewall rules

Firewall rules are required on OpenVPN interfaces to allow network traffic from VPN clients.

Note: You need firewall rules on the new `OpenVPN_XXX` interface to allow network traffic from the clients. See the *firewall rules (forward and input)* on page 131 for instructions on how to create firewall rules for an interface.

Export OpenVPN client configurations

Export OpenVPN client configurations in standard or inline format after creating and saving an OpenVPN server.

After you create and save an OpenVPN server, export OpenVPN client configurations. In addition to the standard export, also export the configuration in an inline format. Find this functionality at the bottom of the page when you edit an OpenVPN server.

Perform the following steps:

1. Navigate to the OpenVPN server configuration page.
2. Scroll to the bottom of the page to find the export functionality.
3. Set the available *export configuration options* on page 350 as needed.
4. Select the export format based on the authentication mode.

   When the authentication mode is **Certificate**, export the client configuration for each certificate. These exports are available in either **Inline** or **P12** format.
5. Select the appropriate export button to generate the client configuration.

TBF generates the OpenVPN client configuration file in the selected format.

For each certificate, set up an *OpenVPN override* on page 351 to customize individual client settings.

OpenVPN export settings

This topic references settings that are available when exporting OpenVPN client configurations.

Several settings are available when exporting OpenVPN client configurations:

Host
: This setting sets up multiple hosts with their hostname or IP address and port.

Advanced Options
: This setting displays more specific configuration settings.

Verify Server CN
: This setting verifies the server certificate common name (CN) when the client connects.

Block Outside DNS
: This setting blocks access to DNS servers outside the OpenVPN connection and forces clients to use only VPN DNS servers.

Legacy Client
: This setting excludes OpenVPN 2.5 settings from the client configuration for compatibility with former clients.

Use Random Local Port
: This setting enables a random local source port (lport) for client traffic. Multiple clients require this option to run simultaneously.

P12 Password
: This field specifies the password to encrypt the configuration in the P12 format.

Custom Options

This field adds custom configuration parameters to the client configuration.

OpenVPN override

The OpenVPN override feature sets up individual overrides for each certificate and user, as well as create custom overrides to meet specific requirements.

The OpenVPN override feature is available with the **Certificate** or **User** authentication modes. After creating and saving an OpenVPN server, you can access this functionality at the bottom of the settings screen. You can set up individual overrides for each certificate and user, as well as create custom overrides to meet specific requirements.

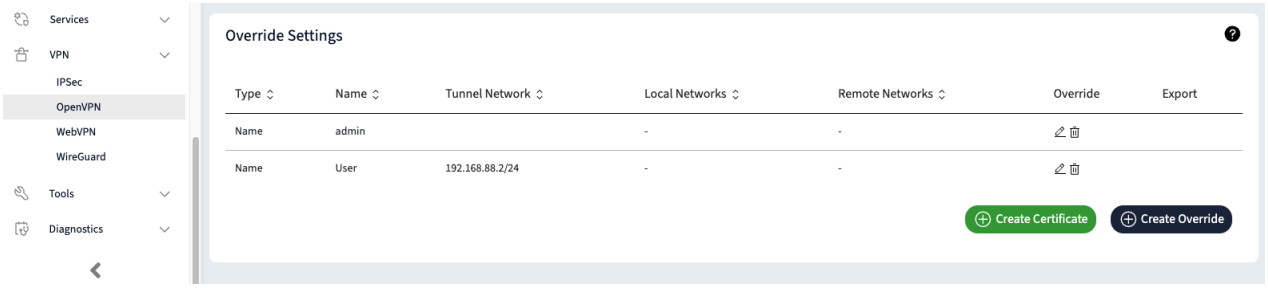


Figure 255: OpenVPN server override

Related information

OpenVPN override settings on page 351
Set up multiple clients with separate networks on page 351

OpenVPN override settings

This topic covers settings that are available for OpenVPN server overrides.

Enabled

This setting enables or disables this override.

IPv4 Tunnel Network

This setting defines the IPv4 tunnel network for the connection.

IPv6 Tunnel Network

This setting defines the IPv6 tunnel network for the connection.

Block Connection

This setting blocks this client connection based on its certificate common name.

Redirect Gateway

This setting forces client-generated traffic through the tunnel.

Local Network(s)

This setting uses multiple local networks.

Remote Network(s)

This setting uses multiple remote networks.

Custom Options

This field defines custom parameters for the configuration.

Set up multiple clients with separate networks

This task describes how to set up OpenVPN overrides to allow multiple clients to access their own separate networks.

To set up an OpenVPN configuration where multiple clients each access their own separate network, use the OpenVPN override feature.

Perform the following steps:

1. Set up the available networks in the OpenVPN server settings under **Remote Network(s)**.
2. For each client, to set their specific network under **Remote Network(s)**, use the override.

WireGuard

WireGuard implements virtual private network (VPN) techniques to assist in establishing security for point-to-point or site-to-site connections.

WireGuard uses cryptographic algorithms to help protect data during transmission. The protocol incorporates several advanced cryptographic components:

- Noise Protocol Framework
- Curve25519
- ChaCha20
- Poly1305
- BLAKE2
- SipHash24
- HKDF

Security experts have carefully selected and reviewed these cryptographic choices to help provide robust assistance with protection.

WireGuard offers several characteristics over traditional VPN solutions like IPsec and OpenVPN. WireGuard delivers high performance, consumes less power, and offers a simpler setup process, making it a suitable choice for both beginners and experienced users.

WireGuard settings

WireGuard VPN settings include connection, security keys, network addressing, and routing options. These settings assist in establishing security for point-to-point VPN tunnels with customizable network and firewall integration.

Find the WireGuard settings under **VPN > WireGuard**.

Port

Each WireGuard connection requires a unique port number for communication.

Private Key

The private key for TBF.

Important: Keep this key secure and confidential—never share it with other parties.

Public Key

The public key for TBF. Share this key with the remote peer to establish the connection.

Tunnel IP

The local IP address for the WireGuard connection. Set the netmask to cover the entire network. This address can be IPv4, IPv6, or both.

DNS Server

The IP address of a DNS server for name resolution within the tunnel.

MTU

The maximum transmission unit size. The default is `1420`. For PPPoE connections, use `1412` to help avoid fragmentation issues.

Associate Firewall Rule

Enable this setting to automatically create and maintain a firewall rule for the WireGuard connection.

Routing Table

Specify which routing table to use. The default `Main` routing table receives routes to peer remote networks automatically.

Master Only

Start WireGuard only when TBF operates as the network master in a high availability (HA) configuration.

Additional Tunnel IPs

Extra IP addresses assigned to the local server or connected peers within the tunnel.

You can export the configuration settings as an Excel spreadsheet using the export option located in the upper-right corner of the interface.

WireGuard export settings

Basic settings for exporting WireGuard peer configurations to client devices.

The export settings define how the system generates peer configuration files and what network access they help provide.

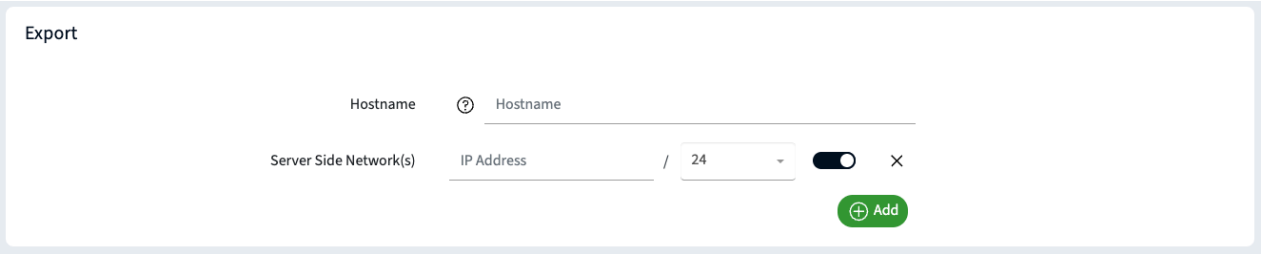


Figure 256: WireGuard export settings

In the **Hostname** field, specify how peers connect to the WireGuard server from the internet. Use either a domain name like `vpn.company.com` or a public IP address. This address tells the peer configuration where to find the server.

The **Server Side Network(s)** field defines which networks the connecting peer accesses through the VPN tunnel. Specify the internal networks behind the WireGuard server that remote users need to reach, such as the office LAN or specific subnets.

Note: When you include the **IPv4 Tunnel Network** in the **Server Side Network(s)** list, peers communicate with each other through the server. TBF automatically adds these local networks to the routing configuration of the peer.

WireGuard peer settings

WireGuard peer settings define the parameters needed to assist in establishing security for connections with remote devices in the VPN network.

A peer represents any remote device in the WireGuard network, whether it is another site or an individual client. WireGuard treats remote endpoints equally as peers, so there is no functional difference between different peer types.

Keep Alive

Sends periodic keep-alive packets every few seconds to maintain an active connection and help prevent timeouts. This helps provide that the tunnel stays open even when no data is being transmitted.

Private Key

The private key of the peer for encryption. While the private key is not required for the connection itself, you can generate a key pair here if the peer device cannot create its own keys. When generated, TBF displays the private key in this field.

Public Key

The public key of the peer for encryption. This key is required to establish the connection and needs to be obtained from the peer device.

Preshared Key

An optional additional layer of assistance in establishing security that helps provide extra authentication beyond the standard key exchange. This shared secret enhances assistance with protection against potential quantum computing attacks.

Tunnel IP

The IP address assigned to the peer within the WireGuard tunnel. This address is IPv4, IPv6, or both, and falls within the IP address range of the tunnel defined in the main settings.

Endpoint Address

The external address of the peer for establishing the connection. This is either a static IP address or a domain name that resolves to the location of the peer.

Endpoint Port

The UDP port number that the peer listens on for WireGuard connections. This needs to match the port set on the peer device.

Remote Networks

Network subnets that are accessible through this peer. Traffic destined for these networks routes through the WireGuard tunnel to reach resources behind the peer.

Additional Tunnel IPs

Extra IP addresses that you assign within the tunnel, either for the local server or for peer devices. These help provide additional addressing options within the VPN network.

WireGuard peer export

Export peer configurations for WireGuard clients.

After you create a peer, you can export its configuration for use with WireGuard clients. To access the export function, navigate to the WireGuard edit page and use the quick edit panel located at the bottom of the page.

WireGuard QR code

Generate QR codes for easy WireGuard peer configuration.

After you create a WireGuard peer, TBF automatically generates a QR code that contains the necessary configuration details for that peer. This QR code is a convenient way to quickly set up the peer connection on mobile devices or other WireGuard clients.

To access the QR code, navigate to the WireGuard edit page and locate the quick edit panel at the bottom of the page. The QR code appears in this panel for the corresponding peer you created.

WireGuard import configuration

To avoid creating new configurations, import existing WireGuard configurations.

To save time, import settings from an existing WireGuard configuration rather than building a new configuration. To import a configuration, locate the **Import Config** button in the lower-right corner of the WireGuard management page. When you import a configuration, TBF automatically creates the necessary data structure and transfers the settings from the configuration file.

Example WireGuard configuration

This topic provides a step-by-step example of how to set up a WireGuard VPN connection.

This example demonstrates how to create a complete WireGuard configuration with server and peer settings. WireGuard is a modern VPN protocol that assists in establishing security for connections between devices.

Perform the following steps:

1. Navigate to **VPN > WireGuard**. Select **Add**.

   The address and address range you define under **Tunnel Network** are the IP addresses that the system assigns to VPN clients. Specify an address range that does not overlap with the existing local network. For a single client connection, select a single IP address with a /30 subnet mask.

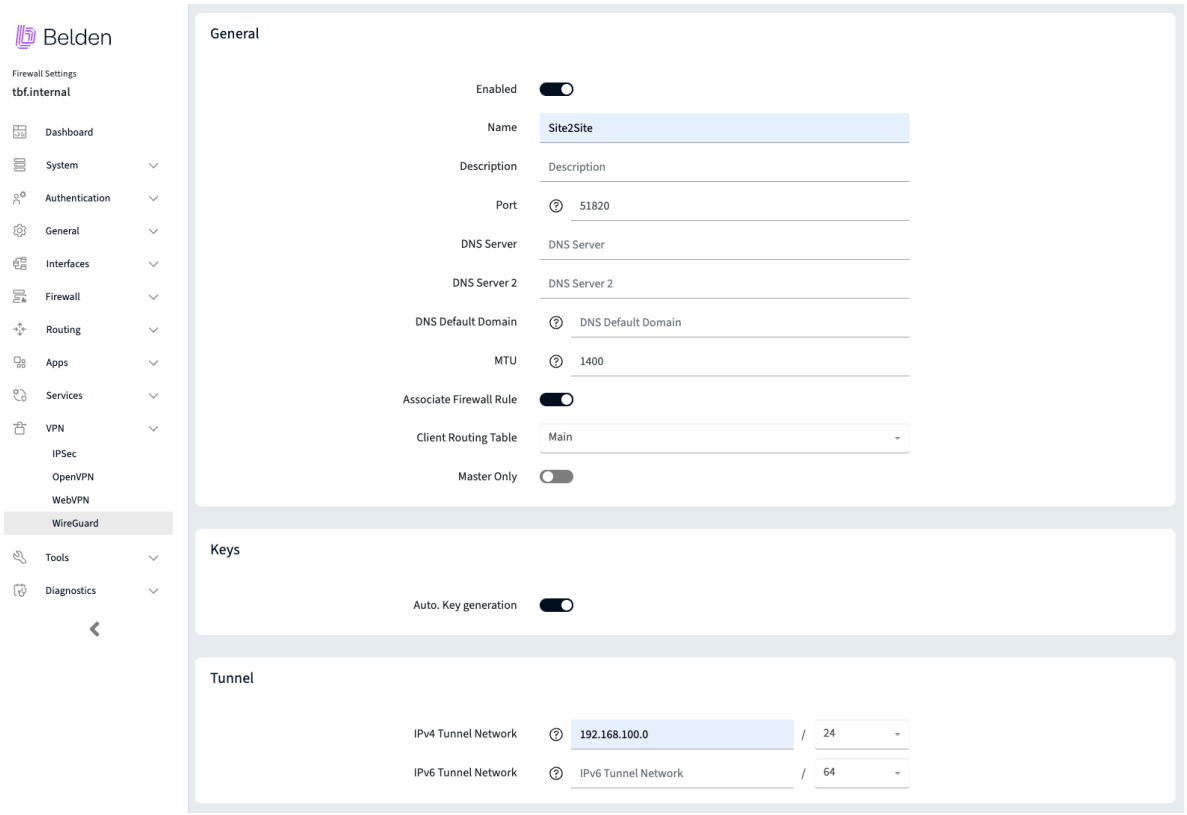


   Figure 257: WireGuard server creation

2. To create a new peer, select **Add**.

   When you create peers (client devices), remember that WireGuard limits you to the number of peers you manually set up. This limit is because the system assigns each peer a unique static tunnel IP address. Additionally, the same peer cannot connect multiple times simultaneously.

3. Enter a unique **Tunnel IP address** that falls within the range you specified in the server configuration.

   Verify that no other peer uses this IP address. If needed, then specify additional tunnel IP addresses under **Additional Tunnel IPs**.

4. If you want assistance in establishing security, then enable **Auto PSK Generation**.

   Auto PSK Generation automatically creates a pre-shared key (PSK) to assist in establishing security for peer authentication. While the PSK is optional, the PSK helps provide an additional layer of assistance in establishing security beyond the standard WireGuard encryption.

5. If you select **Endpoint Type Static**, then enter the server public IP address and the WireGuard configuration port number.

   Under **Remote Networks**, specify which local networks the peer makes available to the server. This setting allows the server to access networks behind the peer device. If you do not need to access any networks behind the peer, then leave **Remote Networks** empty.

6. If required, then set up **Local Network(s)** in the peer configuration.

To define specific networks that only this peer can access, use **Local Network(s)** in the peer configuration. These networks are additional to the global **Local Network(s)** settings.

7. To save the peer configuration, select **Save**.

   After you create all required peers, to save the complete configuration, select **Save** again.

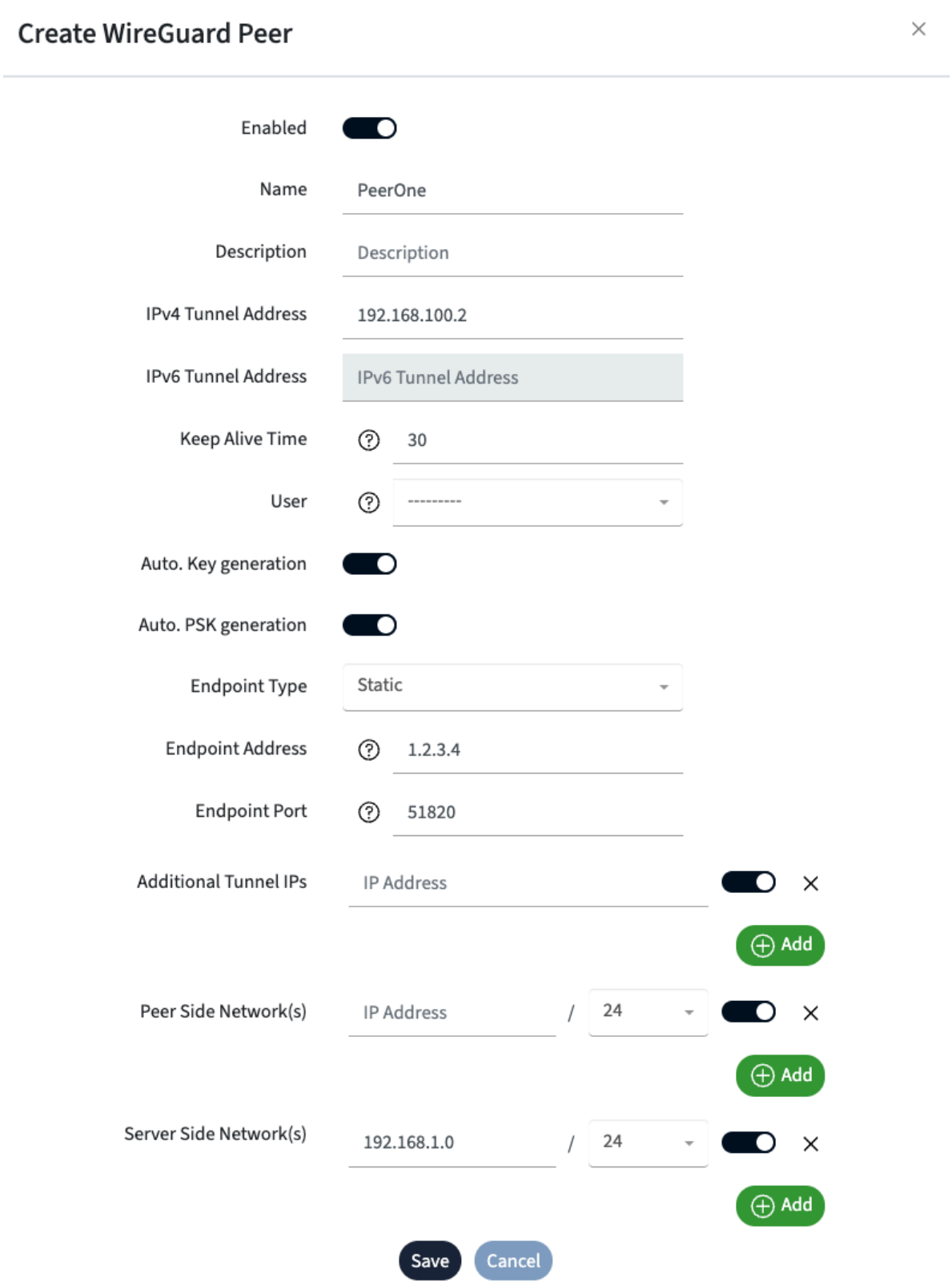


Figure 258: WireGuard peer creation

8. To export the peer configuration, use the **Export** panel located above the **Save** button.

To create multiple peers simultaneously, select **Add Multiple**. Enter only the peer name—if you do not specify IP addresses, then the system automatically assigns the IP addresses.

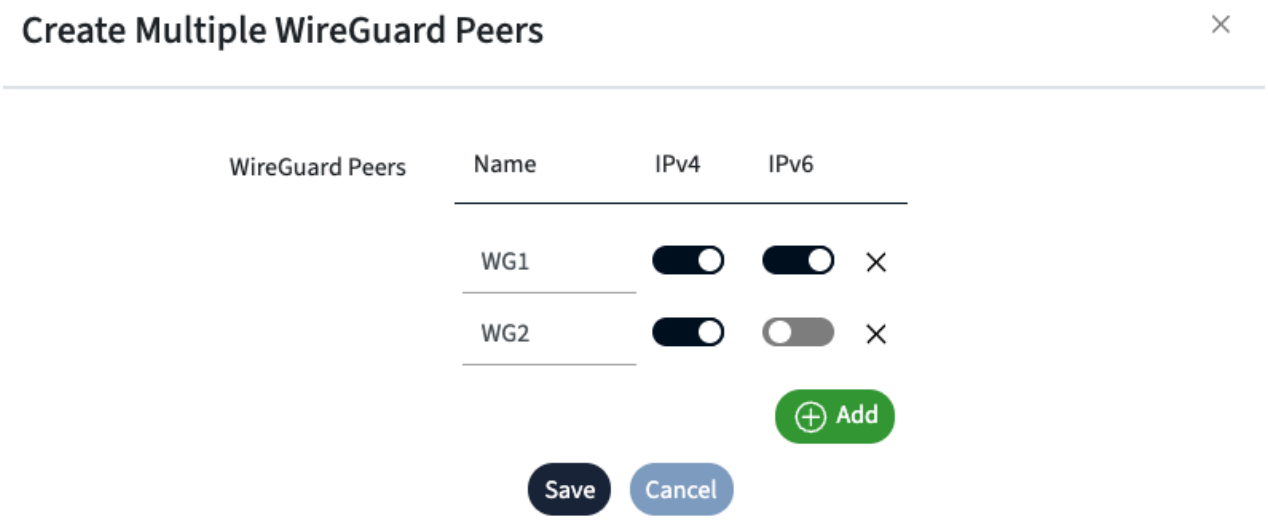


Figure 259: WireGuard multiple peers

Note: Create firewall rules on the new `Wireguard_XXX` interface to allow network traffic from the peers. For instructions, see the *firewall rules (forward and input)* on page 131.

WebVPN

WebVPN offers clientless web-based remote desktop access through Apache Guacamole.

WebVPN allows you to access remote systems directly through a web browser without installing additional client software on the local machine.

Find the WebVPN settings under **VPN > WebVPN**.

Before you use WebVPN, you install it first. To install the WebVPN service, navigate to **System > Addons**.

WebVPN configuration and usage

To set up WebVPN, set up several interconnected components:

- *WebVPN settings* on page 358: Set up the main WebVPN service parameters and global options
- *WebVPN server settings* on page 359: Define server connections and specify supported protocols
- *WebVPN user settings* on page 361: Associate individual users with specific servers they access
- *WebVPN group settings* on page 361: Grant server access permissions to entire user groups
- *Use WebVPN* on page 362: Connect to and interact with remote systems through a browser

WebVPN settings

This topic describes the settings for the WebVPN service.

Enabled

This setting enables or disables the WebVPN service. The service is disabled by default for reasons that assist in establishing security.

Hostname

This field sets the hostname that users connect to for WebVPN access. The system uses TBF as the default hostname.

Web HTTPS Port

This field defines the port to assist in establishing security for WebVPN connections. TBF uses port `443` by default, which is the standard HTTPS port.

Certificate

This setting selects the SSL certificate that assists in establishing security for WebVPN page connections. This certificate helps provide encrypted communication between users and the WebVPN service.

2FA

This setting enables two-factor authentication to assist in establishing enhanced security. This feature only applies to users who have set up a TOTP (Time-based One-Time Password) token in TBF. Users without a TOTP token still access WebVPN using only their regular credentials.

Logo

This setting uploads a custom logo that appears on the WebVPN login page. The logo file requires the PNG format to display correctly.

Export these settings as an Excel spreadsheet by selecting the export option in the upper-right corner of the interface.

After you set up up WebVPN, create a firewall rule that allows incoming traffic to TBF on the specified port. The web server monitors the hostname for connection requests. WebVPN operates over HTTPS protocol and helps provide a web-based interface that assists in establishing security for connecting to remote servers.

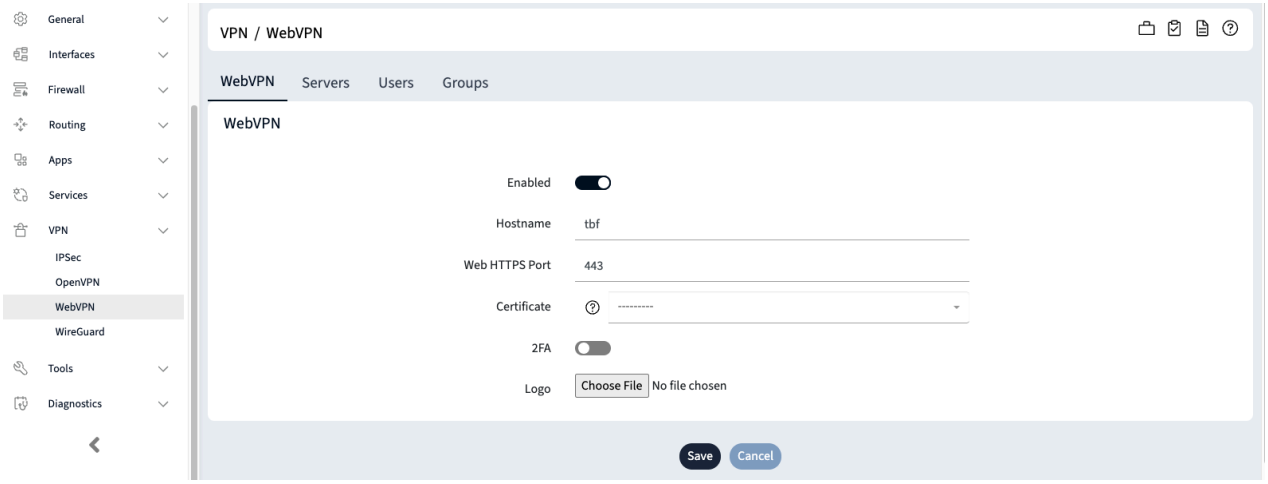


Figure 260: Webvpn settings

WebVPN server settings

This topic describes the settings for WebVPN server connections.

To enable server connections through WebVPN, create server configurations and assign them to users. Each configuration defines how WebVPN connects to a specific server in the network.

Type

This setting determines the connection protocol for the server. Select from:

- **RDP**: Remote Desktop Protocol. This is the default.
- **SSH**: Secure Shell
- **Telnet**
- **VNC**: Virtual Network Computing

Name

This field specifies a unique name to identify this server configuration. This field is required and differs from other server configurations.

Description

This field adds an optional description to help identify the purpose or location of this server.

Hostname

This field specifies the IP address or hostname of the target server. This server is accessible from the WebVPN system. This field is required.

Port

This field determines the port. TBF automatically sets the default port based on the connection type:

- `3389` for RDP
- `5900` for VNC
- `22` for SSH
- `23` for Telnet

Change this value when the server uses a non-standard port.

Username

This field specifies the username for server authentication. This field is optional. If you leave it blank, then TBF prompts users to enter credentials when they connect.

Password

This field specifies the password that corresponds to the username. This field is optional. If you leave it blank, then TBF prompts users to enter their password when they connect.

Domain

This field specifies the Windows domain name for authentication. This field only applies to RDP connections and is optional.

Security

This setting determines the security protocol for the connection. Select from:

- **Any**: Allows the server to decide. This is the default, which helps provide the most compatibility.
- **NLA**: Network Level Authentication
- **Extended NLA**
- **TLS**: Transport Layer Security
- **VM Connect**
- **RDP**: Standard RDP security

Keyboard Layout

This setting determines the keyboard layout for the remote session. The default is **English (US)**. This setting only applies to RDP connections and helps provide proper key mapping during remote sessions.

Keepalive Interval

This field sets how often in seconds the client sends keepalive packets to maintain the connection. The default is `0`, which disables keepalive packets. Use a minimum value of `2` seconds to help prevent connection timeouts.

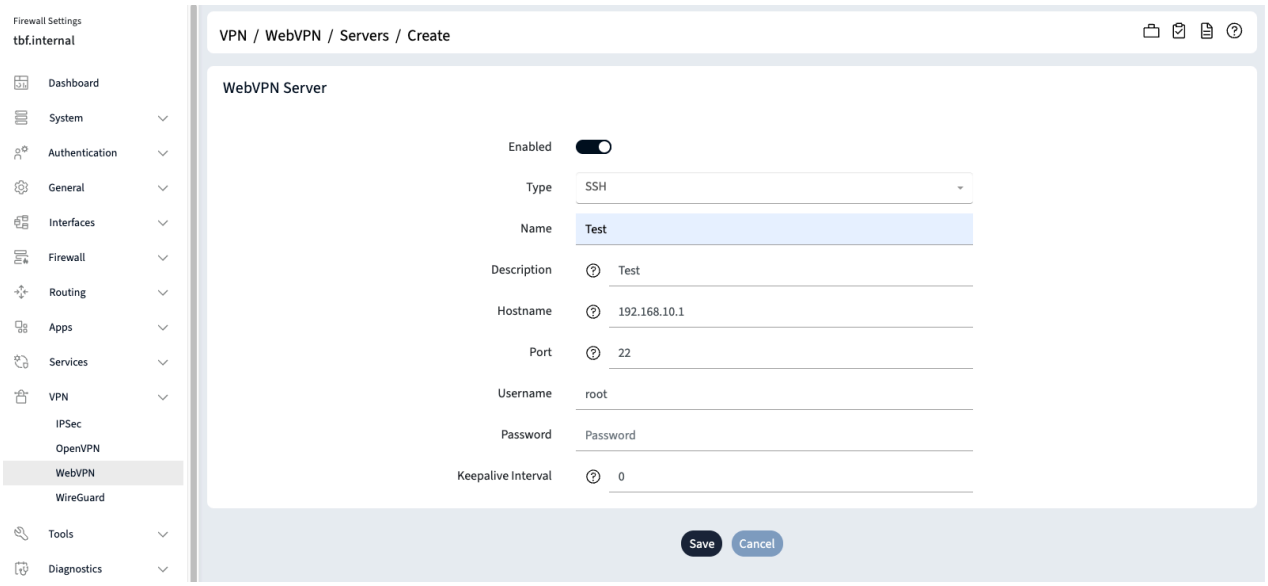


Figure 261: WebVPN server

WebVPN user settings

Settings for linking individual users to specific WebVPN servers.

WebVPN user settings connect individual TBF users to specific WebVPN servers. Assign each user access to multiple servers. This configuration helps provide proper user authentication and controls which resources users can access.

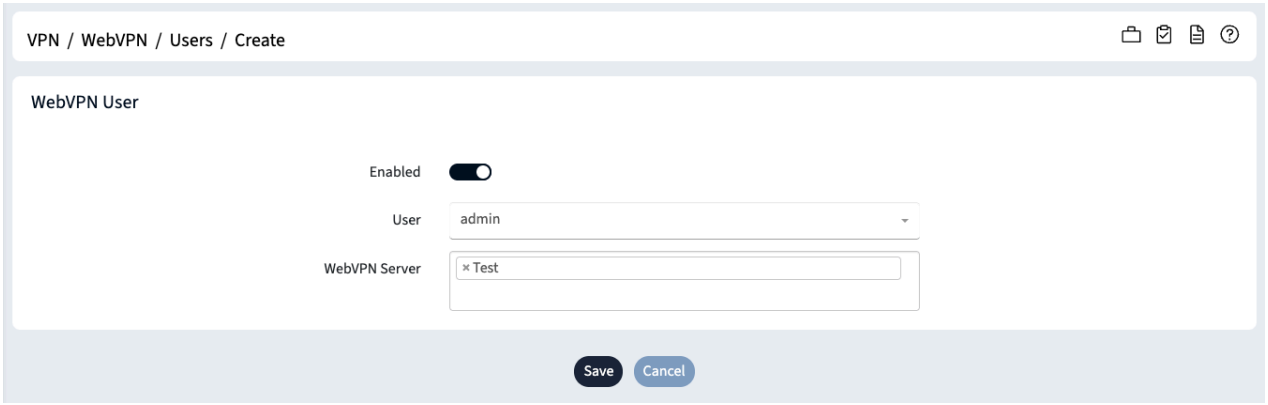


Figure 262: Webvpn user

WebVPN group settings

This topic describes how to set up server access for entire user groups rather than for individual users.

WebVPN group settings grant server access to complete user groups within TBF. This approach streamlines access management because you set permissions at the group level. Each group can access multiple servers based on the configuration.

Note: When a user has both *individual* and *group* server permissions, TBF merges the access rights. The user receives the combined permissions.

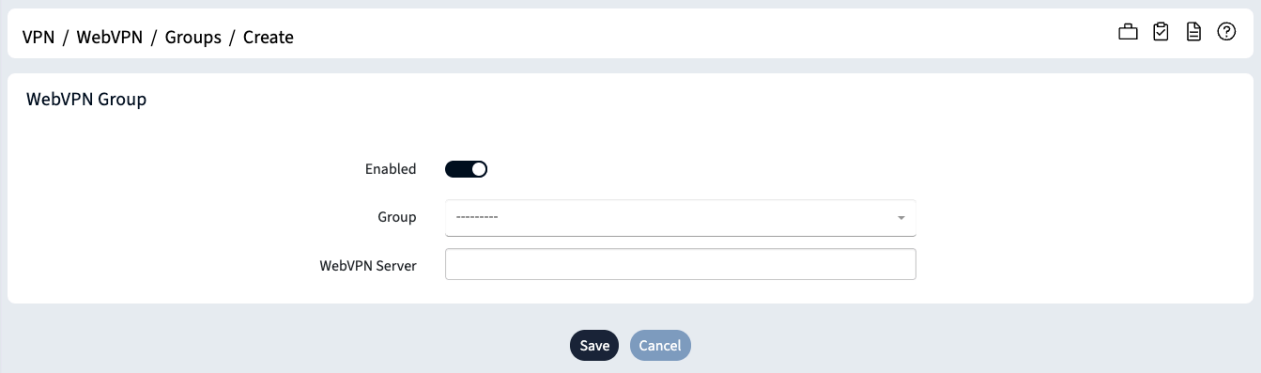


Figure 263: Webvpn group

Use WebVPN

Access remote systems through WebVPN using a web browser.

Note: This section contains screenshots from the former VT Air GUI. These will be updated in a future version.

WebVPN allows you to connect to remote systems directly through a web browser without installing additional software. Before you can access WebVPN, you need valid login credentials. These credentials can belong to either a local TBF user account or a user account from an LDAP authentication server that the system administrator has set up.

Perform the following steps:

1. Open a web browser and navigate to the WebVPN login page.

   The WebVPN login page appears and displays fields for the username and password.

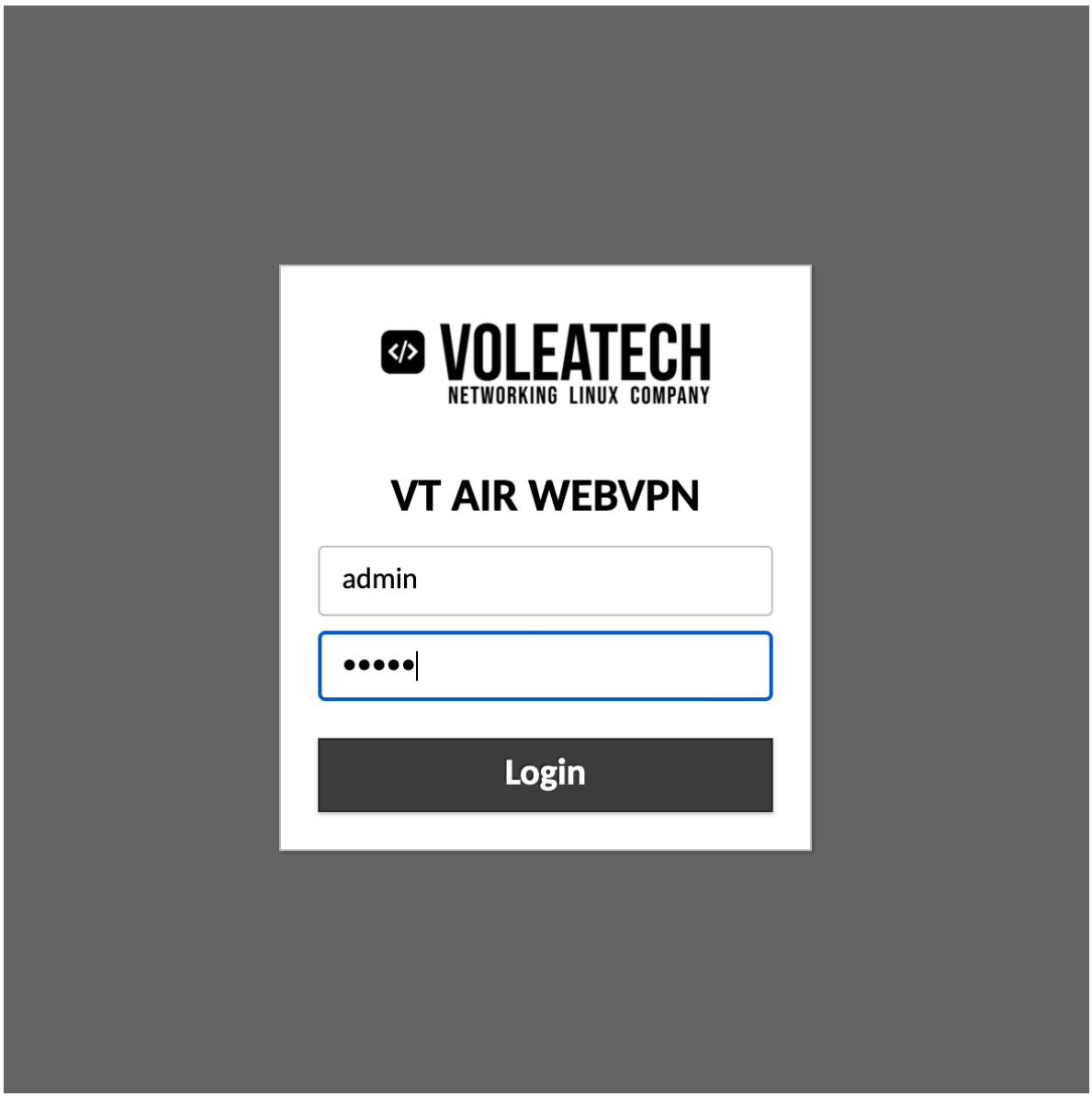


   Figure 264: Webvpn login

2. Enter the username and password in the appropriate fields. Select **Login**.

   After you authenticate successfully, the system displays a list of servers that are available to the user account. The servers you see depend on the permissions assigned to the user account.

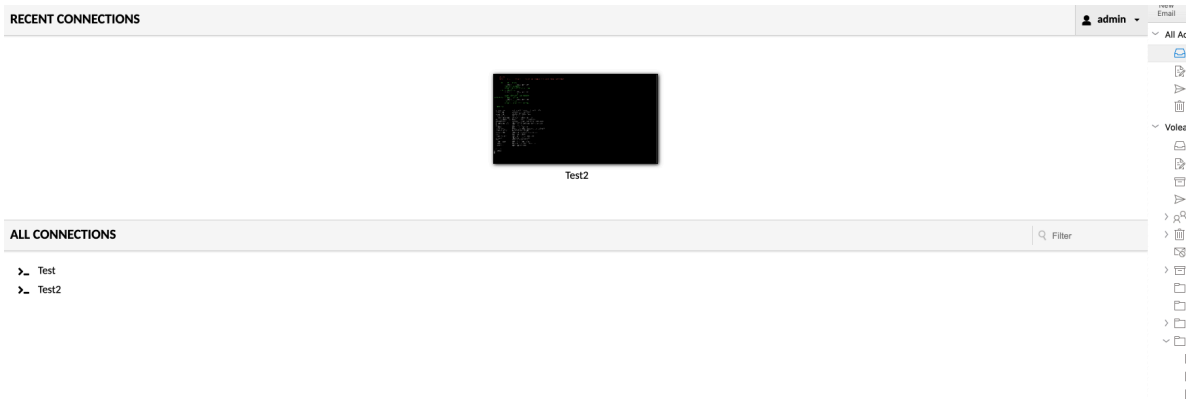


Figure 265: Webvpn GUI server

3. Select the server from the list to connect to a server.

   The browser establishes a secure connection to the selected server. Once connected, you can log in to the remote server and perform tasks as if you were working directly on that system. Interactions happen through the web browser interface. This interface eliminates the need for separate SSH clients or remote desktop applications.

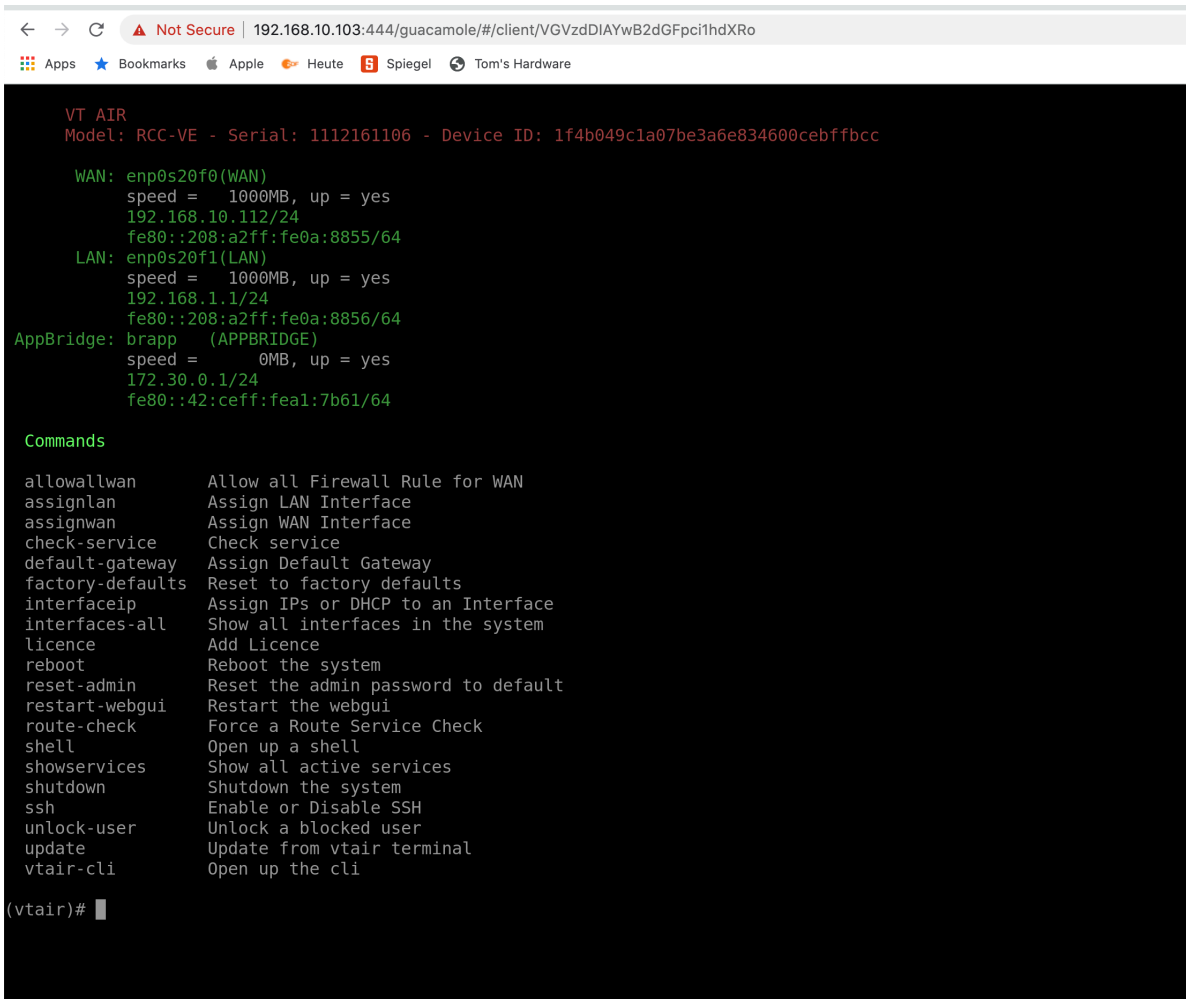


Figure 266: Webvpn SSH

18. Apps

Apps offer Docker-based containerization for running applications on the device with persistent storage and streamlined management capabilities.

Find the apps under **System > Apps**.

Apps use Docker containerization technology to run applications in isolated environments. This approach offers assistance in establishing security, resource management, and easy deployment of third-party software without affecting the core system.

Building blocks

3 main building blocks work together:

Images

Serve as blueprints for containers. Images contain either an operating system, a complete application, or both. Upload custom images or download them from the Docker registry.

Volumes

Offer persistent storage for applications. Volumes keep data available even when containers are stopped or recreated.

Containers

Represent the actual running applications. Containers use images as their foundation and can mount volumes for data persistence.

Typical workflow

To deploy an application using apps:

1. Load or download an image that contains the application.
2. Create volumes if the application requires persistent storage.
3. Create and set up a container based on the image.
4. Start the container to run the application.

Templates provide a streamlined alternative that combines these steps into a single process for common applications.

Management areas

Apps include several management areas:

- *Settings* on page 365: Set up Docker framework and network settings
- *Images* on page 367: Manage application blueprints and access the Docker registry
- *Volumes* on page 367: Create and manage persistent storage
- *Containers* on page 367: Monitor and control running applications
- *Run a new application* on page 369: Step-by-step deployment process
- *Templates* on page 370: Streamlined container creation for common applications
- *Examples* on page 371: Practical implementation guides for specific applications

Settings

This concept describes how to set app settings, understand the Docker framework, and manage networking options.

Find the app settings under **System > Apps > Settings**.

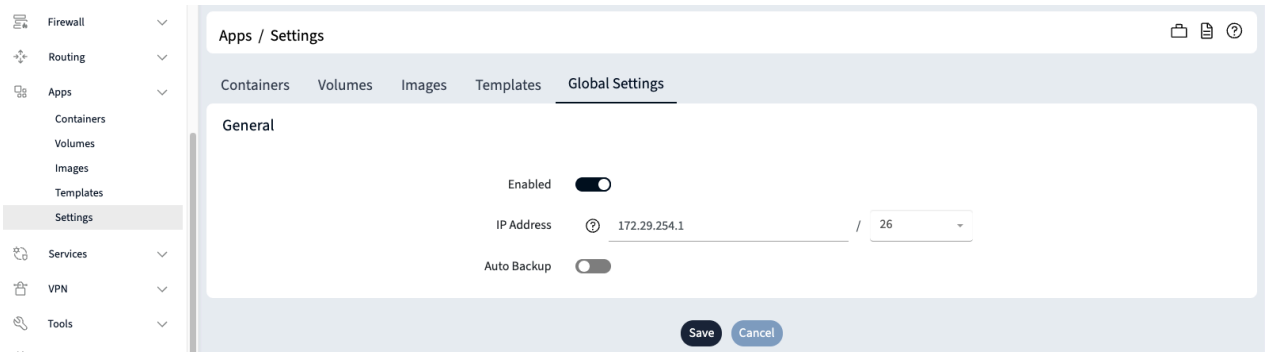


Figure 267: General app settings

This section contains:

- General app configuration options
- Docker framework overview and concepts
- Network bridge specifications and settings

Related information
Set app settings on page 366
Apps overview on page 366
Networking on page 367

Set app settings

Set Docker bridge settings and backup options for the apps system.

In this section, modify the default Docker bridge settings. Select an unused network that is compatible with the existing setup.

Note: While this network is not directly relevant for app functionality, TBF requires this network for apps to operate properly.

Perform the following steps:

1. Navigate to the app settings section.
2. Select a network configuration that is compatible with the existing setup.
3. Set up the **Auto Backup** feature if needed.

   Auto Backup

   This setting automatically creates nightly backups of Docker containers and volumes. TBF retains the last 3 backups in `/opt/docker_backups`. Consider the storage space requirements and verify that the hard drive has sufficient capacity for this feature. You can also perform manual backups using the `docker-backup.sh` command in the shell.

TBF applies the new Docker bridge settings and backup configuration.

Apps overview

This page provides an overview of the Docker-based app container framework.

TBF uses Docker as an app container framework. Apps consist of different building blocks:

- Images
- Volumes
- Containers

Images serve as blueprints for containers. Each container uses a copy of the image to run. An image contains an operating system like Debian or a program like Java.

Volumes are folders or files that you mount inside a container. Without volumes, the system deletes any data when you remove the container.

Containers provide the running environment for apps. Each container loads an image and mounts the selected volumes in a specified location. You can also define hardware limitations, mount devices, and enable networking. The container runs in a sandboxed environment and cannot access the host system data.

Networking

This page describes the Docker networking setup and bridge configuration for apps.

Because of the way Docker sets up the system, the system includes 2 bridges:

- `docker0`
- `brapp`

`docker0` is the default Docker bridge. TBF does **NOT** use the bridge. However, the system requires this bridge.

`brapp` is the network bridge that containers connect to.

Images

Docker images are the foundation for running applications in containers. Obtain images by uploading the images or by searching the Docker registry.

Find application images under **System > Apps > Images**.

You have 2 options for obtaining images:

- Upload an image
- Search for an existing image in the Docker registry

The Docker registry has a wide selection of prebuilt images for applications.

To explore available images, visit the *Docker Hub* and browse through the catalog.

If TBF has an active internet connection, to find Docker images that match the requirements, enter a search term. To download the desired image and make the image available for use in a container, select the image.

Volumes

This page describes how to manage application container volumes.

Find the application volumes under **System > Apps > Volumes**.

A volume is a dedicated directory on the TBF hard drive that helps provide persistent storage for applications.

The system automatically creates volumes in the `/var/lib/docker/volumes/NAME/_data` directory on the hard drive.

Containers

Containers represent running applications and offer management capabilities for application lifecycle operations.

Find the application containers under **System > Apps > Containers**.

A container represents the actual running application. The overview page displays the containers in the system and shows their current status. When a container is running, an **Information** icon appears in the first column. To view the ports that the container uses, hover over this icon. For additional container details, hover over and select the **Information** icon in the second column.

While the container is running, you can retrieve the log output from the container or connect to the screen of the container. You can start, stop, restart, delete, or edit a container as needed.

Note: You cannot change container settings after creating the container. Modifying these settings requires deleting and recreating the container.

For information about available settings, see the *container settings* on page 368.

Container settings

These settings are available when you create or edit containers.

Name

This field specifies a name to identify the container.

Image

This setting specifies the image that the container is based on.

Start

This setting determines whether you want the container to start right away after you create the container.

CPUs

This setting limits the container to a specific number of CPUs.

RAM Limit

This setting limits the amount of memory available to the container.

Restart Policy

This setting determines whether the application starts automatically when the system boots or after a detected failure.

Auto Remove

This setting determines whether to automatically delete the container when it stops running.

Hostname

This field sets a custom hostname inside the container.

Interactive

This setting is necessary for containers that require user input and output interaction. For example, to allow terminal access, enable this setting when you run Debian as a container.

Command

This field specifies commands that the container executes at startup. For example, to start a shell for Debian containers, use `/bin/bash`.

Temporary FS

This setting mounts a RAM file system inside the container.

Networking

This setting enables or disables network access for the container.

IP and MAC

This setting sets a custom MAC address as well as IP addresses. If you do not set one, the system allocates one for you. IP addresses are restricted to the bridge application range.

Environment Variables

This field creates environment variables. Each variable consists of a key-value pair.

Volumes

This setting mounts volumes inside the container. Enter the path inside the application or container where to mount it, and set the access permissions to read-only or read/write.

Devices

This option adds a device such as a USB connector to the container.

Run a new application

To create and run a new application, load images, create volumes, set up containers, and establish network connectivity.

To create and run a new application, follow these steps:

Perform the following steps:

1. Load a new image that contains either the operating system the application runs on or the application itself.

   See *images* on page 367 for more details.

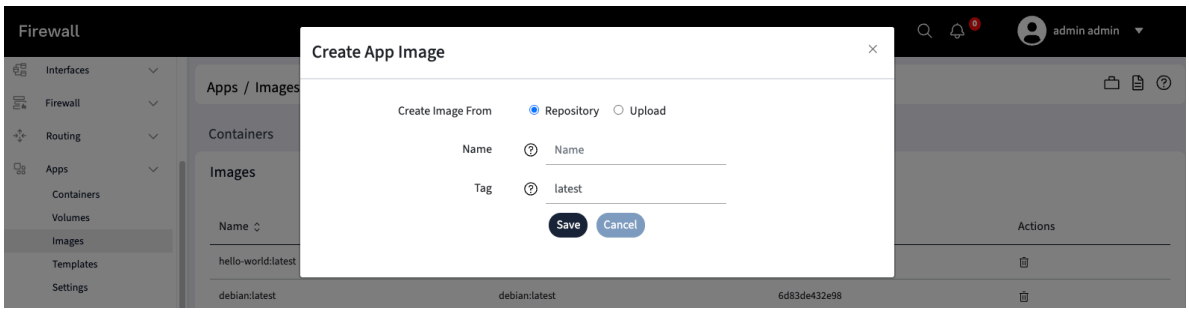


Figure 268: Load new image

2. If the application requires an external folder, create a volume to store the data.

   See *volumes* on page 367 for more details.

   Note: For many images, this step is either not required or the image creates an external folder automatically.

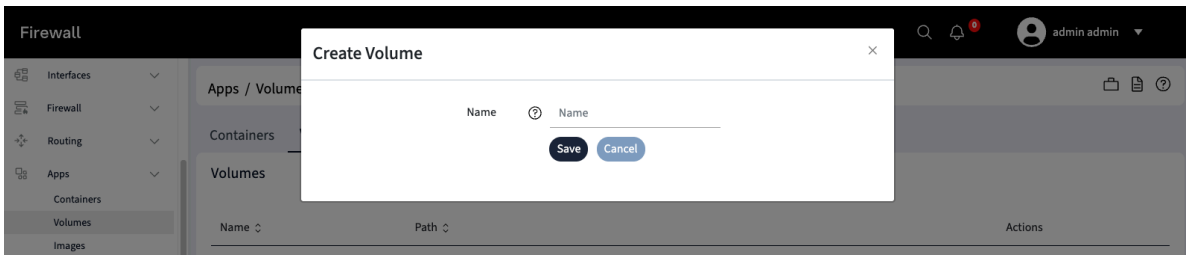


Figure 269: Load new volume

3. Create a new container with the available configuration options.

   See *containers* on page 367 for more details.

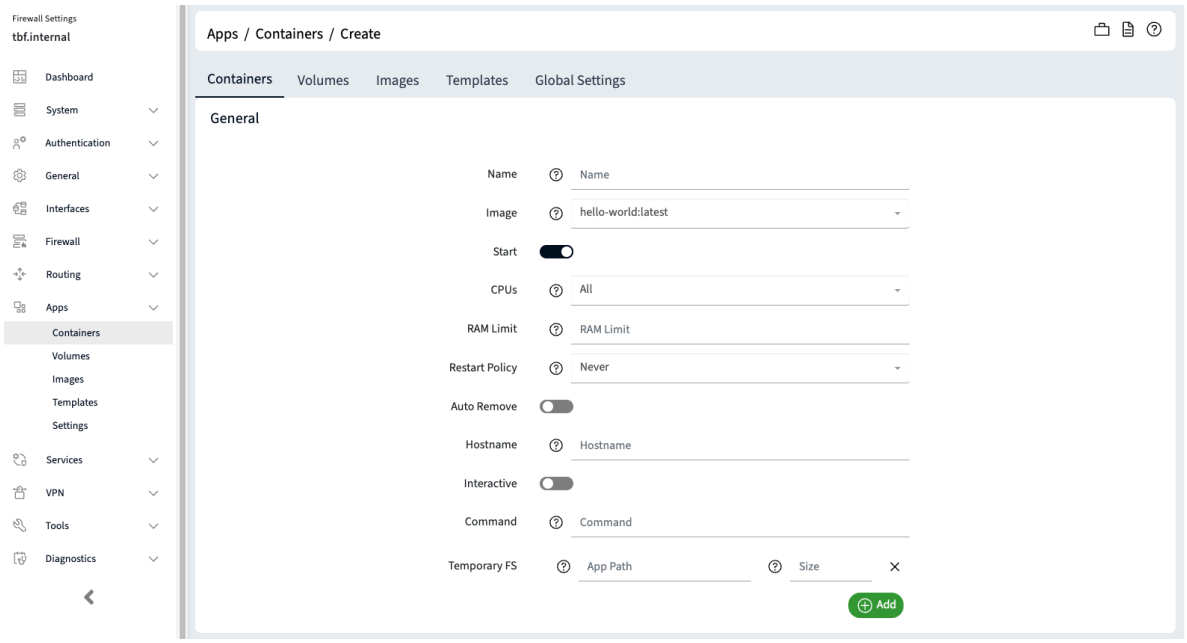


Figure 270: Load new container

4. To enable network connectivity for the container, enable the network connection in its settings.

   If you leave the IP address field empty, TBF automatically assigns an IP address within the application bridge `brapp`.

   Note: The default firewall rule allows access to local and non-local networks on the application bridge. However, TBF blocks incoming traffic by default. For example, the *firewall rules (forward and input)* on page 131 topic provides more details about enabling port forwarding.

5. To access the newly created application, select the **Play** button to run the container.

   You now see new **Stop** and **Reload** options, as well as an option to start the console of the container.

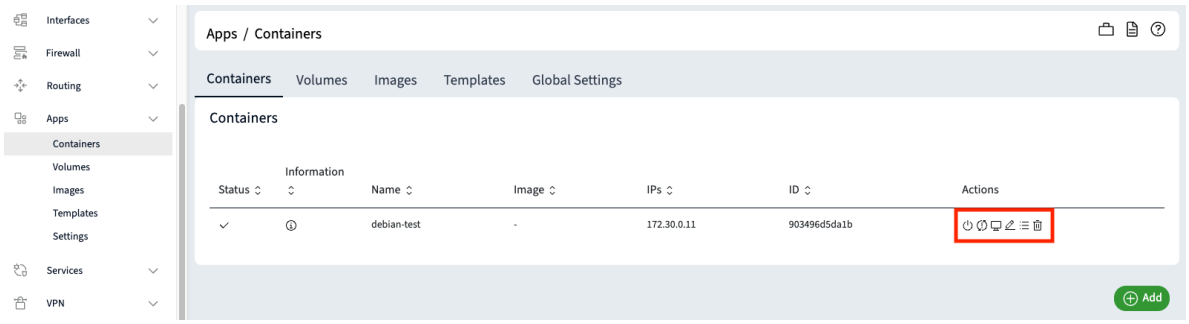


Figure 271: Run new container

Templates

Templates help provide a convenient shortcut for creating containers and downloading required images in one streamlined process.

Find the templates under **System > Apps > Templates**.

Templates help provide a convenient shortcut for creating a container and downloading the required image in one streamlined process.

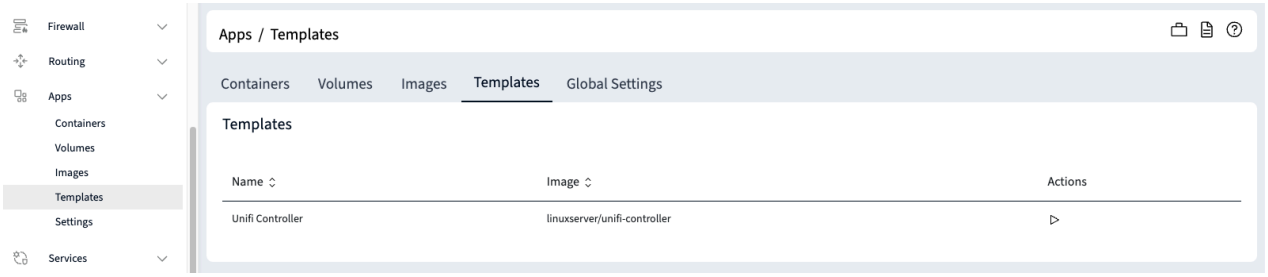


Figure 272: Templates

The overview page displays the available templates. To create and start a container directly from any template, use the **Actions** column.

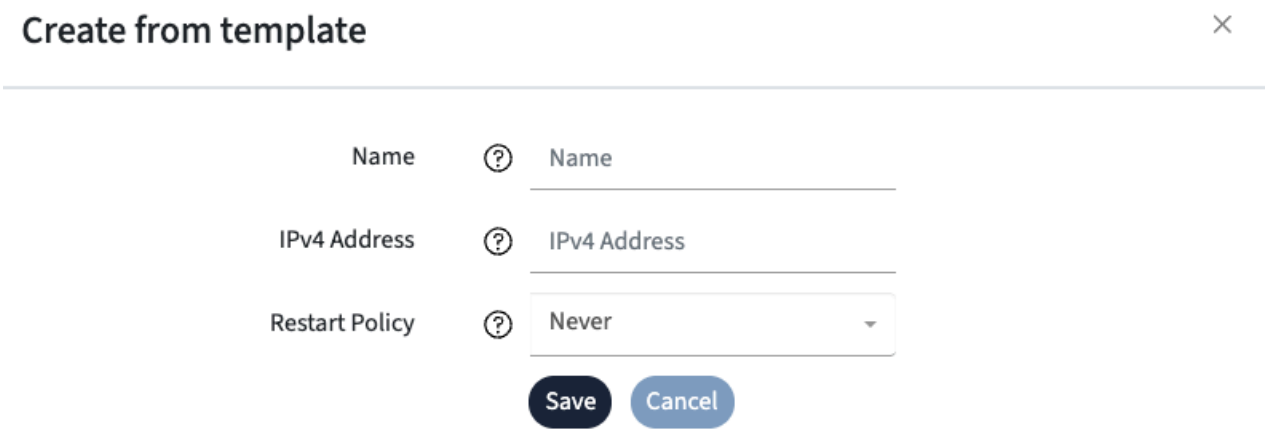


Figure 273: Create from template

Create a container from a template

You can create and start a container directly from a template.

To create a container from a template, define specific settings to set up the container for the environment.

1. Navigate to **System > Apps > Templates**.
2. In the **Actions** column, select the template to use.
3. Define the following settings:
    - **Name**
    - **IPv4 Address**
    - **Restart Policy**

Examples

These are examples of installing and setting up applications using the apps feature.

The following examples demonstrate how to install and set up different types of applications using the apps feature:

- *UniFi Controller* on page 372: Install the UniFi Controller software for network management
- *Debian* on page 372: Install a Debian container for general-purpose computing
- *MySQL* on page 373: Install a MySQL database container for data storage

UniFi Controller

You can install the UniFi Controller software on a device using apps.

To install the UniFi Controller software on the TBF device, use apps. Navigate to **System > Apps > Images**. Select **Add**.

Perform the following steps:

1. Select an available UniFi Controller image, such as the `jacobalberty/unifi` image.
2. Select **Save**. Wait for the system to install the image.

   Note: Creating a volume for the UniFi Controller image is unnecessary because the image automatically creates a volume.

3. Create a container based on the image.

   Leave **Interactive Mode** disabled.

To access the UniFi Controller, navigate to *http://IP-ADDRESS:8080/* or *https://IP-ADDRESS:8443/*, where `IP-ADDRESS` is the address shown in the container overview panel.

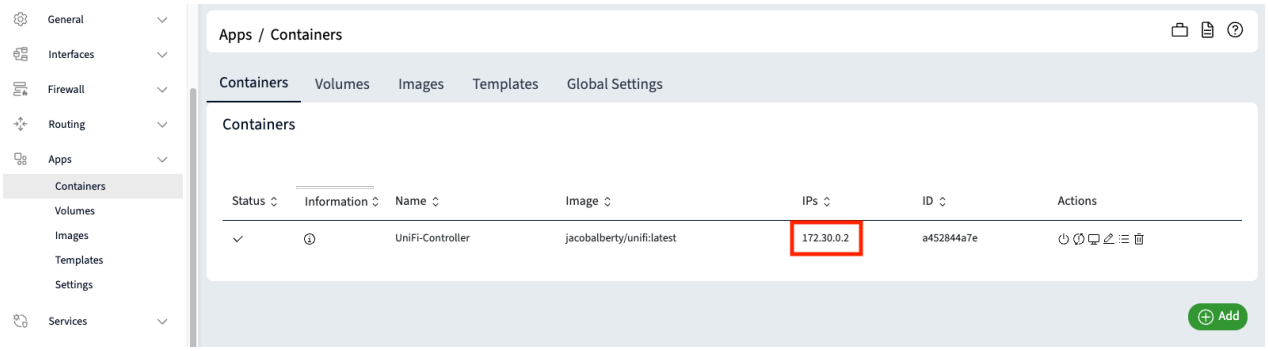


Figure 274: Unifi controller

Debian

Use the apps feature to install the Debian container on a device.

To install the Debian container on the TBF device, use the apps feature. Navigate to **System > Apps > Images**. Select **Add**.

Perform the following steps:

1. Select the main Debian image from the available options.
2. Select **Save**. Wait for the system to install the image.

   Note: The Debian image does not require a volume.

3. Create a container based on the image.

   Enable **Interactive Mode**. Set the command to `/bin/bash`.

To access the console of the Debian image in interactive mode, select the **Console** icon.

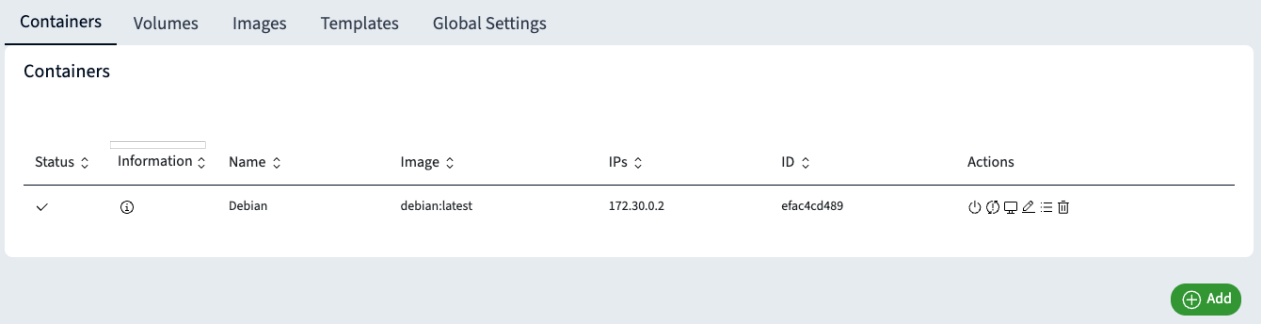


Figure 275: Debian application

MySQL

Use the console to pass along a default user, database, and password to install the MySQL container on a device.

To install the MySQL container on the TBF device, use the console to pass along a default user, database, and password. A MySQL container is useful for storing permanent data from addons such as NtoPNG.

Perform the following steps:

To create a MySQL container with the user, password, and database `ntopng`, run the following command:

```
docker run --network brapp --name mysql-ntopng -e MYSQL_ROOT_PASSWORD=mysqlroot -e MYSQL_DATABASE=ntopng
 -e MYSQL_USER=ntopng -e MYSQL_PASSWORD=ntopng -d mysql:5.7-debian
```

View and manage the container in the web interface after installation.

19. High availability

High availability (HA) helps ensure continuous system operation. HA provides redundancy and automatic failover between multiple devices.

HA in TBF consists of 3 different and independent components that work together to help provide seamless failover and continuous operation:

- **Configuration synchronization**: Keeps settings synchronized between devices
- **States synchronization**: Maintains connection states and session information
- **VRRP virtual IPs**: Helps provide shared IP addresses for automatic failover

Enable each component independently. However, only when you enable 3 components, a complete HA setup works effectively.

Important: Only when both devices run the same TBF version, HA synchronization starts.

An HA wizard simplifies the setup process for the secondary firewall.

HA topics

To set up and manage HA, use these topics:

- *General high availability* on page 374: Overview and interface setup requirements
- *Configuration sync* on page 380: Synchronize settings between devices
- *States sync* on page 382: Maintain connection states during failover
- *VRRP shared virtual IP address* on page 382: Set up shared IP addresses for failover
- *DHCP high availability* on page 384: Set up DHCP failover support
- *Setup examples* on page 386: Complete configuration examples

General high availability

High availability (HA) in TBF offers redundancy through configuration synchronization, states synchronization, and VRRP virtual IPs.

HA in TBF uses 3 separate and independent components:

- Configuration synchronization
- States synchronization
- VRRP virtual IPs

You can enable each component independently without affecting the others. However, for a complete HA setup to work effectively, enable the 3 components.

HA synchronization begins only when both devices run the same TBF version.

An HA wizard simplifies the setup process for the secondary firewall.

For more information about specific aspects of the HA setup, see:

- *Interface setup* on page 374
- *Synchronization interface* on page 375
- *VRRP mode* on page 377
- *High availability nodes* on page 378
- *Secondary firewall* on page 378
- *Onboarding wizard* on page 378

Interface setup

Set up interfaces on both devices before you start high availability (HA) configuration to help ensure proper synchronization.

TBF only synchronizes interfaces when both devices use the same TBF model. If the models differ, set up the interfaces before you start the HA setup. Because the 2 devices are not identical models, you set up the interfaces individually first.

Perform the following steps:

1. Verify that synchronization depends on consistent internal interface names (WAN, LAN, INT1, or INT2).

   These names match on both the master and client devices. TBF automatically assigns the `INTX` numbers in the background. To view them, look under **Interfaces > Assign** or on each interface settings page in the upper-left corner.

2. Confirm that you have the same number of interfaces on both devices.

3. Confirm that the interface names match on both ends.

4. Confirm that the interfaces use different IP addresses that do not conflict with each other.

5. If the devices have different numbers of interfaces, create dummy interfaces on the secondary firewall using the hardware interface `none`.

   Leave these interfaces disabled.

Synchronization interface

Set up a dedicated synchronization interface for secure and timely high availability data exchange between devices.

Use a synchronization interface for synchronization activities. The data is partially unencrypted and it is necessary that it arrives on each device in a secure and timely manner.

Perform the following steps:

1. Use either a separate VLAN or a separate physical interface.



Figure 276: High availability sync VLAN

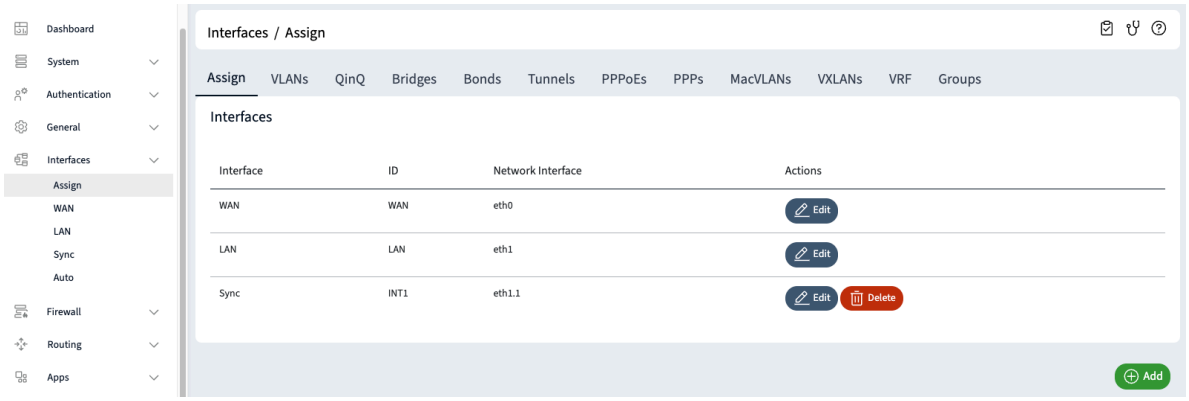


Figure 277: High availability sync interface

2. Give the TBFs a static IP address in this network.

   Do not enable the DHCP server.

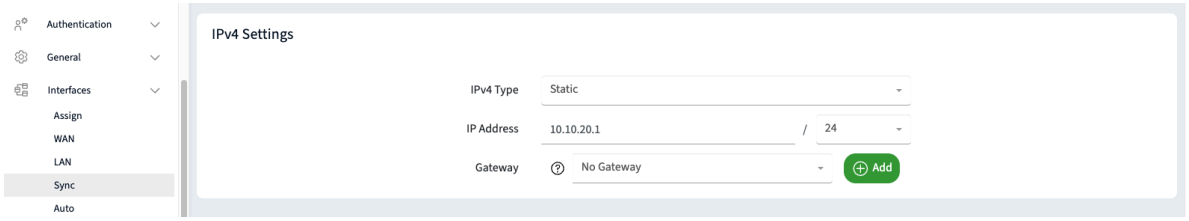


Figure 278: High availability sync interface settings

3. Confirm that a firewall rule allows traffic to the interface IP address on the synchronization interface on each TBF.

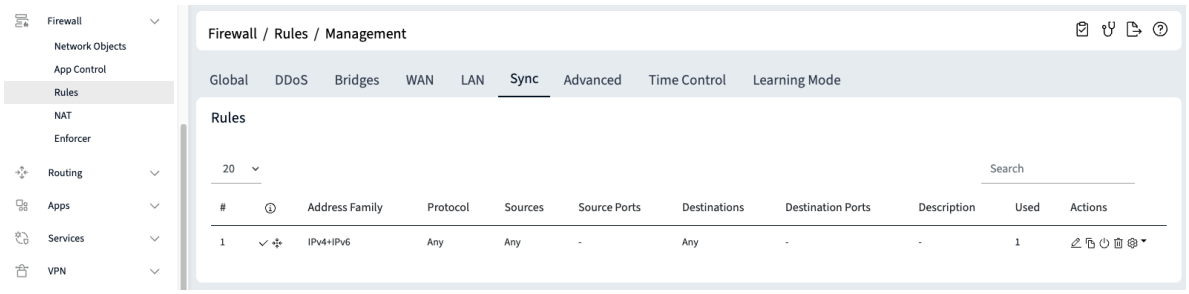


Figure 279: High availability sync firewall rule

4. Set the same password for the `hasync` user on each TBF.
5. Set the synchronization mode.

   Mode defaults to **Multicast**. It allows multiple firewalls to exchange state information. If you have a special requirement, change the mode to **Unicast** and send state information to 1 peer only. Define the peer IP address. Set this setting on both TBFs.

VRRP mode

VRRP mode determines how firewalls exchange Virtual Router Redundancy Protocol state information in multicast or unicast configurations.

The VRRP mode defaults to **Multicast**. TBF exchanges VRRP state information on each interface where you define a VRRP IP via multicast. This exchange also helps confirm that TBF performs a layer 2 check.

In specific cloud or virtual environments, there is no layer 2 multicast connection between firewalls. In this case, set the **Mode** to **Unicast**. A new option is available to also change the VRRP mode from **Multicast** to **Unicast**.

The **Unicast** mode sends the VRRP information on the synchronization interface to the peer IP address instead of on the interfaces where you define the VRRP IP. TBF also detects a loss of the layer 2 check on each interface in this scenario.

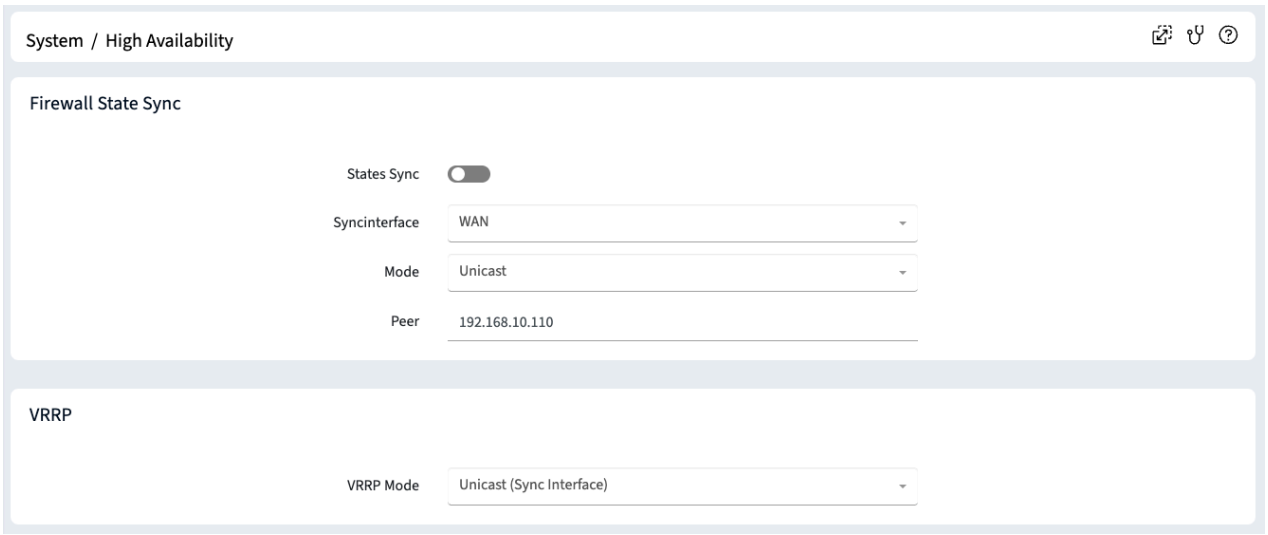


Figure 280: High availability VRRP unicast mode

High availability nodes

High availability (HA) supports multiple nodes in a daisy chain configuration with some system limitations for specific services.

There is no limit to the number of nodes that you add to the HA setup. You can daisy-chain nodes in TBF. Only enable the *configuration synchronization* on page 380 on each node that synchronizes to the next one.

Note: Some systems such as DHCP do not support more than 3 members.

Secondary firewall

The secondary firewall displays visual indicators and has specific configuration restrictions to help prevent conflicts with the master firewall.

The secondary firewall displays a message icon in the upper-right corner.



Figure 281: High availability secondary firewall

Note: The master firewall overrides any configuration changes made to the secondary firewall.

Onboarding wizard

The high availability (HA) onboarding wizard is a streamlined interface for setting up secondary firewalls with predefined configuration options.

The master firewall displays an onboarding wizard the first time you select **High Availability** in the menu.

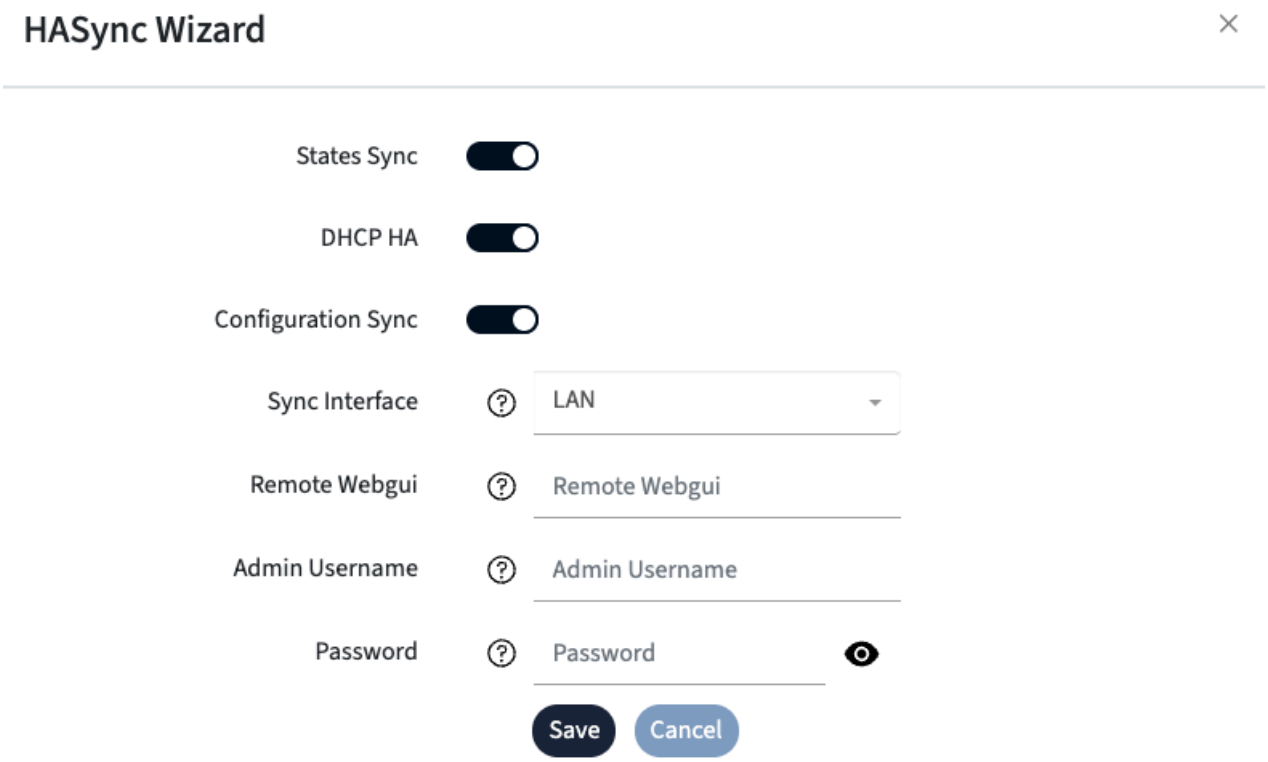


Figure 282: HA synchronization wizard

When you define the main settings, the wizard connects to the secondary firewall to set it up for HA.

Settings

States Sync

This setting enables or disables the synchronization of states between firewalls.

DHCP HA

This setting enables or disables the HA mode for the DHCP server.

Configuration Sync

This setting enables or disables the configuration synchronization from the master firewall to the secondary firewall.

Sync Interface

This setting specifies a separate interface (or VLAN interface) where the master firewall synchronizes data with the secondary firewall.

Remote Webgui

This field contains the URL of the secondary firewall to enable the HA settings. The URL does not need to be the **Sync Interface**. The master firewall only requires access to the secondary firewall on that interface to set it up for HA.

Admin Username

This field specifies the administrator username for the secondary firewall for the onboarding process.

Password

This field contains the password for the administrator user.

Wizard operation

The wizard sets up the secondary firewall for the HA mode. TBF uses the special user `hasync` for the configuration synchronization. TBF randomly generates the password.

Configuration sync

Configuration synchronization automatically transfers settings from a master device to slave devices in a high availability (HA) setup.

Configuration synchronization is a key component of HA that helps ensure the devices in the setup maintain identical configurations. The master device automatically transfers its settings to slave devices whenever you make changes.

TBF triggers synchronization automatically every time you save or change a setting. However, there are a few exceptions. TBFonly synchronizes the following categories when you apply their changes:

- DHCP
- Firewall and NAT
- VirtualIP

You can enable synchronization for different systems. If dependencies exist in the background, TBF synchronizes them automatically even if you have not selected the option.

For example, firewall rules require QoS and routing tables to function properly.

WARNING: Configuration synchronization only requires activation on the master device. The slave device automatically receives the settings from the master device. Enabling configuration synchronization from the slave device to the master device is unsupported. The master device transfers the settings to the slave device.

Note: Interface synchronization only works between TBF devices if both devices are identical hardware models.

Set up configuration sync

Enable and set up configuration synchronization between the master device and slave devices in a high availability (HA) environment.

The HA settings are located under **System > High Availability**.

The process begins with **Configuration Sync**. Enable or disable the entire synchronization process.

Perform the following steps:

1. Navigate to **System > High Availability**.
2. Enable **Configuration Sync**.
3. Enter the remote TBF device URL in the **Remote TBF** field.
   Enter a URL that uses the synchronization interface, for example, `https://10.10.20.2`.
4. Enter the synchronization username in the **Username** field.
   A default user called `hasync` already has the necessary synchronization permissions.
5. Enter the password for the synchronization user in the **Password** field.
6. Select or deselect synchronization options using the appropriate buttons.

Configuration synchronization is now active and automatically transfers settings from the master device to slave devices.

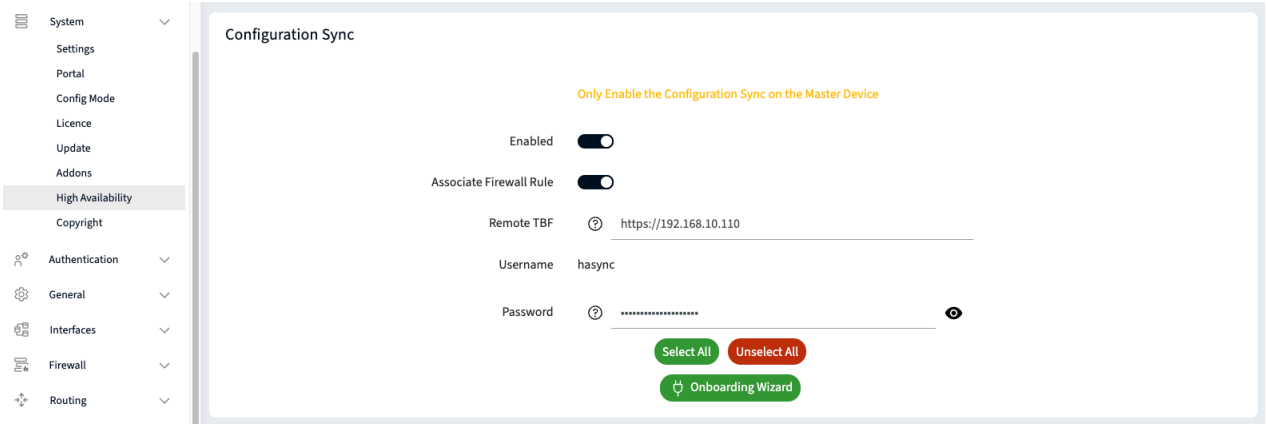


Figure 283: High availability configuration sync

Configuration sync settings

Field definitions and parameters for configuration synchronization settings.

Settings

Remote TBF

This field specifies the URL of the TBF device you synchronize with. Enter a URL that uses the synchronization interface, for example, `https://10.10.20.2`. Create a firewall rule on the synchronization interface to allow access to the web console. Additionally, enable the API and verify that it remains active.

Username

This field specifies the username for logging in to the remote TBF device. A default user called `hasync` already has the necessary synchronization permissions. TBF randomly generates the password during the initial setup. Set the same password on both the slave and master devices.

Password

This field specifies the password for the synchronization user.

Notes

Note: The initial synchronization sets the user password of the slave device to match the `hasync` user password of the master device. Confirm that both passwords are identical.

Trigger manual synchronization

Manually trigger configuration synchronization on any system page where synchronization is enabled.

Configuration synchronization happens automatically when you save or change settings. You can also trigger manual synchronization when needed.

Perform the following steps:

1. Navigate to any system page where you have enabled synchronization.
2. Select the **Synchronization** button in the upper-right corner.

   TBF immediately synchronizes the configuration of the current page to slave devices.

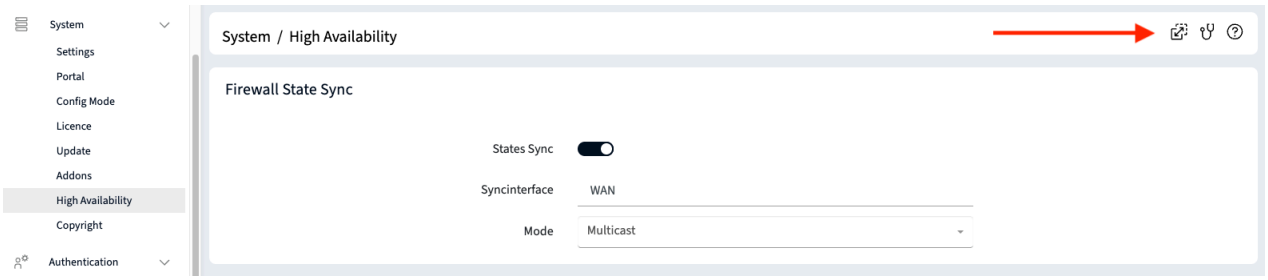


Figure 284: High availability manual synchronization

States sync

States synchronization maintains firewall state information across devices in a high availability (HA) setup, helping ensure seamless failover and consistent network connections.

You can find the high availability settings under **System > High Availability**.

The configuration begins with **Firewall State Sync**. Enable or disable the entire synchronization process.

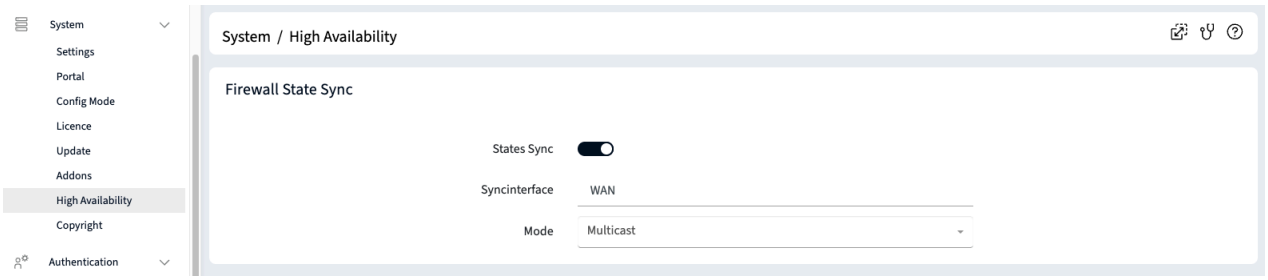


Figure 285: High availability states synchronization

Syncinterface is the interface to which the state updates publish. TBF sends states as multicast messages, which clients in that network receive.

Note: These messages are unencrypted.

TBF automatically generates a firewall rule when you enable this setting. Enable **States Sync** on TBF devices (master and slaves) that participate in the HA setup.

This feature synchronizes the firewall state information, and TBF automatically adds it to TBFs that have the option enabled.

TBF first saves states from other firewalls to a special external states cache instead of applying them directly to the state table. This approach enables faster and more efficient synchronization of states while they remain unused. When a VRRP failover occurs, TBF applies these cached states to the active state table of the firewall.

To view both active states and external states in the states diagnostics, navigate to **Diagnostics > States**. To view external states, you can also use the shell command `conntrackd -e`.

VRRP shared virtual IP address

Set up VRRP shared virtual IP addresses to help provide automatic failover for network interfaces.

Set up VRRP shared virtual IP addresses on the **General > Virtual IPs** page.

Perform the following steps:

1. On the master node, add a virtual IP address to each interface where you enable failover.
2. Verify that the virtual IP addresses are within the same subnet as an IP address defined on a physical interface (WAN, LAN, INT1, etc.).
3. Use a unique router ID for each shared virtual IP address on a given interface.

   The device with the highest priority becomes the master for each VRRP instance.
4. Set the priority value for synchronization.

   The synchronization automatically reduces each priority by 10 when it synchronizes the VIPs to the next TBF. The default priority value is `100` and you must keep it unchanged.

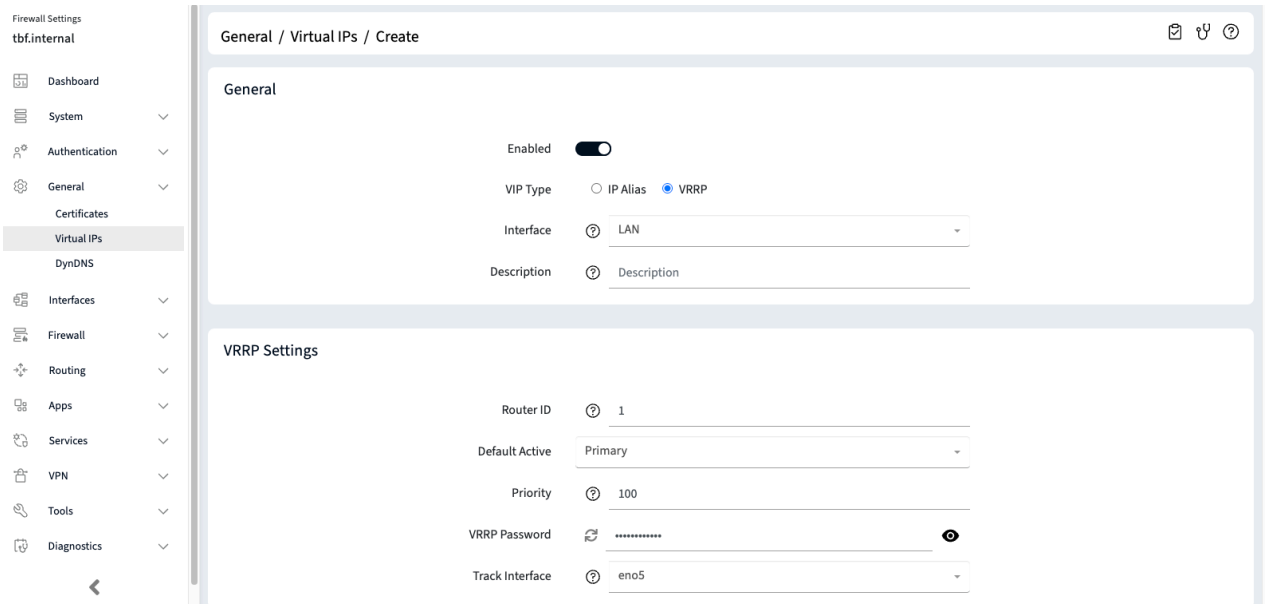


Figure 286: Virtual IP configuration

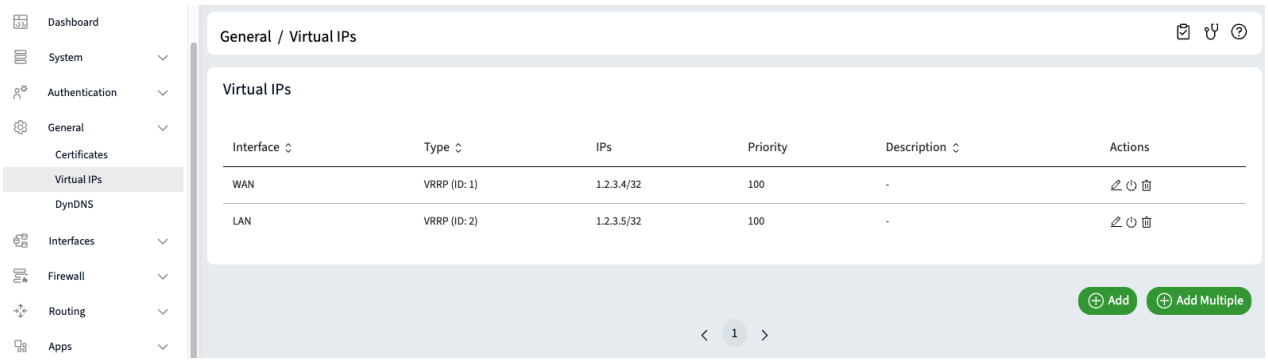


Figure 287: Virtual IPS

For additional VRRP configuration tasks:

- *HA outbound NAT* on page 384
- *Gateway IP address* on page 384

HA outbound NAT

Set up outbound NAT to use VRRP shared virtual IP addresses for proper failover functionality.

For failover to work properly, change the outbound NAT to use the VRRP shared virtual IP address.

Perform the following steps:

1. On each WAN interface that has a VRRP shared virtual IP address defined, create a SNAT rule in *SNAT (postrouting)* on page 144.
2. Set up the rule on the interface where you defined the VRRP shared virtual IP address.
3. Set the appropriate source IP addresses if needed.
4. Select the VRRP shared virtual IP address as the translation IP address.

This way, TBF changes the traffic leaving the interface to come from the VRRP shared virtual IP address. During a failover, the backup TBF can continue sending traffic from this address without any traffic loss.

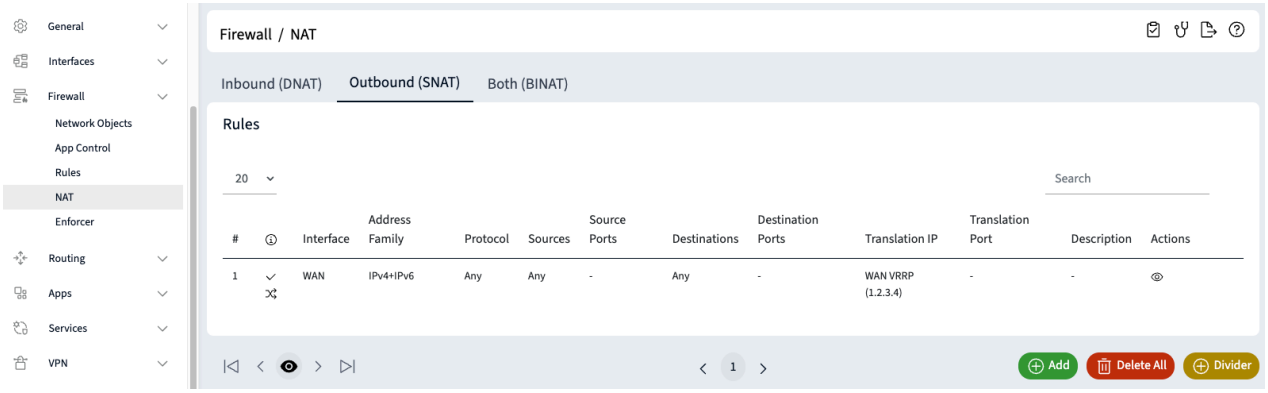


Figure 288: Virtual IP SNAT configuration

Gateway IP address

Set up VRRP shared virtual IP addresses to function as gateway addresses for clients on internal subnets.

A VRRP shared virtual IP address also functions as the gateway address for clients on the internal subnets or interfaces. You can set the gateway VRRP shared virtual IP address manually or use the DHCP server.

For the DHCP server, see *DHCP & RA* on page 259.

Perform the following steps:

Set the gateway settings:

- Set the default gateway to a VRRP shared virtual IP address on each internal network (LAN, for example `192.168.1.3`).
- Set the DNS server to the VRRP shared virtual IP address on the internal network (LAN, for example `192.168.1.3`).

DHCP high availability

This page describes the DHCP high availability configuration.

The DHCP high availability (HA) settings are located under **System > High Availability**.

The configuration begins with **DHCP High Availability**. DHCP is unique in the HA setup because it supports a maximum of 3 TBFs in the configuration. Additional TBFs require DHCP to be disabled, which TBF handles automatically when you enable the settings on each node.

You can enable or disable the entire DHCP HA setup.

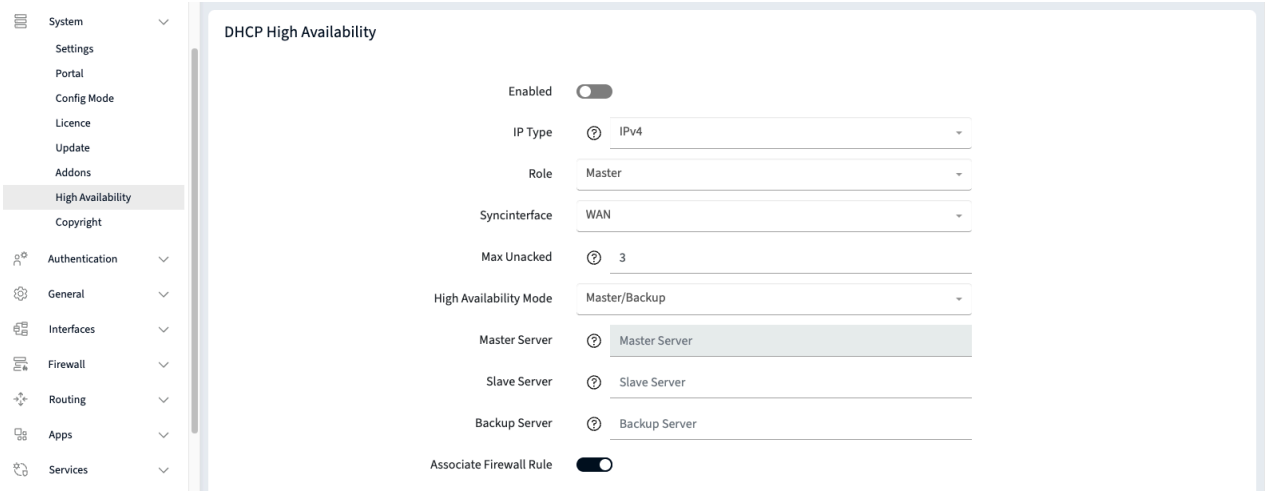


Figure 289: DHCP high availability configuration

DHCP high availability settings

This page describes the configuration parameters for DHCP high availability.

Settings

Enabled

This setting enables or disables DHCP high availability (HA).

IP Type

This setting supports either DHCPv4 or DHCPv6 HA, but avoids both at the same time. This is a current limitation of the DHCP daemon.

Role

This setting determines the DHCP HA role. Set this to **Master** on the initial master TBF. The setup includes only 1 slave and 1 backup. The configuration synchronization automatically assigns the role on the next TBF, so do not change this setting on each member. The **Disable** role disables DHCP of this type for the setup to function properly. Without this role, 2 DHCP servers compete to assign leases.

Syncinterface

This setting uses the separate synchronization interface, as TBF synchronizes information without encryption.

Max Unacked

This field defines how many unacknowledged clients this server allows before it determines the partner is offline and transitions to the *partner-down* state. The special value of `0` disables the failure-detection mechanism. In this case, a server that cannot communicate with its partner over the control channel immediately determines the partner server is down and transitions to the *partner-down* state. The default is `3`.

High Availability Mode

This setting determines the HA mode. Select from:

- **Master/Backup**
- **Load Balancing**

Master Server

This field displays the IP address of the master server. On the master server, TBF populates this field automatically.

Slave Server

This field displays the IP address of the slave server. On the slave server, TBF populates this field automatically.

Backup Server

This field displays the IP address of the backup server. On the backup server, TBF populates this field automatically.

Associate Firewall Rule

This setting creates the firewall rule to allow connections between the TBFs for the DHCP HA setup. Enable this setting.

Setup examples

Set up and use the high availability (HA) feature based on specific requirements with different common example setups.

This page shows different common example setups. The capabilities are not limited to these setups.

The 3 core components of the HA feature are *independent* of each other, allowing you to set up and use them individually.

- Configuration sync
- States sync
- VRRP virtual IPs
- (DHCP HA)

Related information
Configuration sync only on page 386
Full failover setup on page 386

Configuration sync only

Set up configuration synchronization without automatic failover protection for manual device swapping or multi-site consistency.

This section describes a setup where the TBF devices synchronize the configuration but do not provide automatic failover protection. This approach is useful when you manually swap in a secondary TBF device if the primary device detects a loss of communication. The configuration synchronization feature helps assure that the configuration of the secondary device remains up to date.

Another scenario involves configuration synchronization across a common network that spans multiple sites. To maintain consistent configuration parts across multiple sites without individually setting up each device, use the configuration synchronization feature.

Refer to *configuration sync* on page 380 for setup details. If the local network connects to the LAN interface, then create a VLAN on top of the LAN interface with static IP addresses for the TBF devices. The devices then share the configuration through this VLAN. Alternatively, use a separate physical interface for this purpose.

Full failover setup

Set up automatic failover support with configuration synchronization and virtual IP addresses for networks requiring minimal downtime.

If you require fully automatic failover support, set up the TBF devices to automatically synchronize their configuration and automatically switch to a backup device in the case of a detected loss of communication. This approach is useful for networks in remote locations that cannot tolerate significant downtime.

Refer to the section above for details on setting up the configuration synchronization first. Also refer to *DHCP high availability* on page 384 if TBF provides DHCP functionality to the network.

Each router receives its unique IP address in the local network. For an automatic failover setup, the default gateway and its IP address cannot change suddenly during operation. Therefore, the default gateway requires a virtual IP address that the routers share in the network. See *VRRP shared virtual IP address* on page 382 for configuration details. The same principle also applies to the `WAN` interface. When a secondary device continues operation for the primary device, the IP address cannot change.

Note: The HA feature does *not* detect device or media errors of cables or causes outside the device itself. It is designed only to register inoperable devices and re-route traffic through a secondary device.

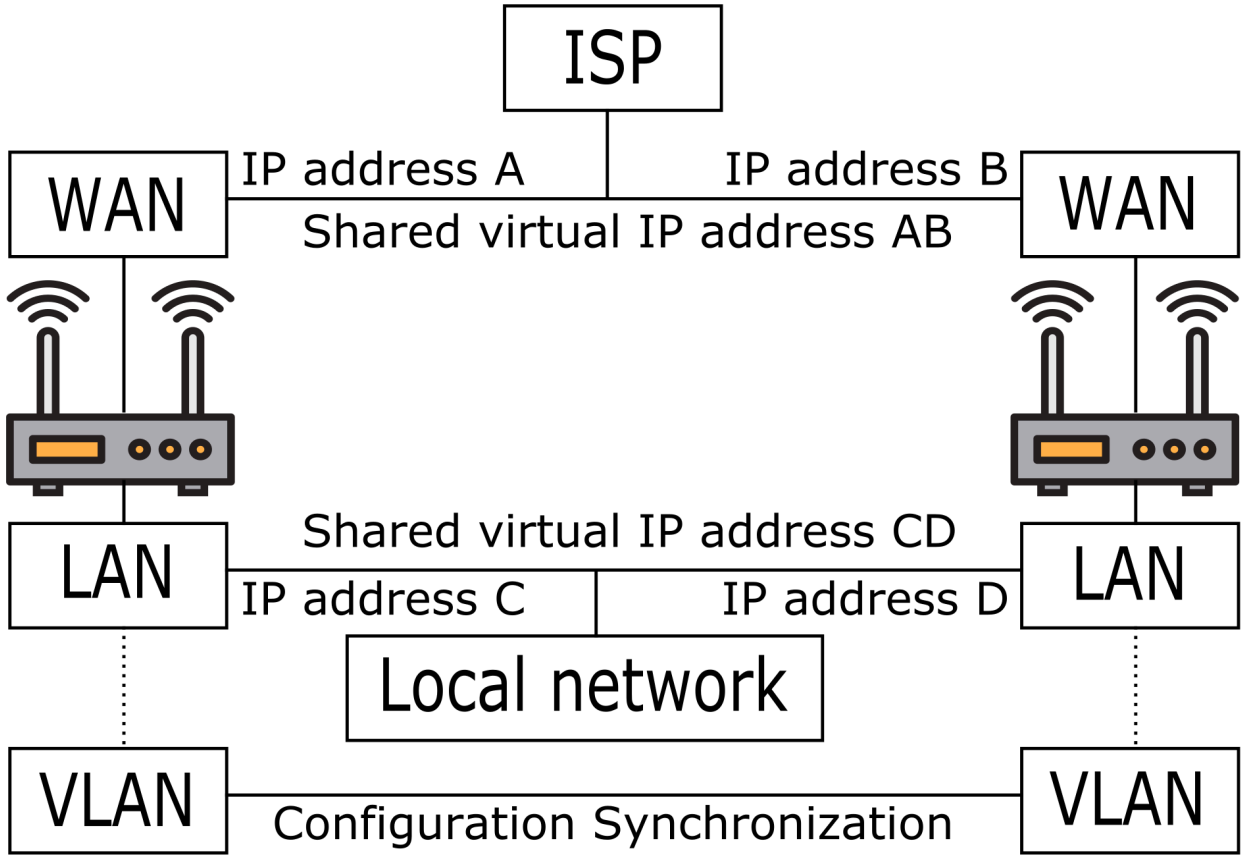


Figure 290: Full failover setup

This image was created with icons by *srip* from *Flaticon*.

20. Diagnostics

This page provides an overview of the diagnostic tools available in TBF.

Find the diagnostics under **Diagnostics**.

TBF systems include a diagnostics page that contains detailed output about the services and software in use. Debug issues and gather information to assist you with virtually any detected problem that occurs in the network setup.

Diagnostic pages refresh automatically. Also manually trigger a refresh by selecting the **Refresh** symbol in the upper-right corner.

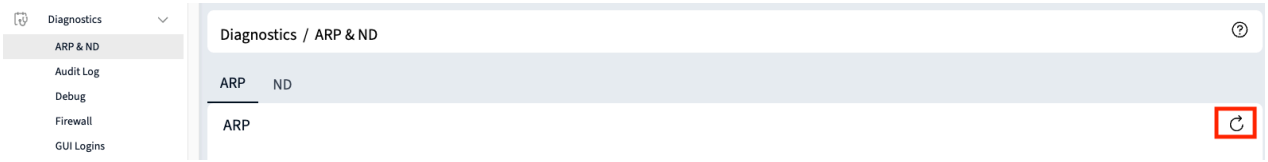


Figure 291: Refresh

ARP & ND

This page displays Address Resolution Protocol (ARP) entries and Neighbor Discovery (ND) entries.

Find the ARP & ND settings under **Diagnostics > ARP & ND**.

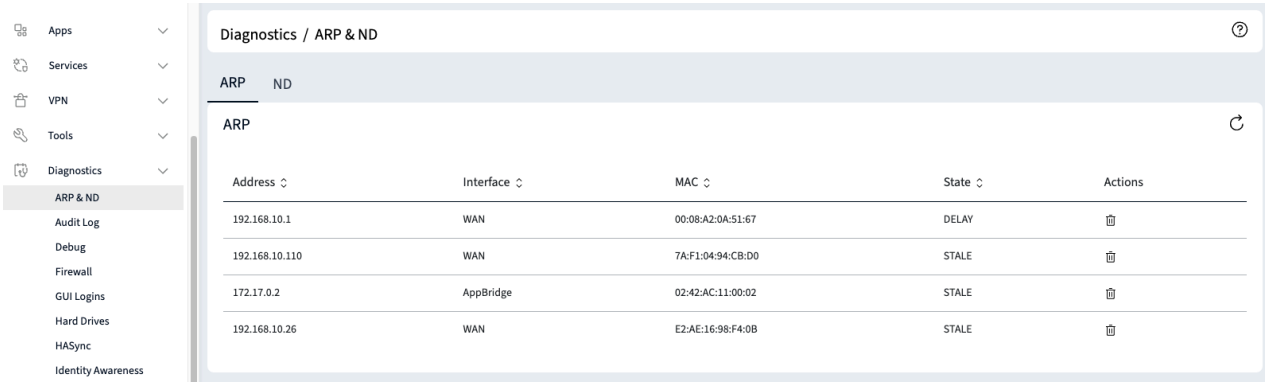


Figure 292: ARP

This page displays an overview of Address Resolution Protocol (ARP) entries and Neighbor Discovery (ND) entries. IPv4 networks use ARP, while IPv6 uses ND for address resolution.

Each item shows the following information:

- IP **Address**
- **Interface**
- **MAC** address
- **State**, such as `STALE`, `DELAY`, `REACHABLE`, or `FAILED`

Under **Actions**, delete entries to remove them from the system.

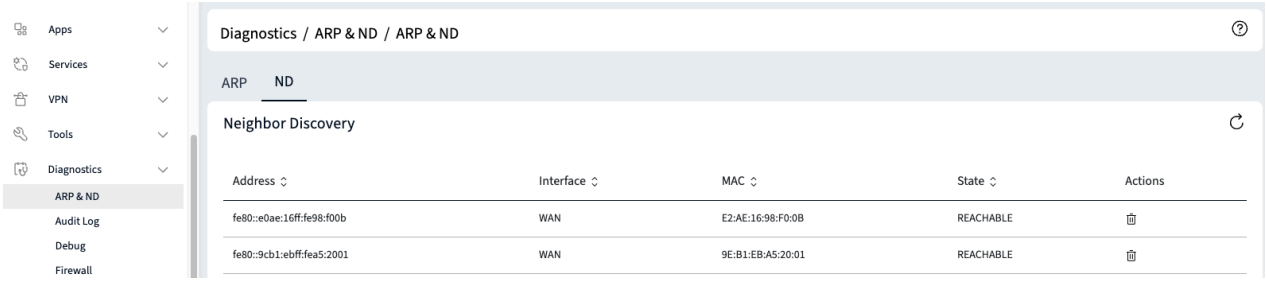


Figure 293: Neighbor discovery

Audit log

This page displays changes made to the system, whether performed by users or triggered automatically.

Find the audit log under **Diagnostics > Audit Log**.

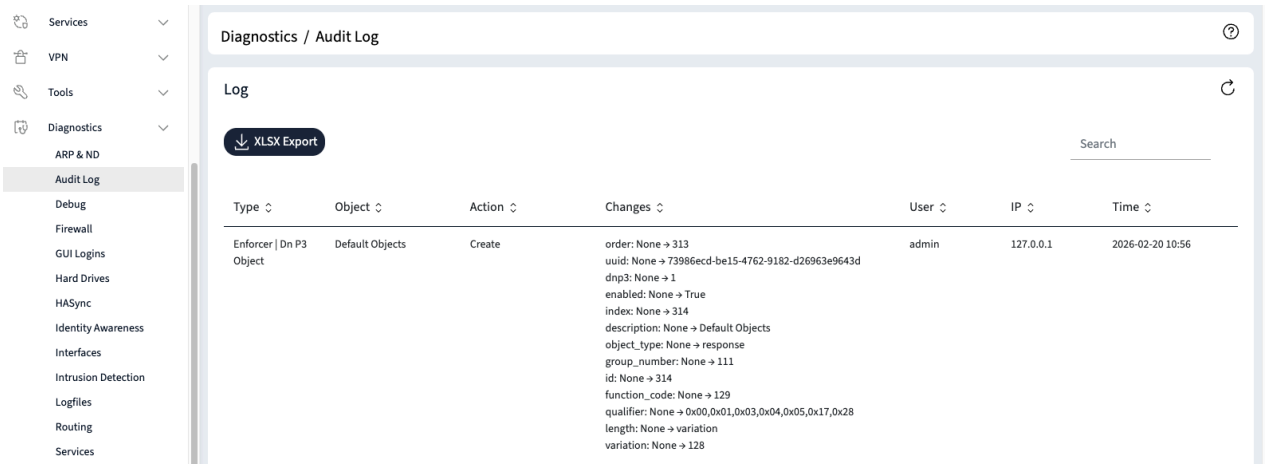


Figure 294: Audit log

This page displays changes made to the TBF system, whether performed by users or triggered automatically by the system itself.

The audit log contains the following information columns:

- **Type**: Indicates the category where the change occurred.
- **Object**: Shows the name of the affected database object.
- **Action**: Specifies the type of operation performed: create, update, or delete.
- **Changes**: Contains a detailed overview of which fields were modified and their previous values.
- **User**: Identifies which TBF user made the change, or displays `None` if the system automatically triggered the change.
- **IP**: Appears only for user-initiated changes and shows the network address from which the change was made.
- **Time**: Records the date and time when the change occurred.

Search for specific changes using the **Search** function, which filters the results based on keywords found in the database object names.

In the upper-left corner, the **XLSX Export** button downloads audit log entries as an Excel file for further analysis.

To send audit log data to syslog, enable this feature in the *logging diagnostics* on page 413 section. The corresponding log file is named `audit.log`.

802.1X authenticator

This page displays authenticated 802.1X clients, timeout settings, and client disconnect options.

Note: This section contains screenshots from the former VT Air GUI. These will be updated in a future version.

Find the 802.1X authenticator diagnostics under **Diagnostics > Services > 802.1X Authenticator**.

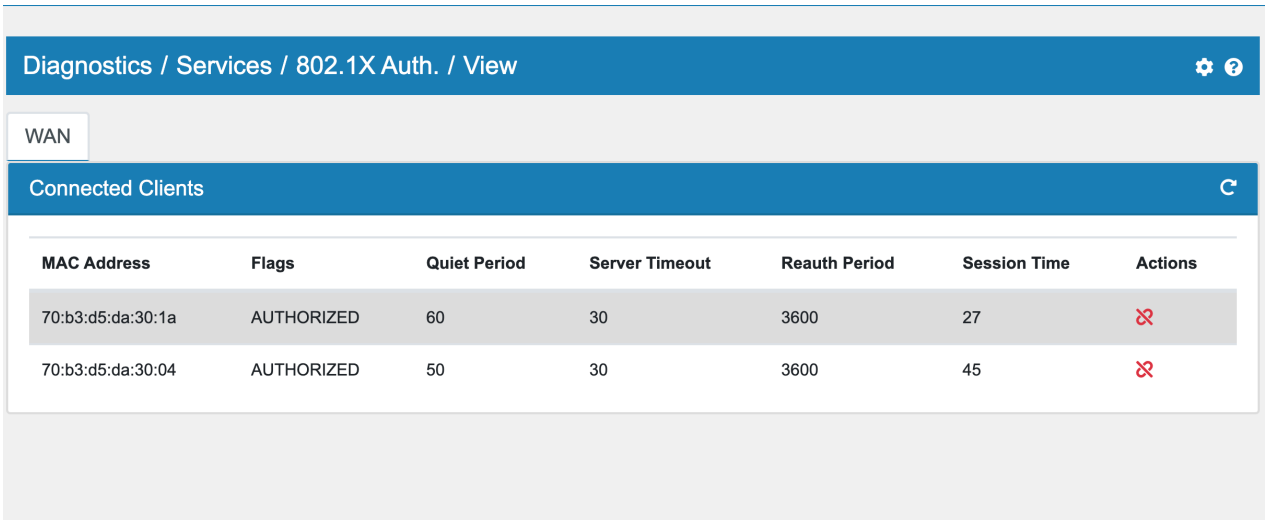


Figure 295: 802.1X authenticator

Here you have an overview of the authenticated clients, timeout settings, and helps you disconnect a client when needed.

BFD

This page displays Bidirectional Forwarding Detection (BFD) peer status and session information.

Find the BFD diagnostics under **Diagnostics > Routing > BFD**.



Figure 296: BFD

Here you see information about the Bidirectional Forwarding Detection (BFD) setup.

Note: TBF automatically refreshes the information on this page at regular intervals without needing any user input.

To change the BFD setup, navigate to **Routing > BFD** and see the *BFD settings* on page 223.

BGP

This page displays Border Gateway Protocol (BGP) routing information and neighbor status.

Find the BGP diagnostics under **Diagnostics > Routing > BGP**.

Here you see information about the Border Gateway Protocol (BGP) setup. The diagnostics are organized into the following categories:

- **Summary**
- **Neighbors**
- **IPv4**
- **IPv6**
- **NextHop**
- **Memory**

Note: TBF automatically refreshes the information on this page at regular intervals without the need for any user input.

To change the BGP setup, navigate to **Routing > BGP** and see the *BGP settings* on page 224.

Bond

This page displays bond interface status and member link information.

Find the bond diagnostics under **Diagnostics > Interfaces > Bond**.

Here you see bond information.

Select a bond from the list to view its detailed information.

To change the bond settings, navigate to **Interfaces > Assign > Bonds** and see *bond* on page 104.

The output for bond looks like this:

```
Ethernet Channel Bonding Driver: v6.6.0-1-tbf-arm64

Bonding Mode: load balancing (xor)
Transmit Hash Policy: layer2+3 (2)
MII Status: up
MII Polling Interval (ms): 100
Up Delay (ms): 100
Down Delay (ms): 100
Peer Notification Delay (ms): 0

Slave Interface: eno4
MII Status: up
Speed: 10000 Mbps
Duplex: full
Link Failure Count: 0
Permanent HW addr: c8:98:db:80:02:62
Slave queue ID: 0

Slave Interface: eno5
MII Status: up
Speed: 10000 Mbps
Duplex: full
Link Failure Count: 0
Permanent HW addr: c8:98:db:80:02:63
Slave queue ID: 0
```

Bridge

This page displays bridge interface information and MAC address tables.

Find the bridge diagnostics under **Diagnostics > Interfaces > Bridge**.



Figure 297: Bridge result

Here you see bridge information organized into 2 categories:

- **Bridge**
- **Mac**

Select a bridge to view its information.

To change the bridge settings, navigate to **Interfaces > Assign > Bridges** and see *bridge* on page 99.

The output for the bridge looks like this:

```
bridge  name    bridge id           STP enabled  interfaces
br0             8000.0008a20a8857   yes          enp0s20f2
```

The output for Mac looks like this:

```
port no mac addr            is local?   ageing timer
1       00:08:a2:0a:88:57   yes         0.00
1       00:08:a2:0a:88:57   yes         0.00
```

Captive portal

This page displays captive portal session information and connected clients.

Find the captive portal diagnostics under **Diagnostics > Services > Captive Portal**.

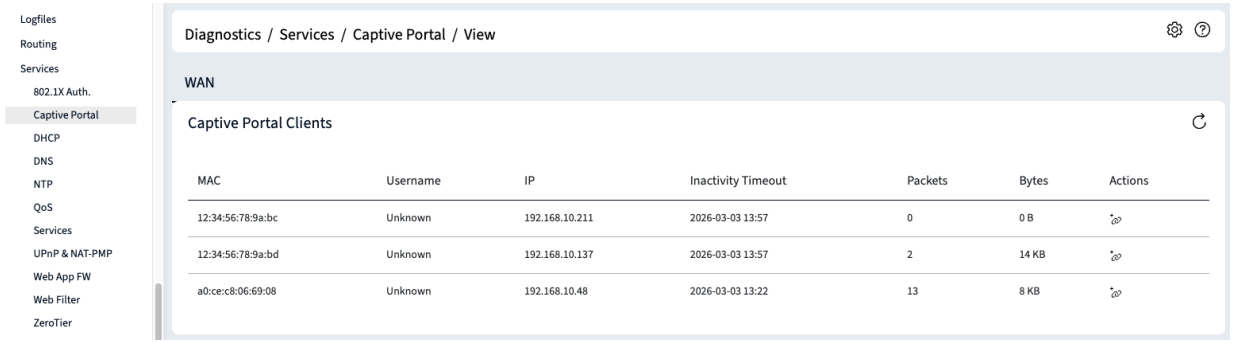


Figure 298: Captive portal clients

Here you see information about the various captive portal setups. For each connected client, you can view the following details:

- MAC address
- Username
- IP address
- Session expiration time

You can manually disconnect clients when necessary.

DDoS

This page displays clients blocked due to DDoS activity.

Find the DDoS diagnostics under **Diagnostics > Firewall > DDoS**.

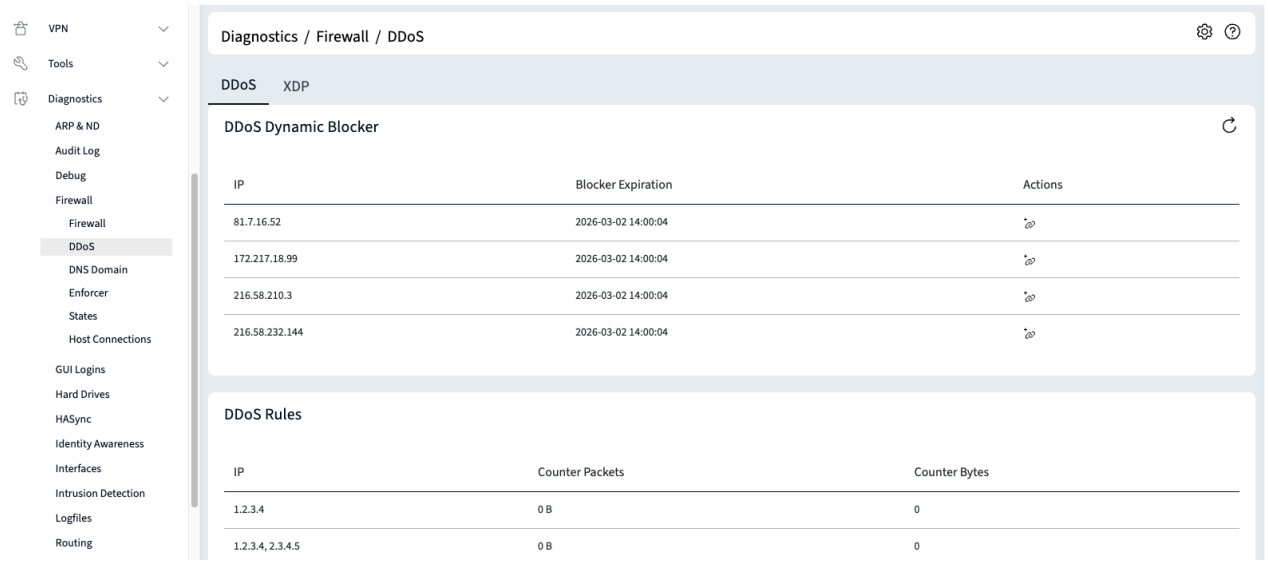


Figure 299: DDOS

Here you see information about clients that have been blocked because of DDoS activity. TBF categorizes these blocks into 2 types:

- Clients blocked by firewall DDoS rules
- Dynamic block entries

To change the blocked clients from firewall DDoS rules, navigate to **Firewall > Rules** and select **DDoS**. See also the *DDoS rules* on page 165.

You can manually remove dynamic block entries. TBF also shows the remaining block time for each entry.

Debug report

This page generates a debug report with system uptime, disk space, and service status information.

Find the debug report under **Diagnostics > Debug > Debug Report**.

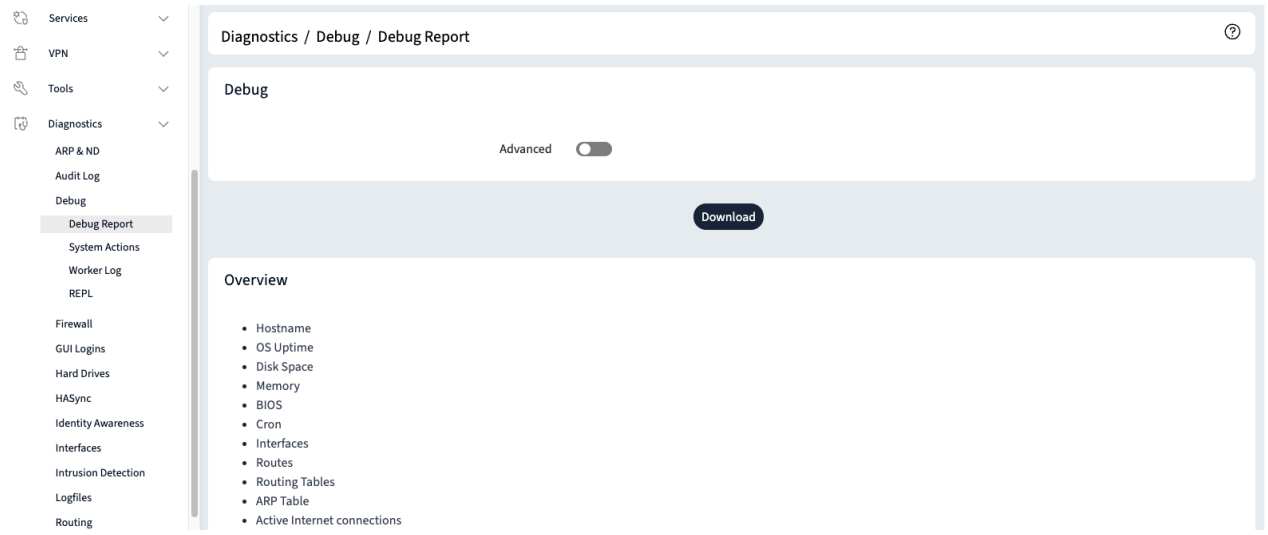


Figure 300: Debug report

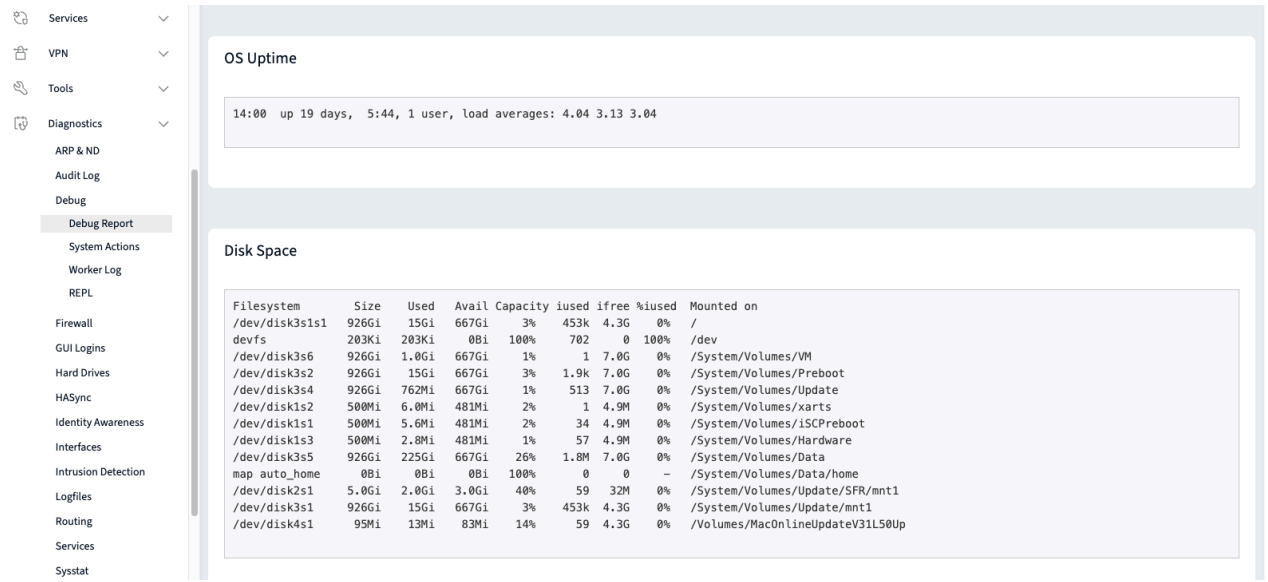


Figure 301: OS uptime and disk space

The debug report contains diagnostics and debugging information about the system. When you contact Belden for technical support, you are sometimes asked to provide a debug report to help resolve the issue.

Note: For security purposes, TBF automatically removes passwords from the generated report.

The debug report includes the following information to help you identify and troubleshoot the system:

- Log files
- Configurations
- System reports

DHCP

This page displays DHCP server lease information for IPv4 and IPv6.

Find the DHCP diagnostics under **Diagnostics > Services > DHCP**.

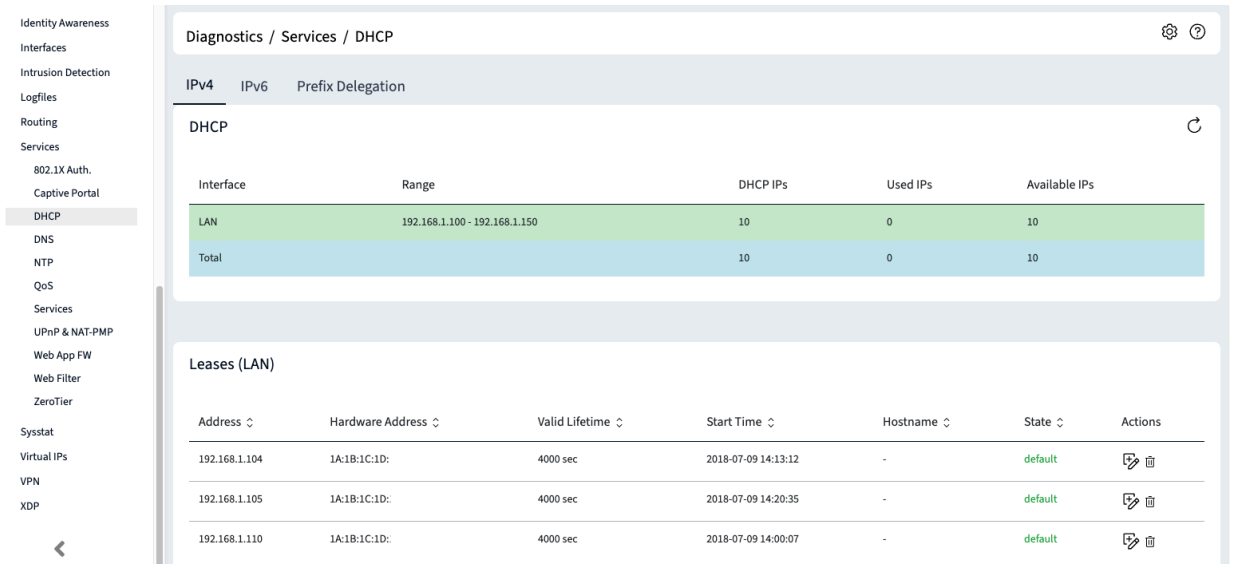


Figure 302: DHCP

Here you see information about the DHCP server. TBF divides it into IPv4 and IPv6. Both tabs have a **DHCP** table and a **Leases** table.

Under **DHCP**, you see the interfaces where the DHCP server is enabled and their pools.

Under **Leases**, you see each lease with its IP **Address**, **Hardware Address**, and **Hostname**. The **Valid Lifetime** and **Start Time** show how long the lease remains valid. Lastly, you see the **State** of the lease.

In the **Actions** column, you can edit, create, or delete static leases, as well as remove dynamic leases from the server.

To change the DHCP setup, navigate to **Services > DHCP & RA** and see *DHCP & RA* on page 259.

DHCP HA modes

During normal high availability (HA) operation, the DHCP server runs in hot-standby mode. The master server handles DHCP lease distribution, and the secondary server remains in standby mode.

The DHCP server operates in the following states:

- `backup`: Normal operation mode for the backup server. In this state, the server receives lease updates from active servers.
- `hot-standby`: Normal operation mode for the active server running in hot-standby configuration. Both primary and standby servers remain in this state during normal operation. The primary server responds to DHCP queries and sends lease updates to the standby server and any existing backup servers.
- `in-maintenance`: An active server enters this state when its partner notifies it that an administrator has requested maintenance of the HA setup. The administrator initiates maintenance by sending the `ha-maintenance-start` command to the server that handles DHCP client responses and the other server goes offline for maintenance. When a server remains in the `in-maintenance` state, safely shut it down. The partner remains in the `partner-in-maintenance` state and transitions to the `partner-down` state immediately after detecting that the maintenance server has shut down.
- `partner-down`: An active server enters this state after detecting that its partner (another active server) has gone offline. The server does not enter this state when only a backup server becomes unavailable. In the `partner-down` state, the active server responds to all DHCP queries, including those that the unavailable server normally handles.
- `partner-in-maintenance`: An active server enters this state after receiving the `ha-maintenance-start` command from an administrator. The server in this state becomes responsible for responding to DHCP requests and sends the `ha-maintenance-notify` command to the partner, which enters the `in-maintenance` state. The server in the `partner-in-maintenance` state continues sending lease updates to the partner until it detects that the partner stops responding to lease updates, heartbeats, or other commands. When this occurs, the server transitions to the `partner-down` state and continues responding to queries but stops sending lease updates.

- `passive-backup`: A primary server running in passive-backup HA mode enters this state immediately after startup. The primary server in this state handles all DHCP traffic and sends lease updates to connected backup servers. By default, the primary server does not wait for acknowledgments from backup servers and responds to DHCP queries immediately after sending lease updates. If lease updates fail, backup servers may miss updates, but DHCP clients are still provisioned.
- `ready`: An active server enters this state after synchronizing its lease database with an active partner. This state signals to the partner, which may be in the `partner-down` state, that it returns to normal operation. When the partner returns to normal operation, the server in the `ready` state also begins normal operation.
- `syncing`: An active server enters this state to retrieve leases from the active partner and update the local lease database. In this state, the server issues the `dhcp-disable` command to disable the DHCP service of the partner from which it retrieves leases. The system disables the DHCP service for a maximum of 60 seconds, after which it automatically re-enables if the syncing partner cannot re-enable the service. When synchronization completes, the syncing server issues the `dhcp-enable` command to re-enable the DHCP service of its partner. The syncing operation runs synchronously; the server waits for a response from the partner and performs no other tasks during lease synchronization. A server set up not to synchronize the lease database with its partner (when the `sync-leases` configuration parameter is set to false) never enters this state. Instead, it transitions directly from the `waiting` state to the `ready` state.
- `terminated`: An active server enters this state when the high availability hooks library cannot provide reliable service and requires manual administrator intervention to resolve the problem. Various HA setup issues may cause the server to enter this state. While in this state, the server continues responding to DHCP clients based on the selected HA mode (load-balancing or hot-standby), but the system does not exchange lease updates or send heartbeats. Once a server enters the `terminated` state, it remains there until you restart it. The administrator is required to correct the underlying issue before restarting the server (for example, synchronize the clocks). Otherwise, the server returns to the `terminated` state when it detects that the issue persists.
- `waiting`: Each server instance enters this state upon startup. Backup servers transition directly from this state to the `backup` state. An active server sends a heartbeat to its partner to check its status. If the partner appears unavailable, the server transitions to the `partner-down` state. If the partner is available, the server transitions to either the `syncing` or `ready` state, depending on the `sync-leases` configuration parameter setting. During concurrent startup (when both servers are in the `waiting` state), the primary server transitions to the next state first. The secondary or standby server remains in the `waiting` state until the primary transitions to the `ready` state.

DNS

This page displays current DNS entries including DHCP static, DHCP dynamic, and host entries.

Find the DNS diagnostics under **Diagnostics > Services > DNS**.

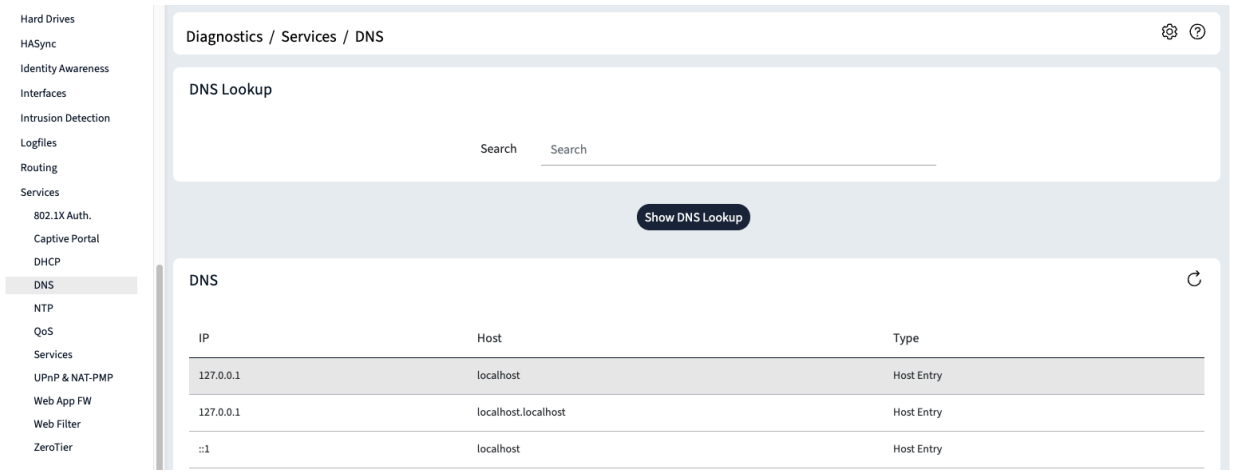


Figure 303: DNS

Here you see the current DNS entries. There are 3 different types available:

- DHCP static
- DHCP dynamic
- Host entry

To change the DNS setup, navigate to **Services > DNS** and see *DNS* on page 263.

To perform a DNS lookup for a specific domain directly on the firewall, select **Show DNS Lookup**.

DNS domain

This page displays IPv4 and IPv6 entries for DNS wildcard domains.

Find the DNS domain diagnostics under **Diagnostics > Firewall > DNS Domain**.

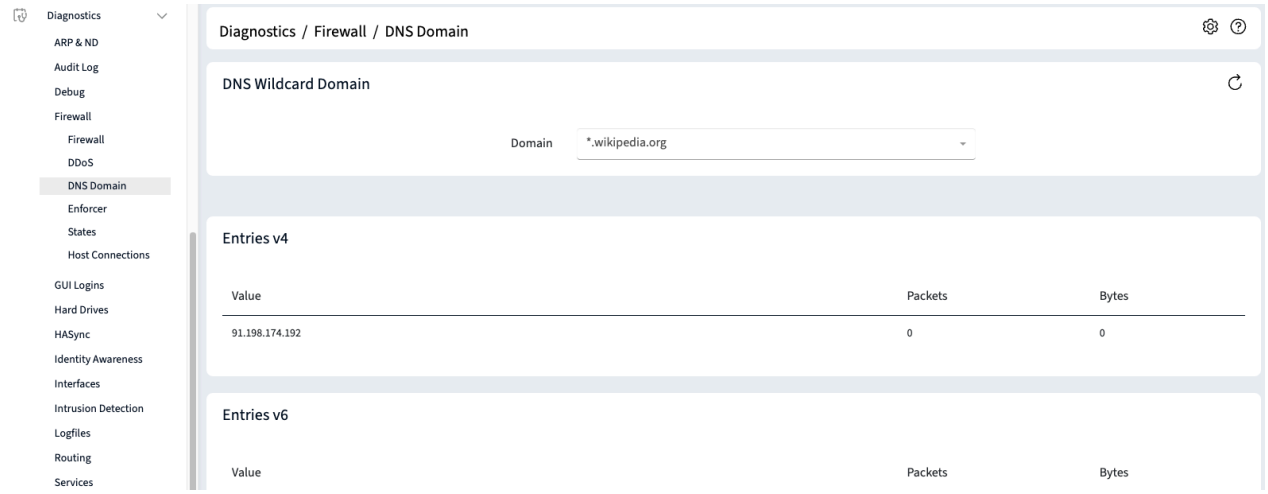


Figure 304: DNS domain

To view the IPv4 and IPv6 entries of a DNS wildcard domain, select the domain.

Note: TBF automatically updates the information on this page at regular intervals without requiring any user input.

To change the DNS domain setup, navigate to **Firewall > Network Objects**.

Dynamic routing

This page displays dynamic routing information including FRR routes, access lists, prefix lists, and route maps.

Find the dynamic routing diagnostics under **Diagnostics > Routing > Dynamic Routing**.

Here you see information about the dynamic routing setup. TBF organizes the information into the following categories:

- FFR routes
- FFR routes v6
- Access list
- Access list v6
- Prefix list
- Prefix list v6
- Route map

Note: TBF automatically refreshes the information on this page at regular intervals without requiring any user input.

To change the dynamic routing setup, navigate to **Routing > Dynamic**. For more information, see *dynamic* on page 222.

Firewall

This page displays firewall logs, active rulesets, and dashboard statistics.

Find the firewall diagnostics under **Diagnostics > Firewall > Firewall**.

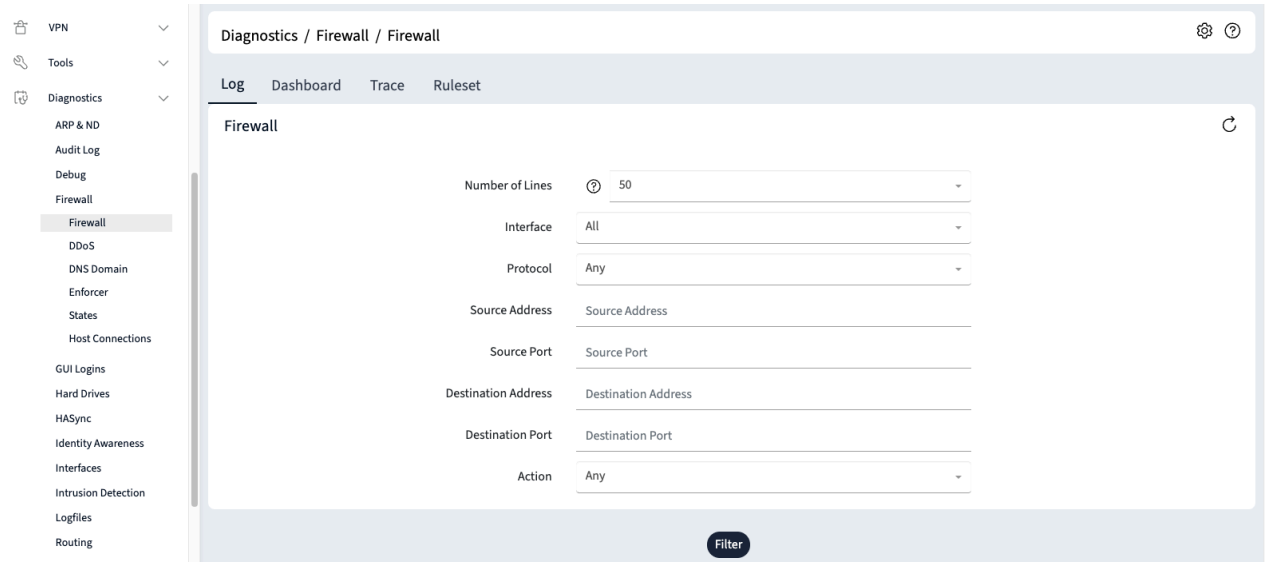


Figure 305: Firewall

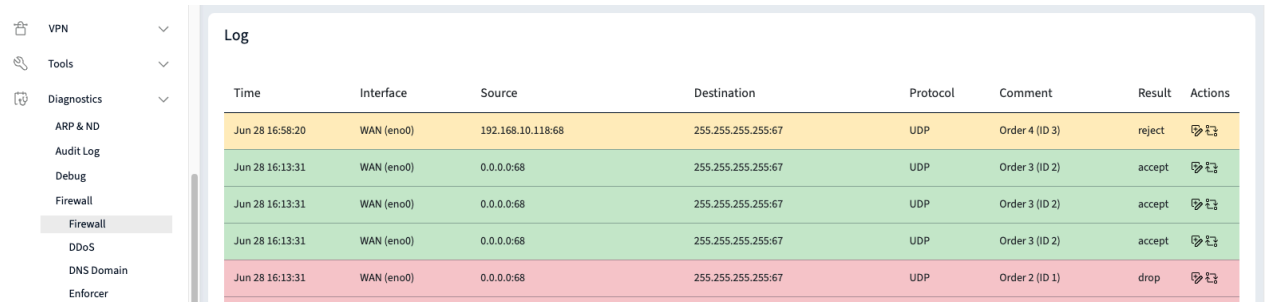


Figure 306: Log

There are 3 tabs. On the **Log** tab, browse, search, and parse the current firewall log. Each log entry comes from the firewall log.

Important: By default, TBF logs only deny entries. If you need more logging, enable it in each firewall rule. See the *firewall rules (forward and input)* on page 131 for more details.

Create a firewall rule directly from a log entry by selecting the symbol in the **Actions** column.

Source and destination IP addresses show reverse DNS entries on hover as long as TBF resolves the IP address.

On the **Ruleset** tab, view the current system firewall ruleset.

The **Dashboard** tab displays an interactive view of the collected data from log events.

Firewall dashboard

The **Dashboard** tab shows you firewall information by country and origin.

The dashboard is a convenient way to visualize the event data.

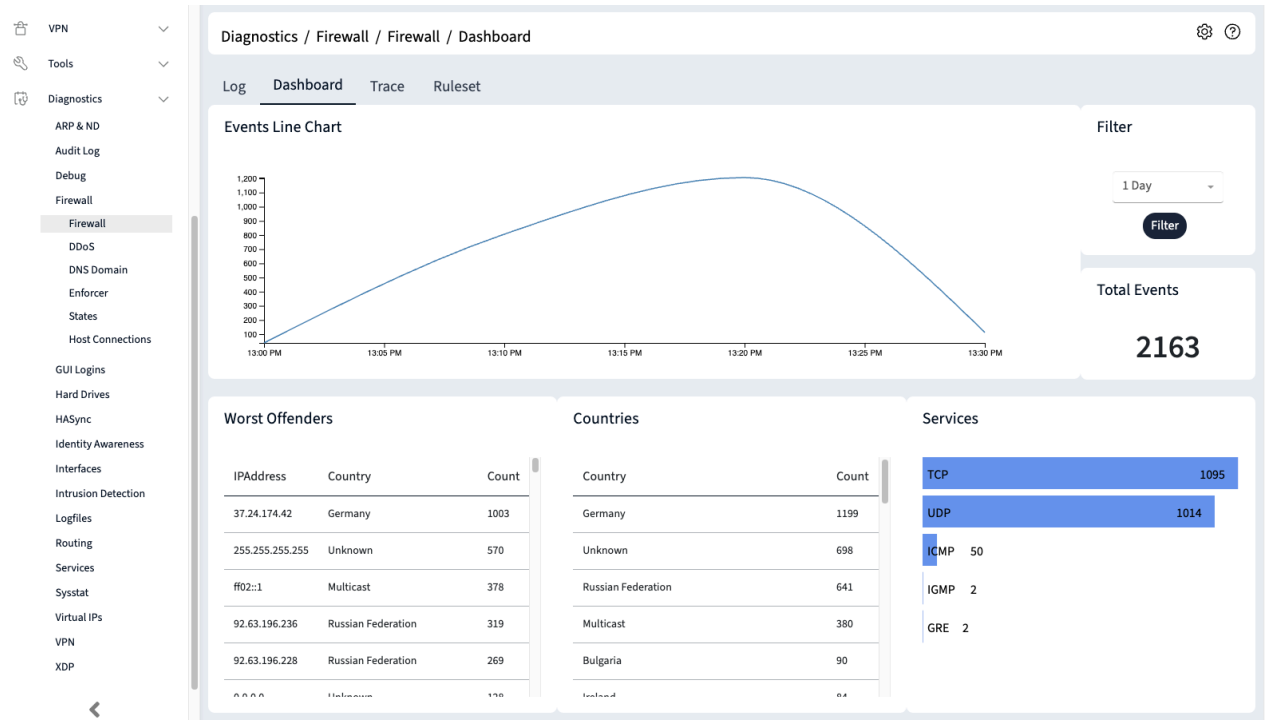


Figure 307: Dashboard

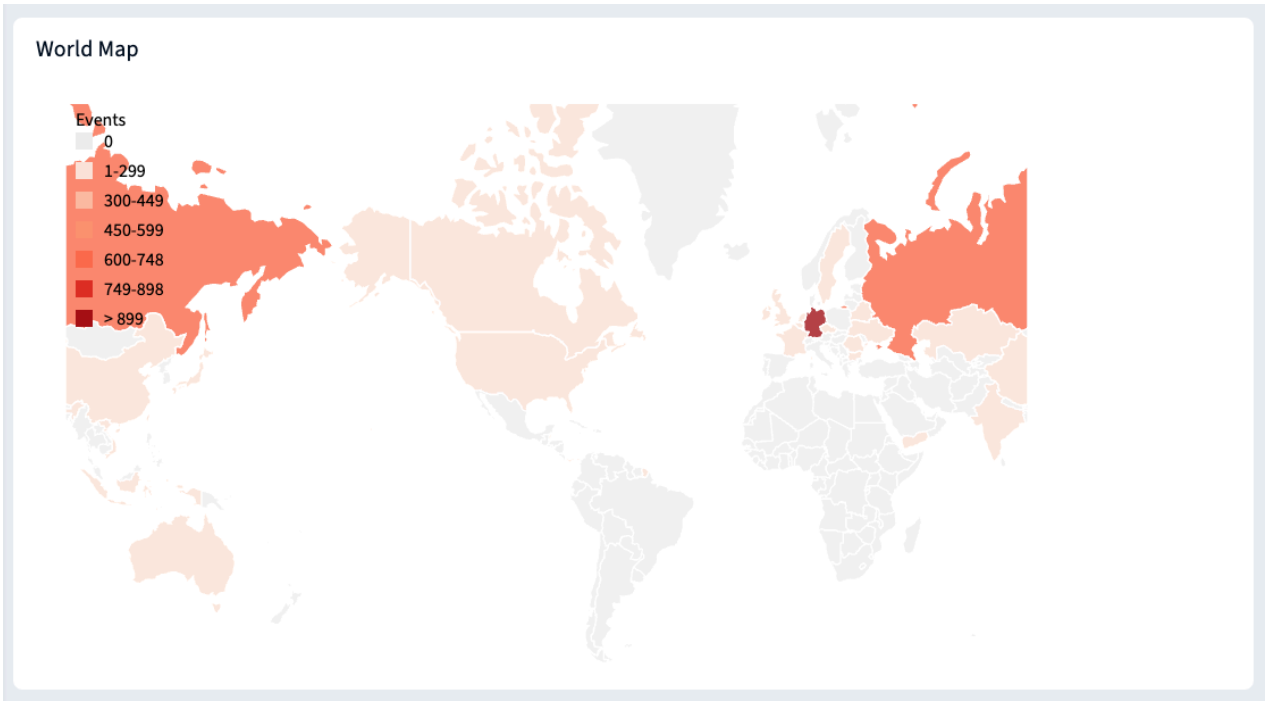


Figure 308: World map

Note: To see data in the dashboard, enable log file analysis. TBF disables it by default as it costs performance. Enable it in the *settings* on page 63.

Ruleset

The **Ruleset** tab shows you the current firewall rules in the system.



Figure 309: Ruleset

Trace

If you enable the trace setting on 1 or more firewall rules, you see matching traffic here. This is a good tool for debugging firewall rules.

The system follows the packet through the firewall from DNAT to rule until SNAT.

Gateways

This page displays gateway status and monitoring information.

Find the gateway diagnostics under **Diagnostics > Routing > Gateways**.

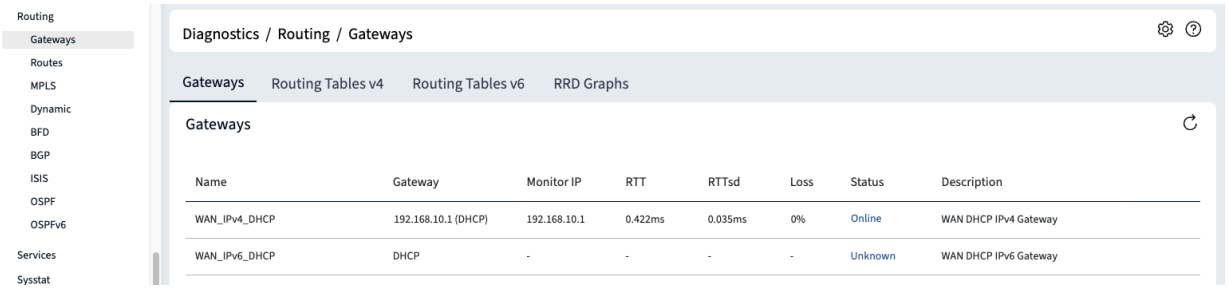


Figure 310: Gateways

Here you see an overview of all TBF gateways and their monitoring status (if set up).

To create new gateways or edit existing ones, navigate to **Routing > Gateways**. For more information, see *gateway* on page 214.

Routing tables

2 additional tabs display **Routing Tables v4** and **Routing Tables v6**. These tabs show the routing tables and their associated gateways, organized by priority.

To create additional routing tables or add gateways to existing ones, navigate to **Routing > Routes > Routing Tables**. For more information, see the *routes settings* on page 209.

RRD graphs

The final tab is **RRD Graphs**. Here you see historical data for TBF gateway checks.

Select a specific gateway, from date, and to date. TBF then generates 3 graphs:

- latency
- stddev
- loss

GUI logins

This page displays GUI login attempts, both successful and unsuccessful.

Find the GUI logins diagnostics under **Diagnostics > GUI Logins**.

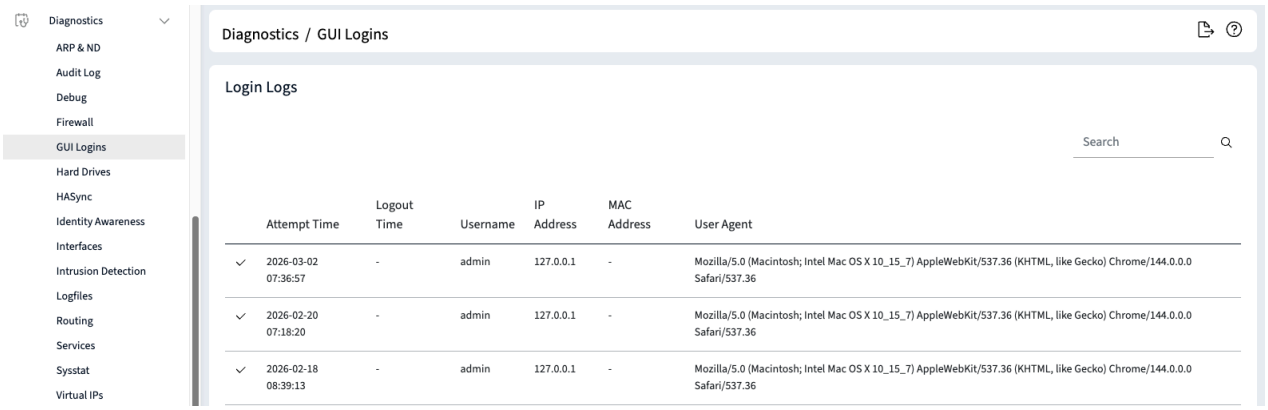


Figure 311: GUI logins

Here you see the TBF GUI login attempts, both successful and unsuccessful. Each login entry contains the following information:

- **Status**: Indicates whether the login was successful or unsuccessful
- **Attempt Time**: Shows when the login attempt occurred
- **Logout Time**: Records when the user logged out, if applicable
- **Username**: Displays the username used for the login attempt
- **IP Address**: Shows the IP address used for the login attempt
- **MAC Address**: Records the MAC address of the device used
- **User Agent**: Identifies the browser or application used for the login

Hard drives

This page displays hard drive health and status information.

Find the hard drive diagnostics under **Diagnostics > Hard Drives > Hard Drives**.

Note: This feature is not available on TBF 100/300 devices because of their eMMC drive type.

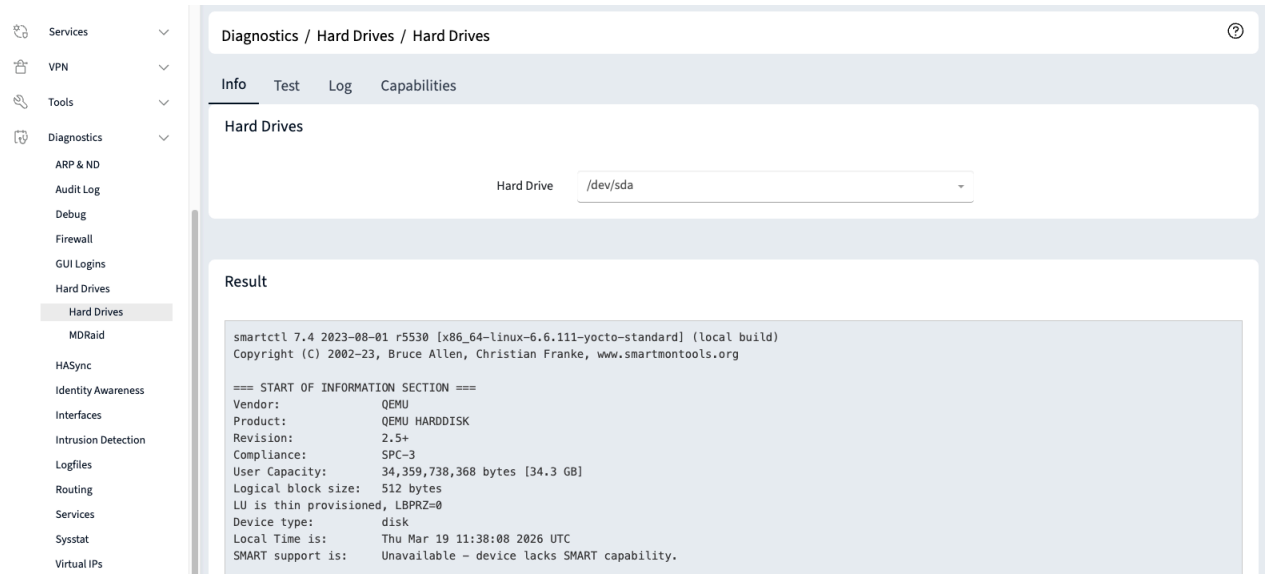


Figure 312: Hdd result

Here you see information for the hard drives.

TBF organizes the information into the following sections:

- **Info**
- **Test**
- **Log**
- **Capabilities**

The **Info** looks like this:

```
=== START OF INFORMATION SECTION ===
Device Model:     TS128GMTS400S
Serial Number:    7612325CE35121310830
Firmware Version: Q0926B
User Capacity:    128,035,676,160 bytes [128 GB]
Sector Size:      512 bytes logical/physical
Rotation Rate:    Solid State Device
Form Factor:      M.2
Device is:        Not in smartctl database [for details use: -P showall]
ATA Version is:   ACS-2 (minor revision not indicated)
SATA Version is:  SATA 3.1, 6.0 Gb/s (current: 3.0 Gb/s)
Local Time is:    Mon May 20 15:06:44 2019 CEST
SMART support is: Available - device has SMART capability.
SMART support is: Enabled
```

Under **Test**, run either a short or a long diagnostic test on a specific hard drive. The test results appear under **Log** after a few minutes.

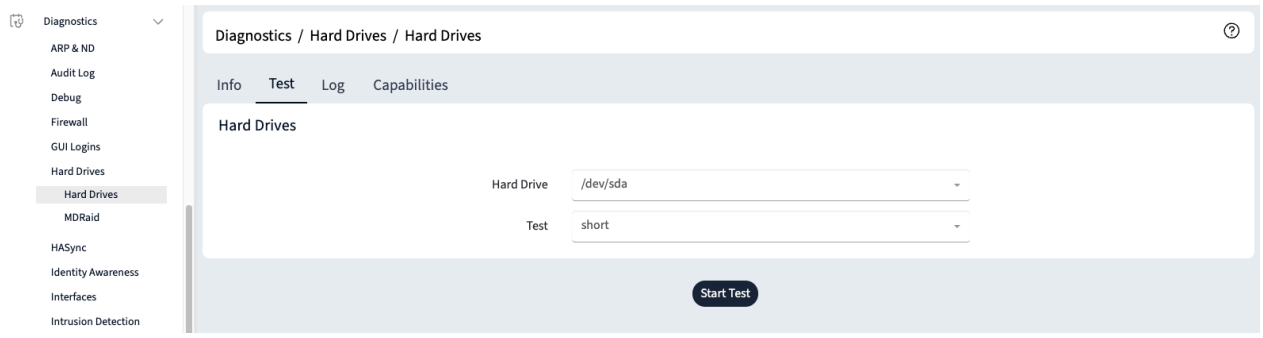


Figure 313: Test

The **Log** looks like this:

```
=== START OF READ SMART DATA SECTION ===
SMART Self-test log structure revision number 1
Num  Test_Description    Status                  Remaining  LifeTime(hours)  LBA_of_first_error
# 1  Short captive       Completed without error       00%      3327         -
# 2  Short offline       Completed without error       00%      3325         -
# 3  Extended offline    Completed without error       00%      3324         -
# 4  Short offline       Completed without error       00%      3324         -
# 5  Short offline       Completed without error       00%      3251         -
```

HASync

This page displays high availability synchronization status and last sync timestamps.

Find the HASync diagnostics under **Diagnostics > HASync**.

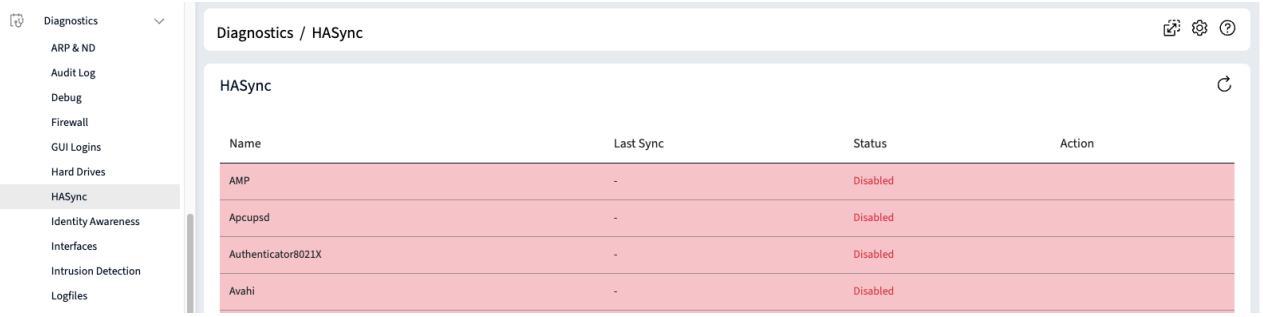


Figure 314: Hasync

This page contains an overview of the TBF HASync status. The page displays the HASync name, current status, and the date of the last synchronization.

Actions

Manually trigger a resync when HASync is enabled. To initiate an HASync for all enabled HASyncs, use the control located in the upper-right corner.

Note: Keep in mind that synchronization starts with a slight delay. Wait up to 1 minute to see the results.

Host connections

This page displays open network connections to the firewall.

Find the host connections diagnostics under **Diagnostics > Firewall > Host Connections**.

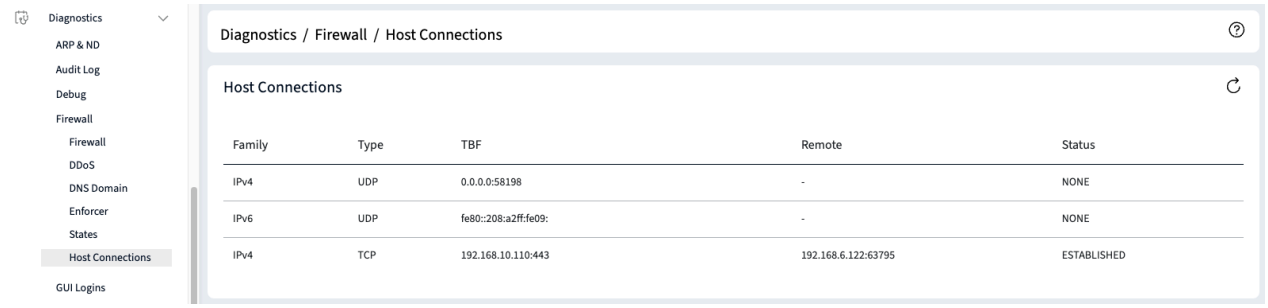


Figure 315: Host connections

Here you see the open connections to TBF. Each connection entry contains the following details:

- IP address family
- Type
- TBF IP address
- Remote IP address
- Status

Identity awareness

This page displays identity awareness IP address assignments by category.

Find the identity awareness diagnostics under **Diagnostics > Identity Awareness**.

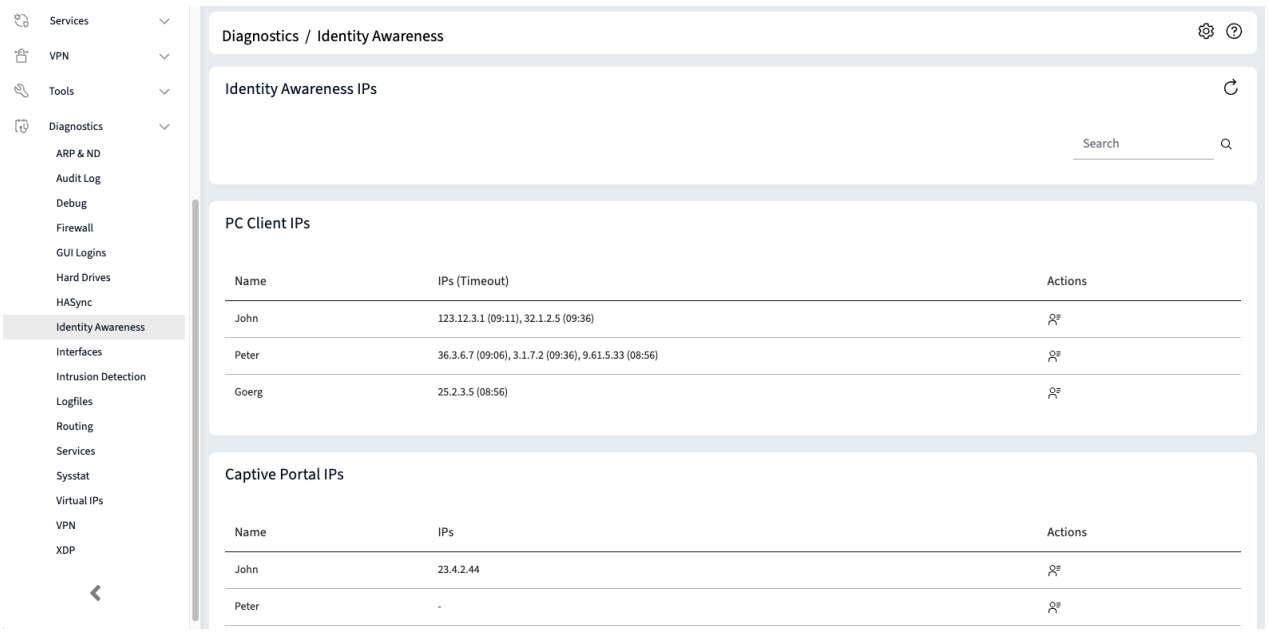


Figure 316: Identity awareness

Here you see the identity awareness IP addresses organized by category:

- PC client IP addresses
- Captive portal IP addresses
- AD IP addresses
- DHCP IP addresses

To change the identity awareness setup, navigate to **Authentication > Identity Awareness**. For more information, see *identity awareness* on page 405.

Interfaces

This page displays interface status, addresses, MTU, and traffic statistics.

Find the interface diagnostics under **Diagnostics > Interfaces > Interfaces**.

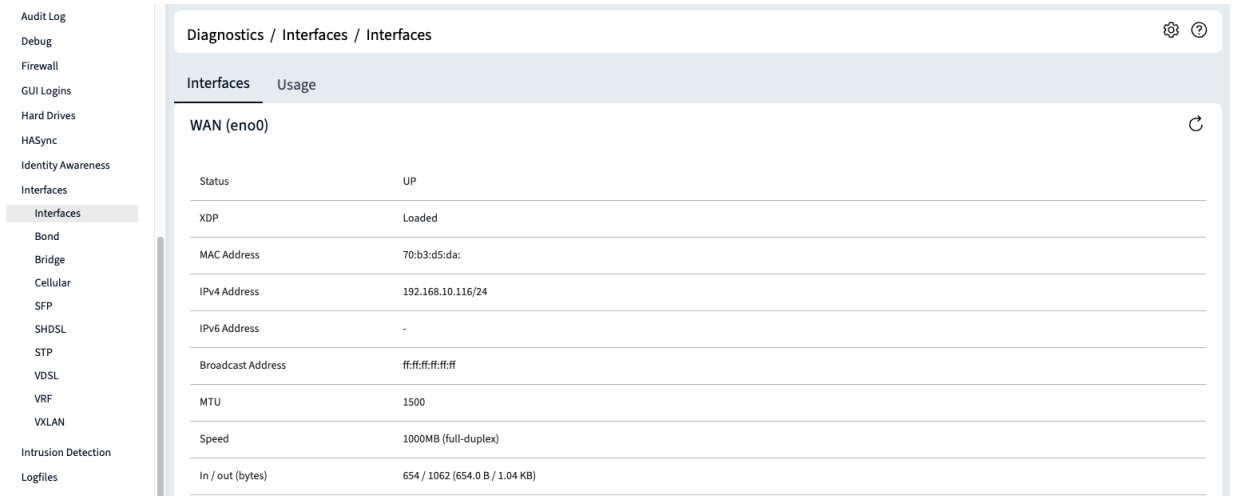


Figure 317: Interfaces

Here you see each interface along with its current status, MAC address, IPv4 address, IPv6 address, broadcast address, MTU, and traffic statistics.

TBF categorizes traffic data into the following categories:

- In (incoming) traffic
- Out (outgoing) traffic
- Further breakdown by bytes, packets, errors, and dropped packets

To assign interfaces, navigate to **Interfaces > Assign**. For more information, see *assign interfaces* on page 91.

Usage

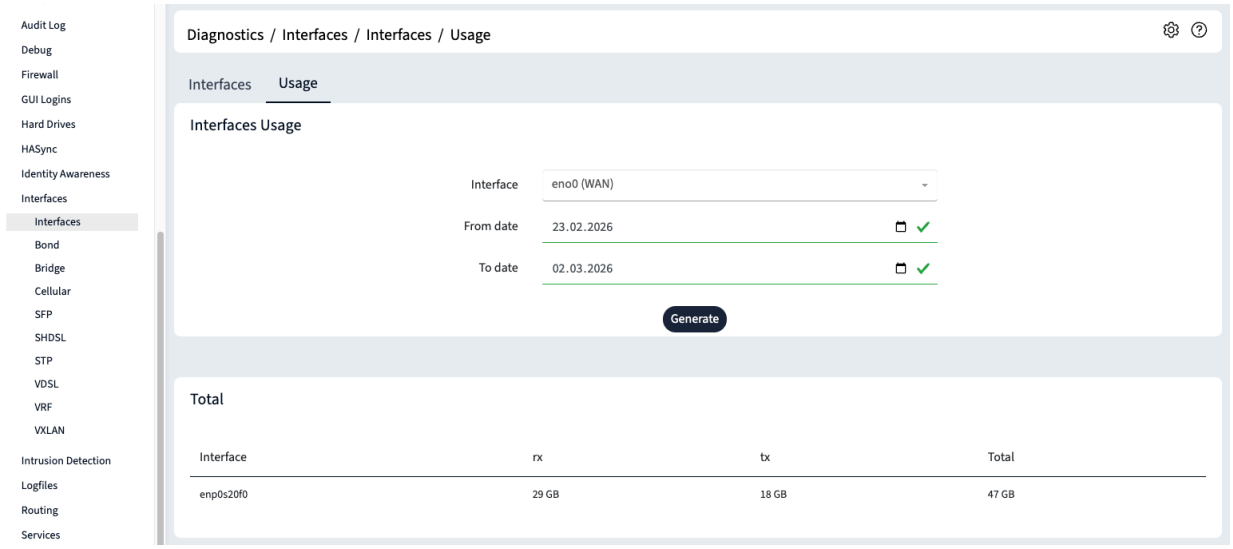


Figure 318: Interfaces usage

Here you see the usage statistics for each interface. To generate usage data:

1. Select an **Interface**.
2. Select a **From date** and a **To date**.
3. To see the usage data, select **Generate**.

The usage data looks like this:

```
Database updated: 2021-12-08 15:35:00

enp0s20f0 since 2021-12-08

        rx:  5.50 MiB      tx:  1.39 MiB      total:  6.90 MiB

monthly
                     rx      |     tx      |    total    |   avg. rate
    ------------------------+-------------+-------------+---------------
    2021-12      5.50 MiB |    1.39 MiB |    6.90 MiB |       87 bit/s
    ------------------------+-------------+-------------+---------------
    estimated     20.43 MiB |    5.11 MiB |   25.54 MiB |

daily
                     rx      |     tx      |    total    |   avg. rate
    ------------------------+-------------+-------------+---------------
        today      5.50 MiB |    1.39 MiB |    6.90 MiB |    1.03 kbit/s
    ------------------------+-------------+-------------+---------------
    estimated      8.48 MiB |    2.14 MiB |   10.62 MiB |
```

Intrusion detection

This page displays intrusion detection memory usage, packet statistics, alerts, and security logs.

The intrusion detection diagnostics have 3 tabs:

- The **Overview** tab displays general information on memory usage and packet statistics.
- The **Dashboard** tab offers an interactive view of the collected data from alerts and block events.
- The **Security Logs** tab shows fast log matches for drop or reject events.

Note: The IDS engine drops packets if they are too broken. In such cases, no rule ID appears in the drop message.

The **Audit Logs** tab shows fast log matches for audit events.

The **Event Log** displays detailed information for each matched flow, including protocol and application data that the system associates with a flow or event. When you hover over source and destination IP addresses, reverse DNS entries appear if TBF can resolve the IP address.

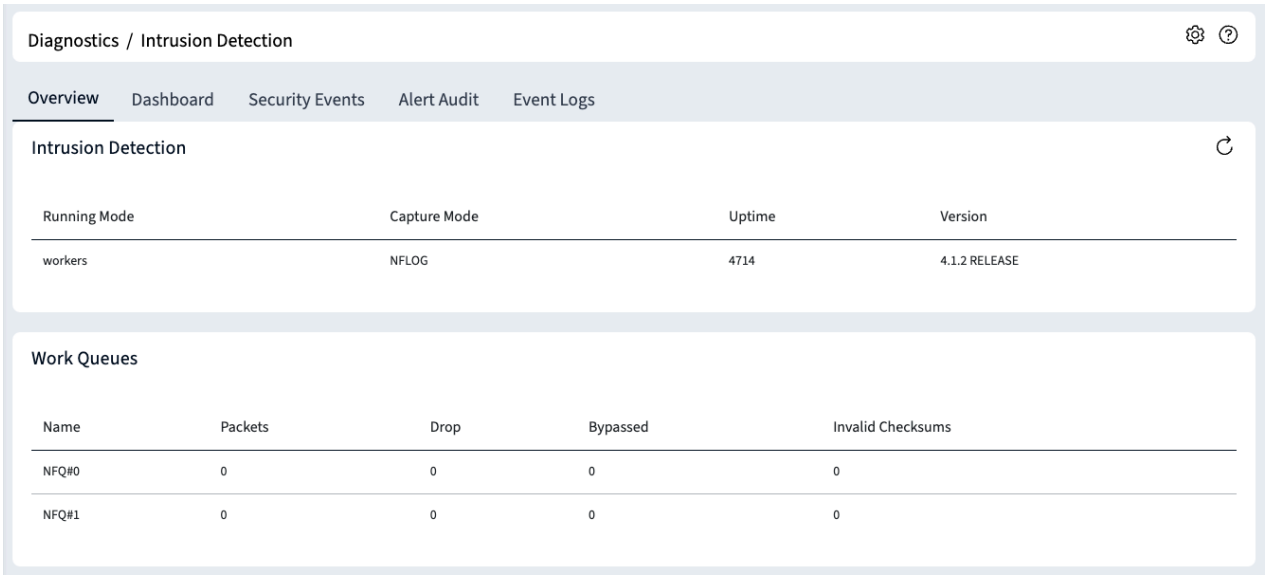


Figure 319: Intrusion detection

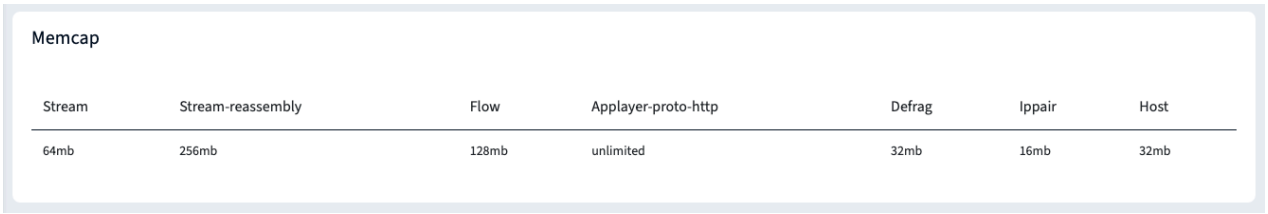


Figure 320: Memcap

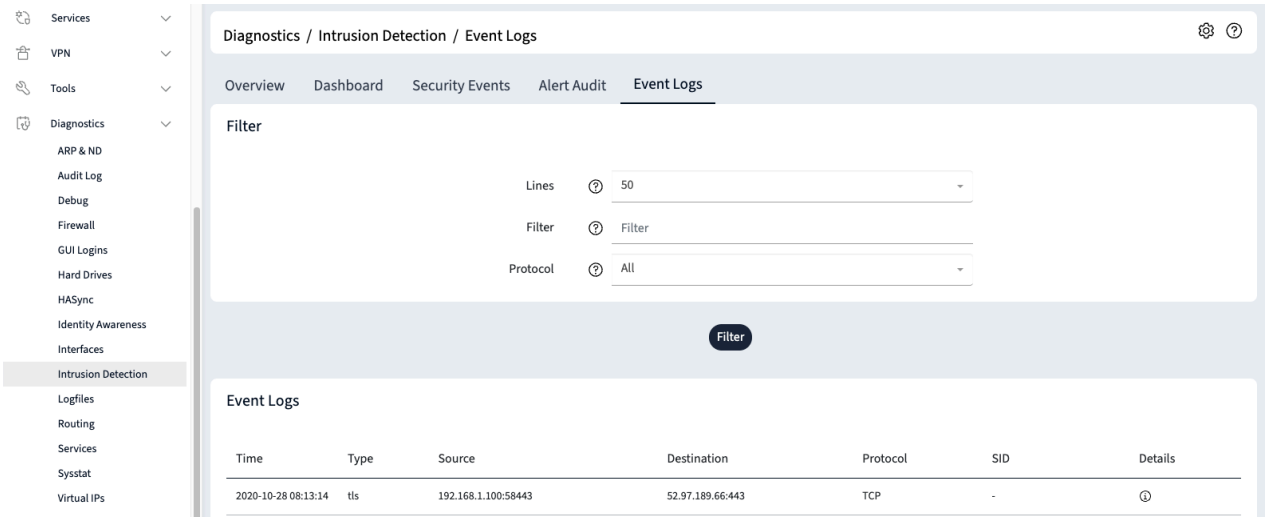


Figure 321: Event logs

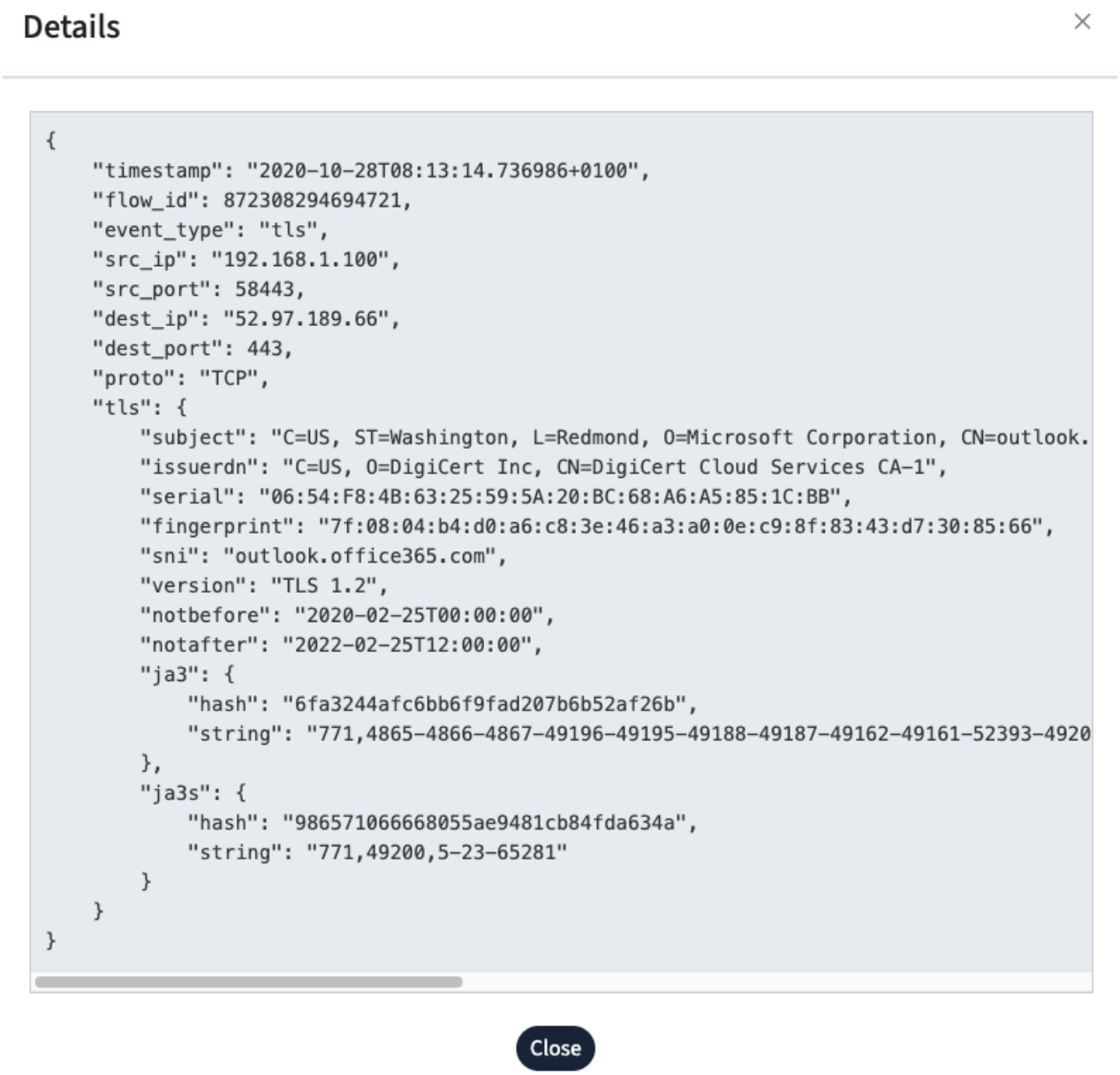


Figure 322: Details

Intrusion detection dashboard

The **Dashboard** tab shows you alert and block information organized by country and origin.

The dashboard is a convenient way to visualize event data.

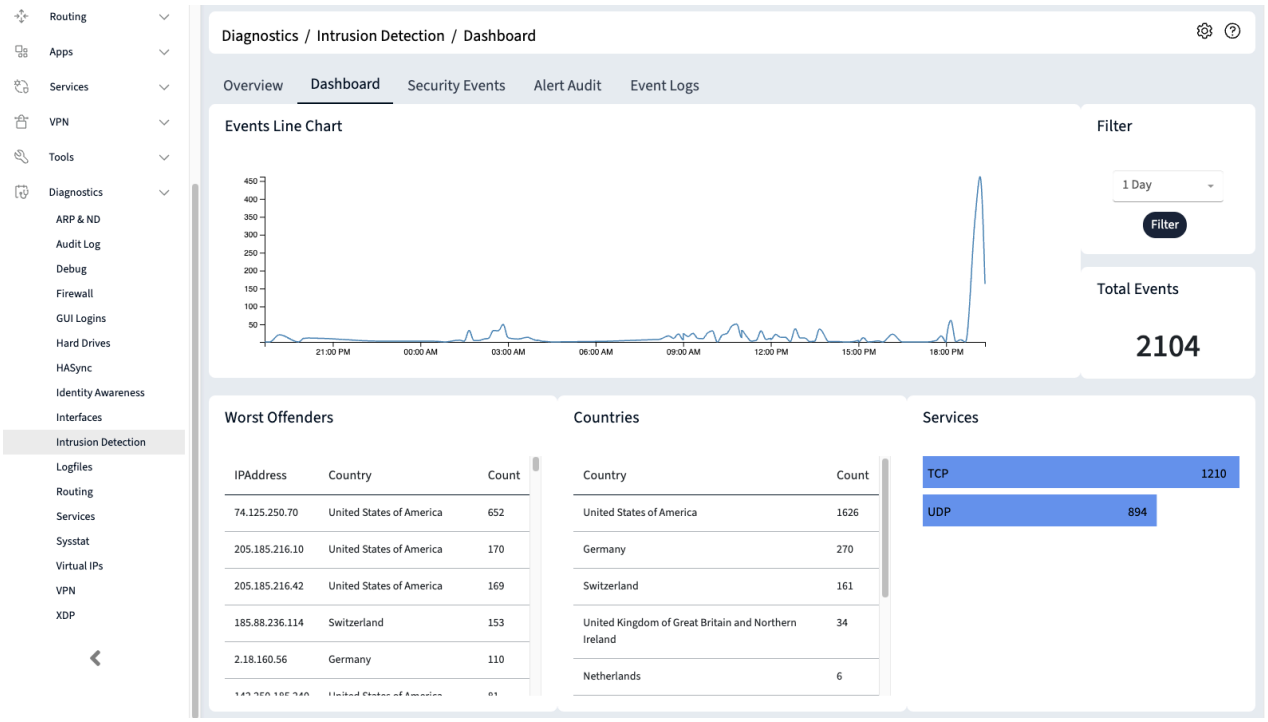


Figure 323: Dashboard

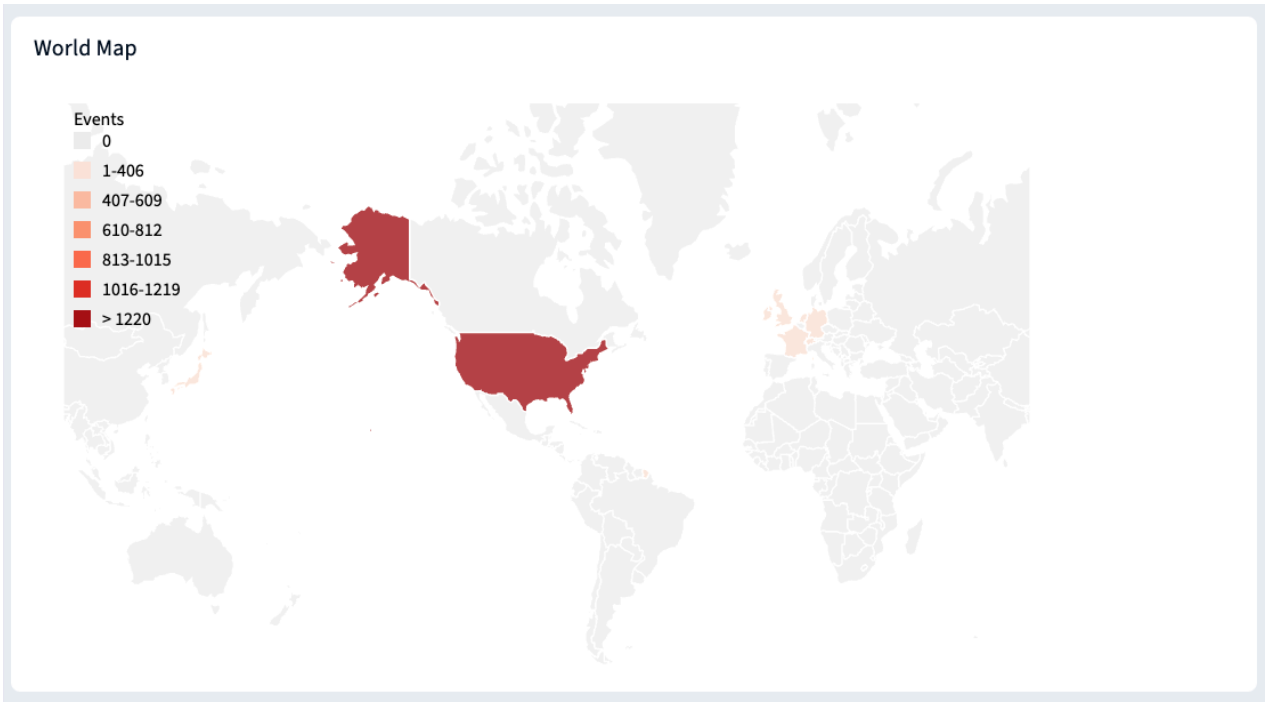


Figure 324: World map

Note: To view data in the dashboard, enable log file analysis. TBF disables it by default because it affects performance. Enable it in the *settings* on page 63.

IPsec

This page displays IPsec tunnel connection status and security association details.

Find the IPsec diagnostics under **Diagnostics > VPN > IPSec**.

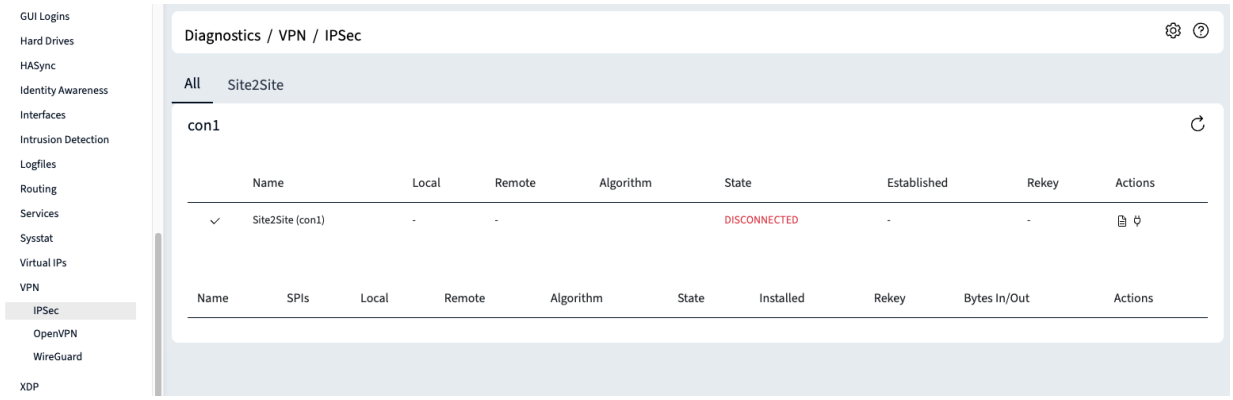


Figure 325: IPSEC

Here you see connection information on the IPsec setup. There is the **All** tab that shows the IPsec setups, and each IPsec entry has its own dedicated tab.

TBF organizes the connection information into 2 main sections:

- Phase1
- Phase2

Phase1

The first column shows whether the entry is enabled or disabled.

Local and **Remote** display their respective IP addresses.

Reauth shows when the system is scheduled to perform the next reauthentication.

Algo lists the encryption algorithms in use.

State is *Connected* or *Disconnected*.

Under **Actions**, you can manually start the connection or disconnect it, depending on the current status. There is also a **Log** icon. To open a window containing detailed log information for the specific connection, select the **Log** icon.

Phase2

The system lists the phase2 setup here.

Local and **Remote** display their respective IP addresses.

SPIs shows a security parameter index, which serves as an identification tag that is helpful for the kernel.

Rekey shows when the system negotiates a new key.

Algo displays the encryption algorithm in use.

Bytes In/Out shows the traffic.

Under **Actions**, you can manually disconnect the connection. There is also a **Log** icon. To open a window containing detailed log information for the specific connection, select the **Log** icon.

To change the IPsec setup, navigate to **VPN > IPSec** and see *IPsec phase 1* on page 324.

IKEv2

There is 1 aspect that affects IKEv2 behavior. The system derives the encryption keys for the CHILD_SA that it implicitly creates during the IKE_AUTH exchange from the IKE keys, even if you set up PFS.

Therefore, if the communicating peers have different configurations regarding PFS usage (or different Diffie-Hellman groups), the system does not detect this mismatch until the CHILD_SA undergoes its first rekeying process with a CREATE_CHILD_SA exchange, which is then unsuccessful.

This behavior also explains why you do not see a Diffie-Hellman group displayed in the Phase2 status until the SA undergoes its first rekeying process.

ISIS

This page displays Intermediate System to Intermediate System (IS-IS) routing information.

Find the ISIS diagnostics under **Diagnostics > Routing > ISIS**.

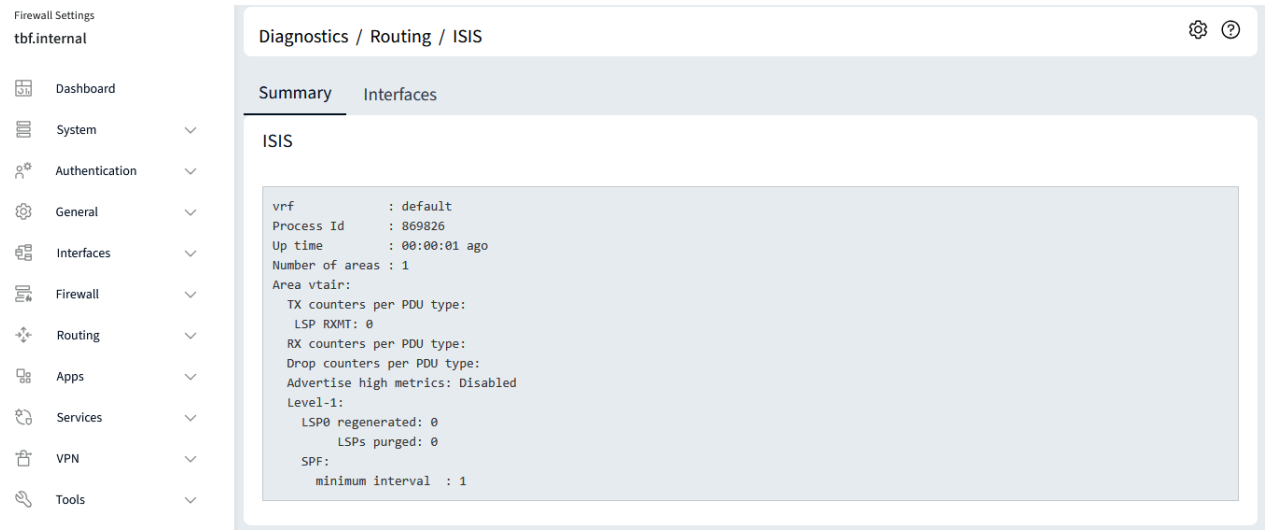


Figure 326: ISIS summary

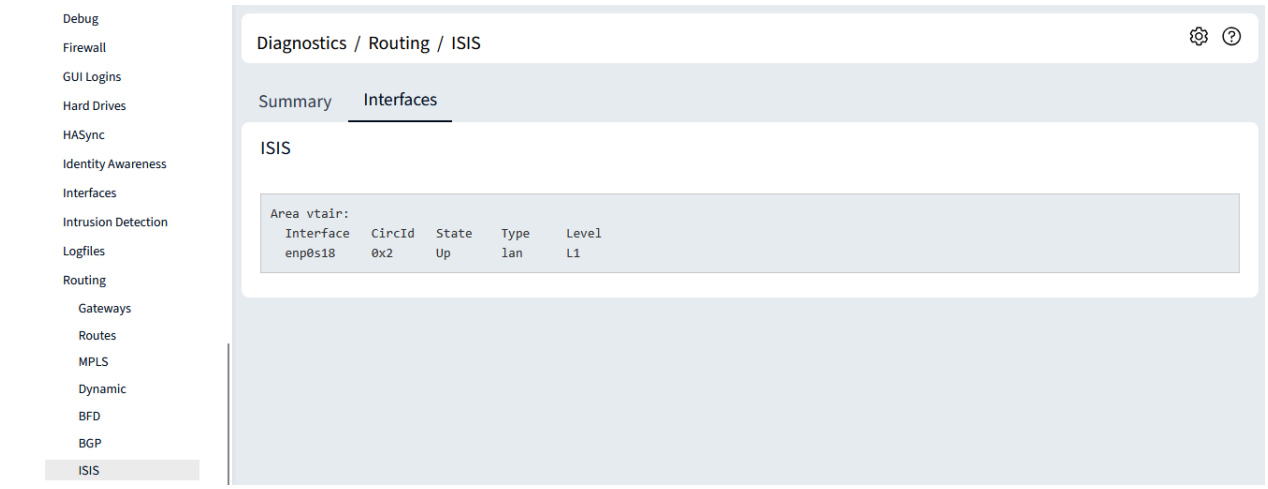


Figure 327: ISIS interfaces

Here you see information about the Intermediate System to Intermediate System (ISIS) setup.

There are 2 tabs:

- ISIS **Summary**: Shows an overview of the ISIS configuration
- ISIS **Interfaces**: Shows detailed interface-specific information

Note: TBF automatically refreshes the information at regular intervals without requiring user input.

To change the ISIS setup, navigate to **Routing > ISIS**. For more information, see the *ISIS settings* on page 227.

Logging

This page displays system logs organized by service or category.

Find the logging diagnostics under **Diagnostics > Logfiles**.

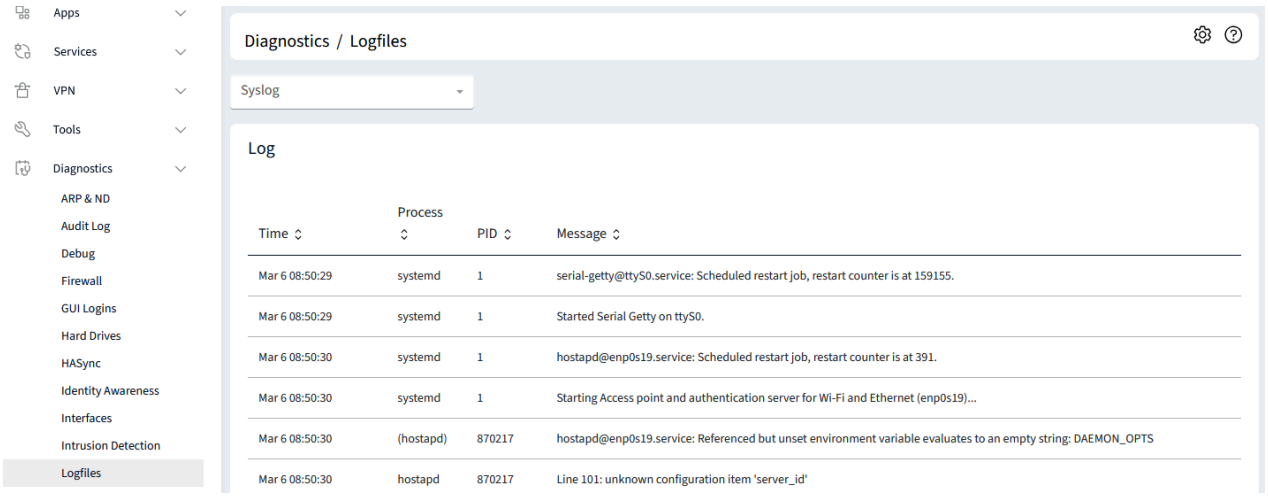


Figure 328: Logfiles

Here you see the system logs organized by their respective service or category.

By default, the system displays the most recent 50 entries from the log file. If you select **Realtime**, you see the latest log entries as the system writes them to the log file in real time.

Settings

In the general settings, specify the **Maximum Logfile Size in MB** for each file. The default is `5` MB.

Tip: The number should not be too high, or it consumes excessive hard drive space.

The **Logfile Rotation** specifies the number of log files the system retains before it deletes older ones.

Enable **Enable Auditlog Syslog**, which records audit events in the `audit.log` file.

Under **GUI Logging**, adjust the logging detail level for each TBF category. This only impacts the TBF GUI and management logs.

Additionally, you can enable **Enable Remote Logging**. This feature adds multiple syslog servers by specifying their:

- IP address
- Port
- Protocol

For TLS connections, the system verifies the **Certificate Authority** to use for checking the TLS connection to the syslog server.

Cellular

This page displays cellular modem reception and system status information.

Note: This section contains screenshots from the former VT Air GUI. These will be updated in a future version.

Find the cellular diagnostics under **Diagnostics > Interfaces > Cellular**.

Here you see the settings and the status of the cellular setup.

Note: To change the cellular setup, navigate to the interface page of the cellular under **Interfaces**.

Reception shows data on the signal reception of the cellular modem from the cell tower.

System Status shows the current operational status of the cellular modem.

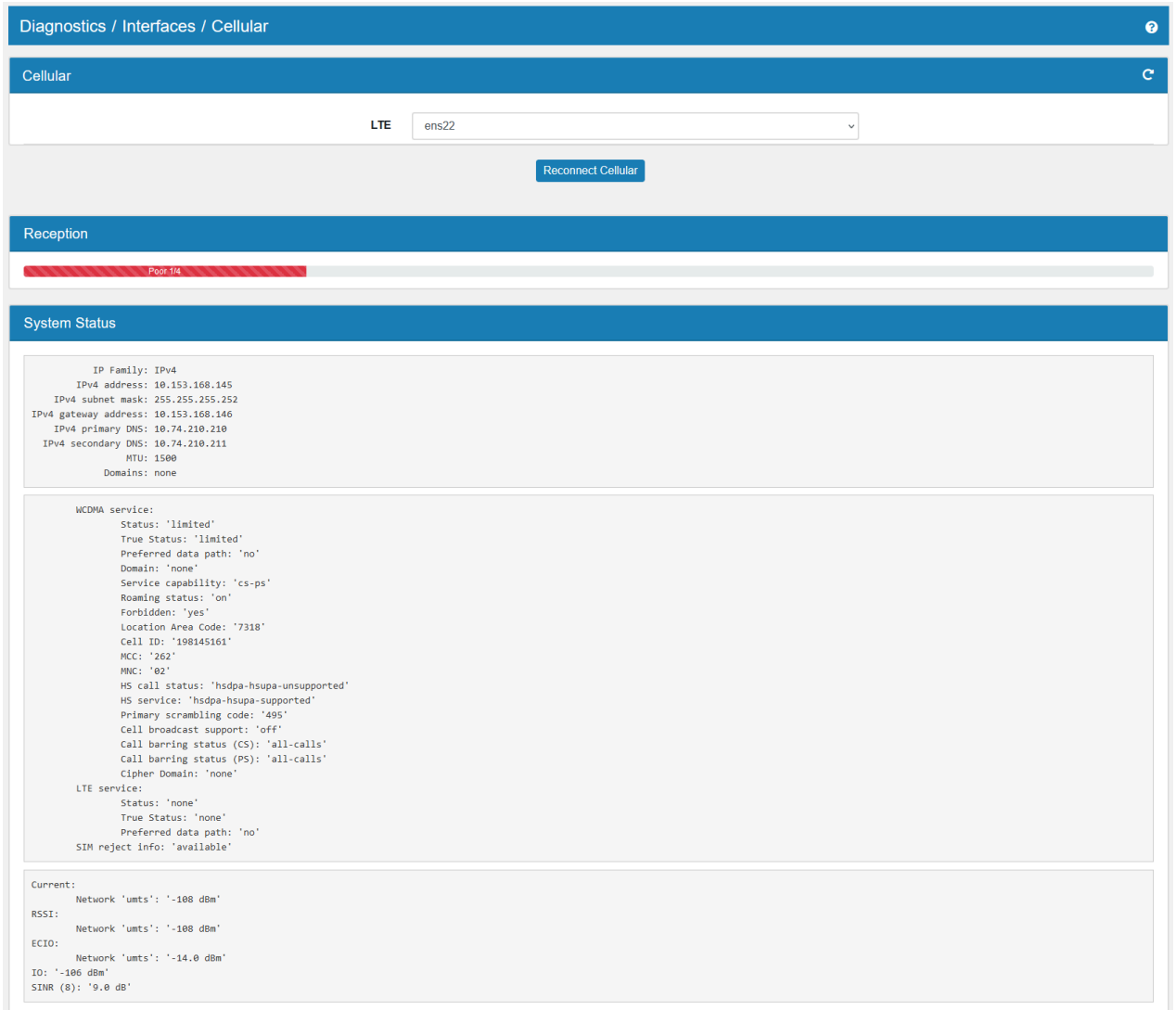


Figure 329: Reception and system status

SIM Status shows the status of the SIM card and indicates whether it is locked.

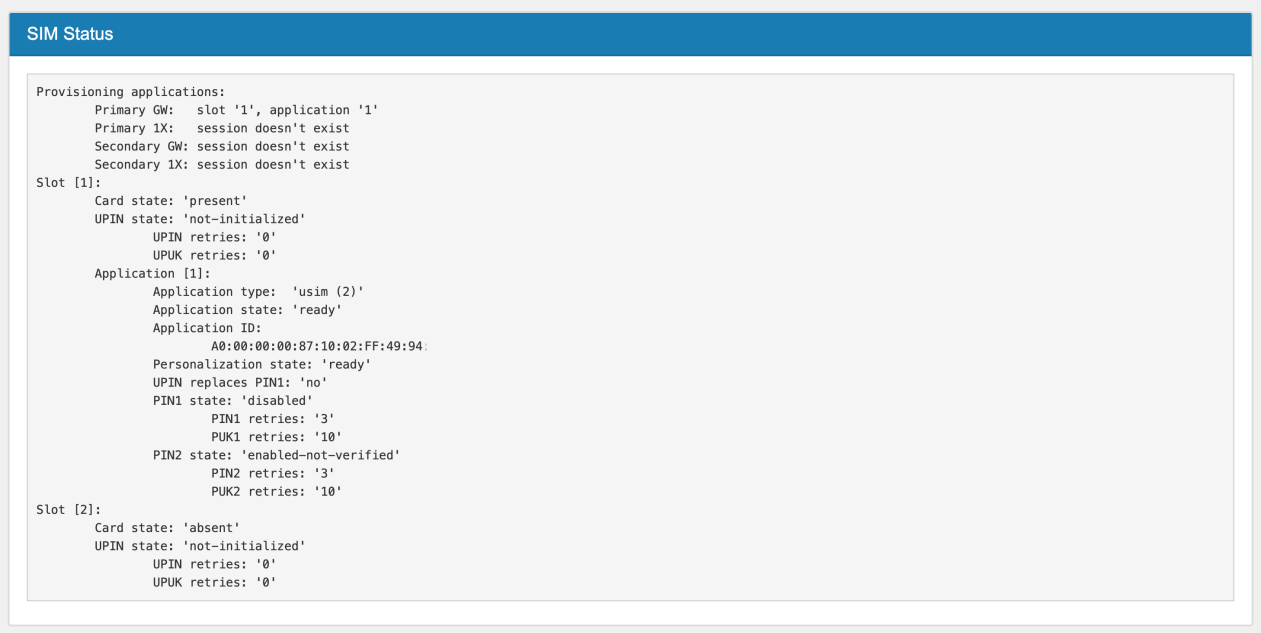


Figure 330: Sim status

Channel Rates displays both the theoretical maximum speed and the actual speed that the cellular modem negotiated with the service provider.

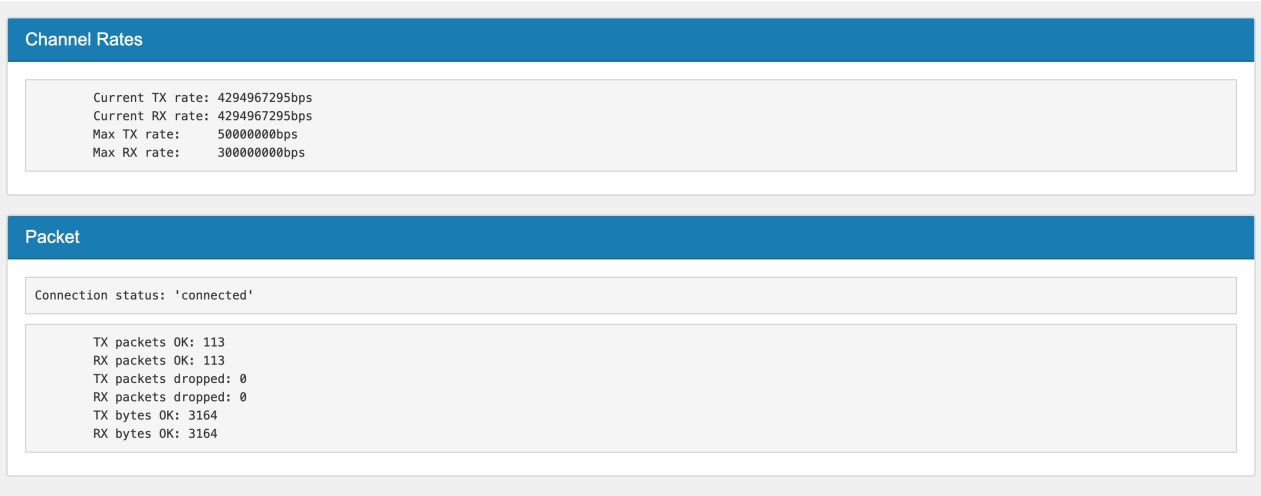


Figure 331: Channel rates and packet

Packet shows the statistics on the currently active cellular connection.

```
Current:
        Network 'lte': '-77 dBm'
RSSI:
        Network 'lte': '-77 dBm'
ECIO:
        Network 'lte': '-2.5 dBm'
IO: '-106 dBm'
SINR (8): '9.0 dB'
RSRQ:
        Network 'lte': '-10 dB'
SNR:
        Network 'lte': '6.6 dB'
RSRP:
        Network 'lte': '-106 dBm'
```

Figure 332: Connection statistics

Cell Location Info displays information on the cellular tower location.

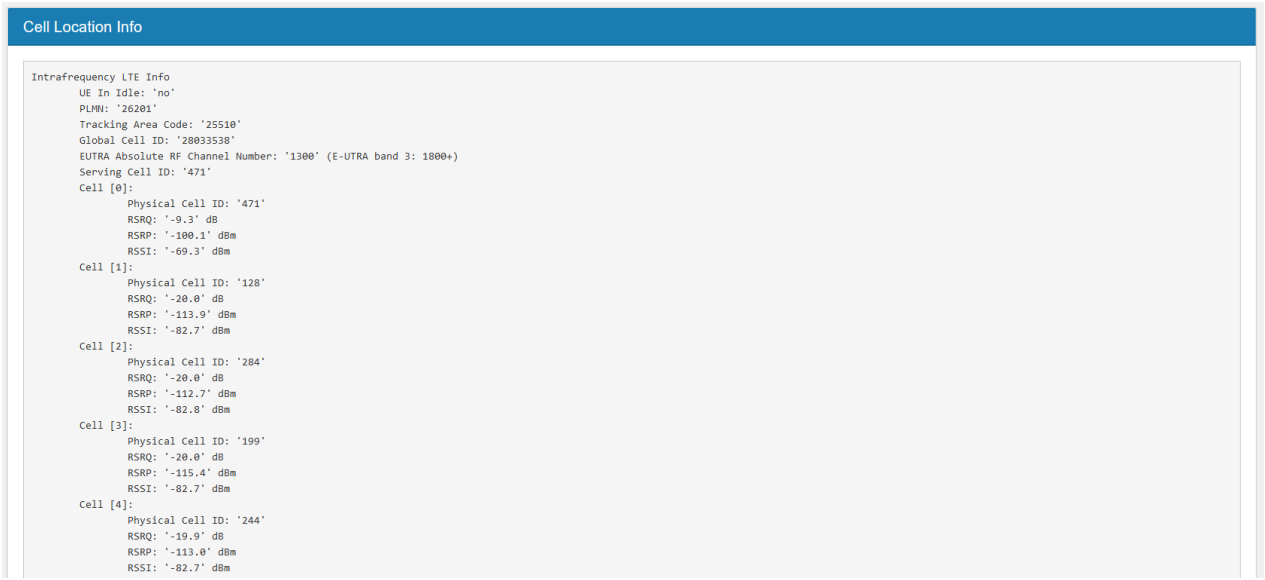


Figure 333: Cell location information

LTE RSSI

Signal quality	RSSI
Excellent	> -65
Good	-65 .. -75
Fair	-75 .. -85
Poor	< -85

LTE SINR

Signal quality	SINR (dB)
Excellent	>12.5
Good	10 .. 12.5
Fair	7 .. 10
Poor	<7

MDRaid

This page displays software RAID status, disk information, and partition details.

Find the MDRaid diagnostics under **Diagnostics > Hard Drives > MDRaid**.

TBF automatically uses MDRaid as the software RAID solution when you have 2 identical drives in the device during installation. To receive email notifications about detected drive failures, set up the email notification settings in *notifications* on page 69.

Info

This tab displays the information about MDRaid disks and partitions and displays an overview of the current functionality.

Disks

This tab helps you examine each disk and disk array in detail. View the available partitions in each disk array, monitor their health status, and check the overall status of the disk array.

When a failure is detected, you can mark partitions as inoperable or add a new disk to the array.

Copy disk (repair)

This tab is useful if you need to replace an inoperable disk. The replacement disk needs the exact disk layout as the original disk.

This tool helps you copy the disk layout from the working disk to the replacement disk.

Replace a disk

In the case of a detected disk failure, replace the inoperable drive. Go to **Disks** and confirm that the partitions on the inoperable drive are marked as *inoperable* in the MDRaid arrays.

While still in **Disks**, remove the inoperable partitions from the MDRaid arrays.

The MDRaid arrays now properly exclude the inoperable drive.

Note: Now shut down TBF and replace the disk with a new one. In the case of a hot swap, perform this operation without shutting down TBF.

Now go to **Copy Disk (Repair)** and copy the partition table from the working disk to the new disk. Once this operation completes successfully, go back to **Disks**.

Now add the corresponding partition on the new disk back to each MDRaid array.

After you successfully complete the addition, a resync operation starts automatically to synchronize the data to the new disk.

MPLS

This page displays MPLS and LDP status information.

Find the MPLS diagnostics under **Diagnostics > Routing > MPLS**.

Here you have an overview of the TBF MPLS and LDP diagnostics. The MPLS table shows them.

You see the labels of each route, the gateway it uses, and whether it uses a gateway. The interface also shows the MPLS.

To create MPLS routes, navigate to **Routing > MPLS** and see the *MPLS settings* on page 218.

LDP

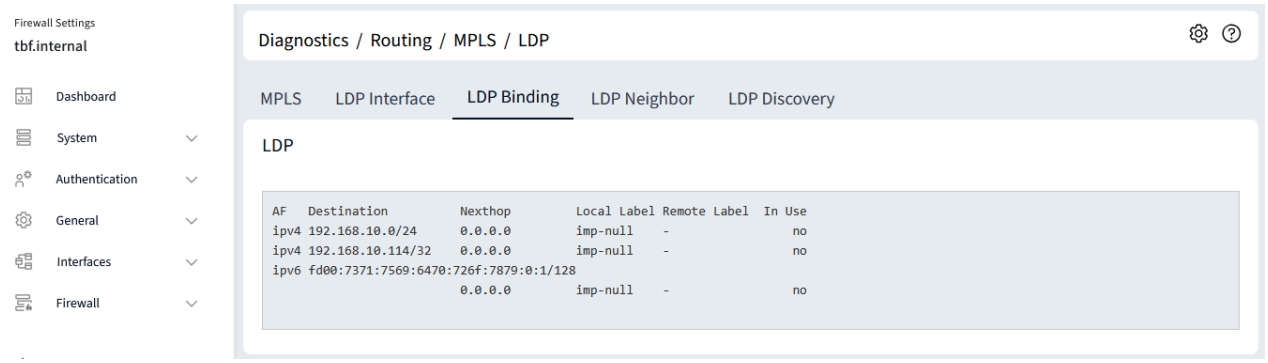


Figure 334: Ldp

Here you have an overview of the 4 LDP diagnostic categories:

- LDP interface
- LDP binding
- LDP neighbor
- LDP discovery

NTP

This page displays NTP peer status and synchronization information.

Find the NTP diagnostics under **Diagnostics > Services > NTP**.

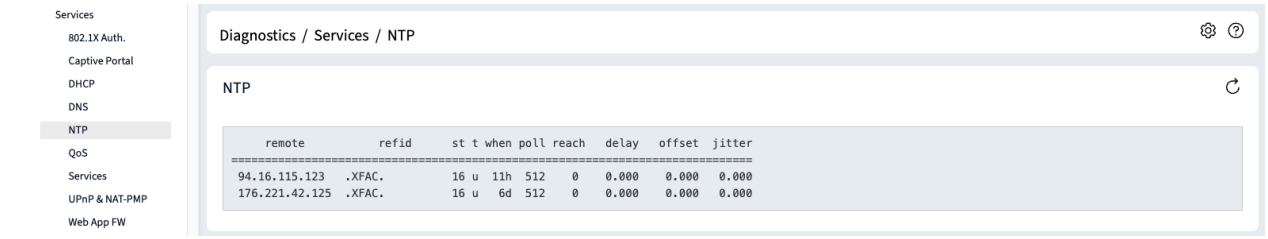


Figure 335: NTP

To change the NTP setup, navigate to **Services > NTP**. For more information, see *NTP* on page 294.

The output looks like this:

```
remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
94.16.115.123   .XFAC.          16 u  11h  512    0    0.000    0.000   0.000
176.221.42.125  .XFAC.          16 u   6d  512    0    0.000    0.000   0.000
```

OpenVPN

This page displays OpenVPN server and client connection status.

Find the OpenVPN diagnostics under **Diagnostics > VPN > OpenVPN**.

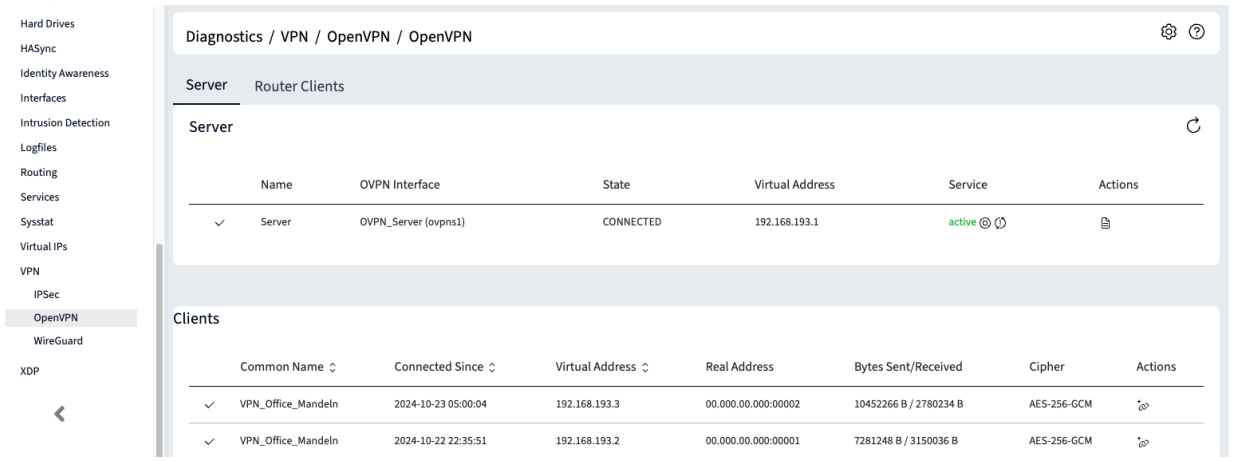


Figure 336: OpenVPN server

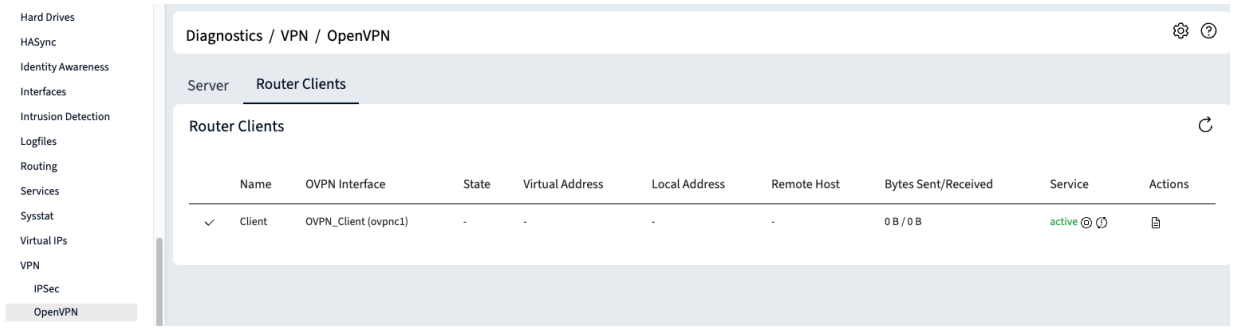


Figure 337: OpenVPN router clients

Here you see information about the OpenVPN setup. The main page lists all **OpenVPN Router Clients**, while each **OpenVPN Server** has its own dedicated page.

To change the OpenVPN setup, navigate to **VPN > OpenVPN**. For more information, see *OpenVPN general* on page 339.

Router clients

This page lists router clients set up in TBF.

The **State** shows whether a connection is currently established.

The **Local Address** and the **Virtual Address** display more information about each client.

The **Remote Host** displays the OpenVPN server that the client connects to.

If a connection succeeds, **Bytes Sent/Received** shows the traffic.

In the **Service** column, you can start, stop, or reload the service based on its current state.

The **Actions** column has a **Log** icon. To open a window with detailed log information on the client, select the **Log** icon.

Servers

This page lists servers set up in TBF.

The **State** shows whether a connection is currently established.

The **Virtual Address** displays the current IP address of the server.

In the **Service** column, you can start, stop, or reload the service based on its current state.

The next table shows clients that connect to the server.

Under **Connected Since**, you see whether they connect or not.

Real address shows their IP address.

Bytes Sent/Received displays the traffic.

Cipher shows which encryption algorithm the system uses.

The **Actions** help you disconnect clients manually.

There is also a **Log** icon. To open a window with detailed log information on the server, select the **Log** icon.

OSPF

This page displays Open Shortest Path First (OSPF) routing information.

Find the OSPF diagnostics under **Diagnostics > Routing > OSPF**.

Here you see information about the Open Shortest Path First (OSPF) setup. TBF divides it into:

- **Summary**
- **Routes**
- **Neighbors**
- **Database**
- **Interfaces**

Note: TBF automatically refreshes the information on this page at regular intervals without requiring any user input.

To change the OSPF setup, navigate to **Routing > OSPF**. For more information, see the *OSPF settings* on page 229.

OSPFv6

This page displays OSPFv3 routing information for IPv6 networks.

Find the OSPFv6 diagnostics under **Diagnostics > Routing > OSPFv6**.

Here you see information about the Open Shortest Path First v6 (OSPFv6) setup. TBF divides it into:

- **Summary**
- **Routes**
- **Neighbors**
- **Database**
- **Interfaces**

Note: TBF automatically refreshes the information on this page at regular intervals without requiring any user input.

To change the OSPFv6 setup, navigate to **Routing > OSPFv6**. For more information, see the *OSPFv6 settings* on page 233.

QoS

This page displays Quality of Service (QoS) settings and queue status.

Find the QoS diagnostics under **Diagnostics > Services > QoS**.

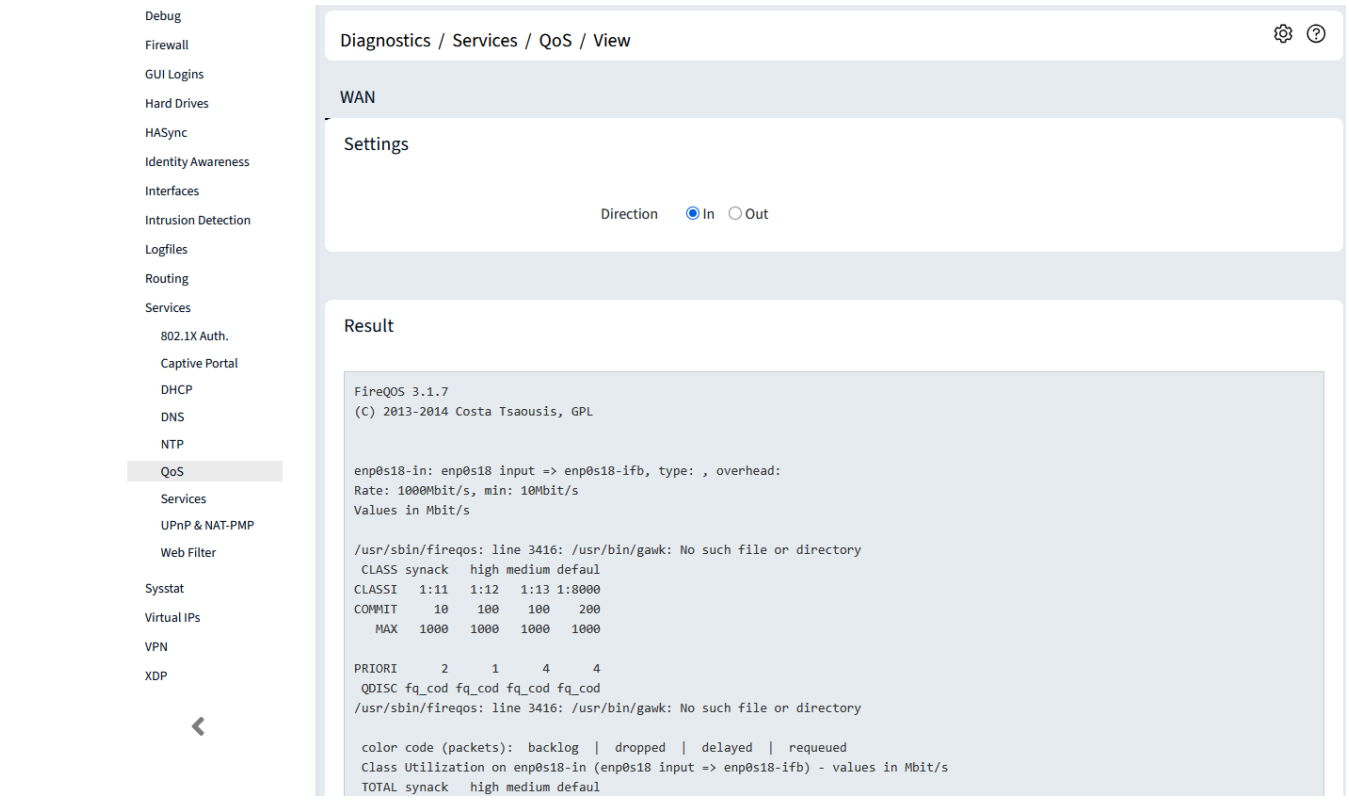


Figure 338: QOS result

Here you see the settings and status of the quality of service (QoS) setup. TBF organizes the information by interface and categorizes it as either *input* or *output*. You can view only 1 category at a time.

To change the QoS setup, navigate to **Services > QoS**. For more information, see *QoS* on page 296.

Routes

This page displays the routing table for IPv4 and IPv6.

Find the route diagnostics under **Diagnostics > Routing > Routes**.

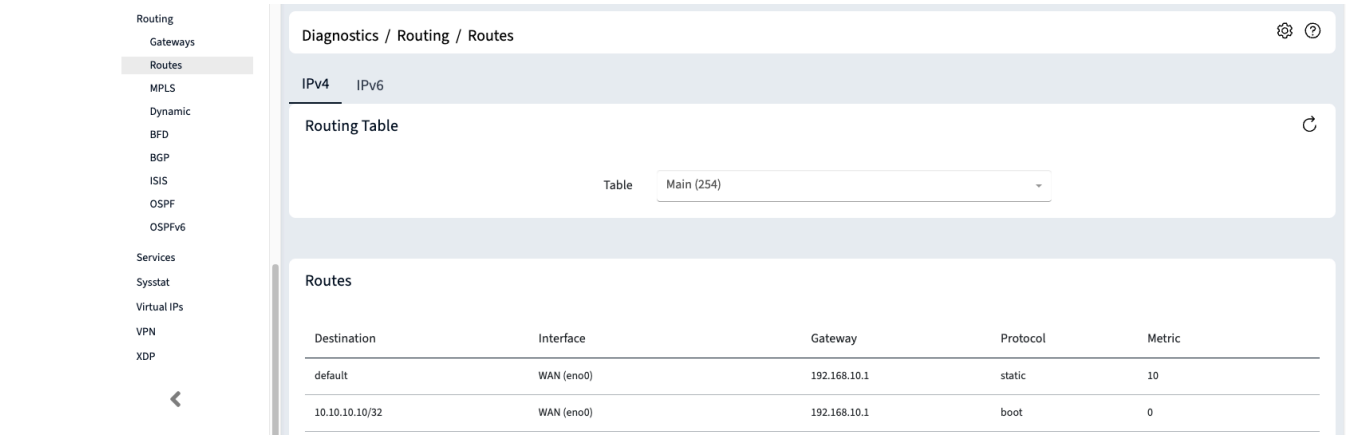


Figure 339: Routes

Here you see the TBF routes, organized by IPv4 nad IPv6. The **Routing Table** displays only the routes assigned to a specific routing table. By default, TBF lists all existing system routes.

You see information for each route, including:

- **Destination**: The target network or host for each route.
- **Interface**: The network interface used by the route.
- **Gateway**: Indicates whether the route uses a gateway for forwarding traffic.
- **MTU**: The maximum transmission unit, which defines the largest data packet size that the system transmits in a single network transaction.
- **Priority**: A numeric value that determines route preference (displayed for some routes only).
- **Type**: Shows the source of the route. Common types include static, OSPF, and BGP.

To create routes, navigate to **Routing > Routes**. For more information, see the *routes settings* on page 209.

Services

This page displays service status, names, and port assignments.

Find the service diagnostics under **Diagnostics > Services > Services**.

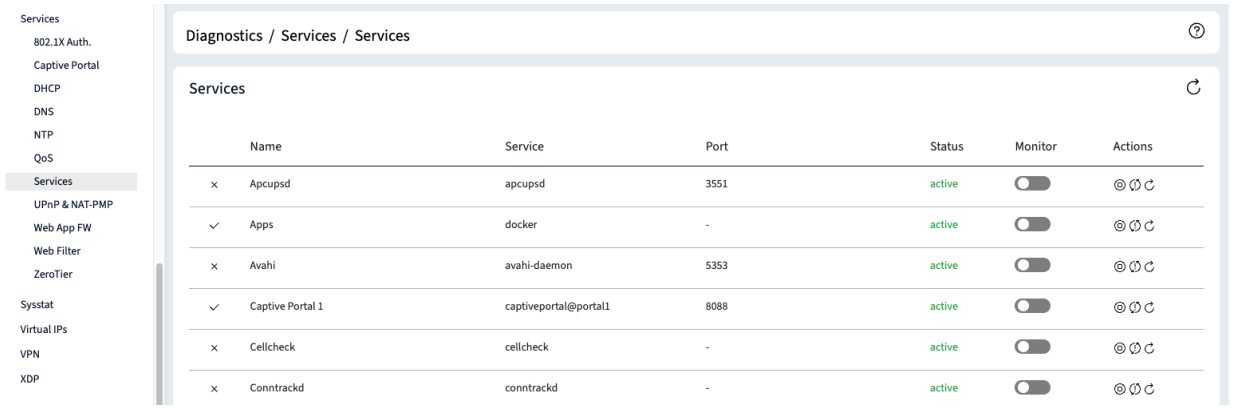


Figure 340: Services

Here you see the TBF services. The first column indicates whether each service is enabled or disabled.

The remaining columns show the TBF service name, the corresponding system name, and the ports that each service uses.

Note: These ports remain closed to external access unless you create specific firewall rules to open them.

Status

The status is one of the following:

- Active
- Reloading
- Inactive
- Failed
- Activating
- Deactivating

Monitor

Check a service if you want TBF to monitor the service. TBF then regularly checks the status of the service and restarts it if it is not running.

Actions

Perform different actions depending on the status of the service.

If the service status is inactive or unsuccessful, start the service.

If the status is active, reloading, activating, or deactivating, stop or reload the service.

SFP

This page displays Small Form-factor Pluggable (SFP) module information per interface.

Note: This section contains screenshots from the former VT Air GUI. These will be updated in a future version.

Find the SFP diagnostics under **Diagnostics > Interfaces > SFP**.

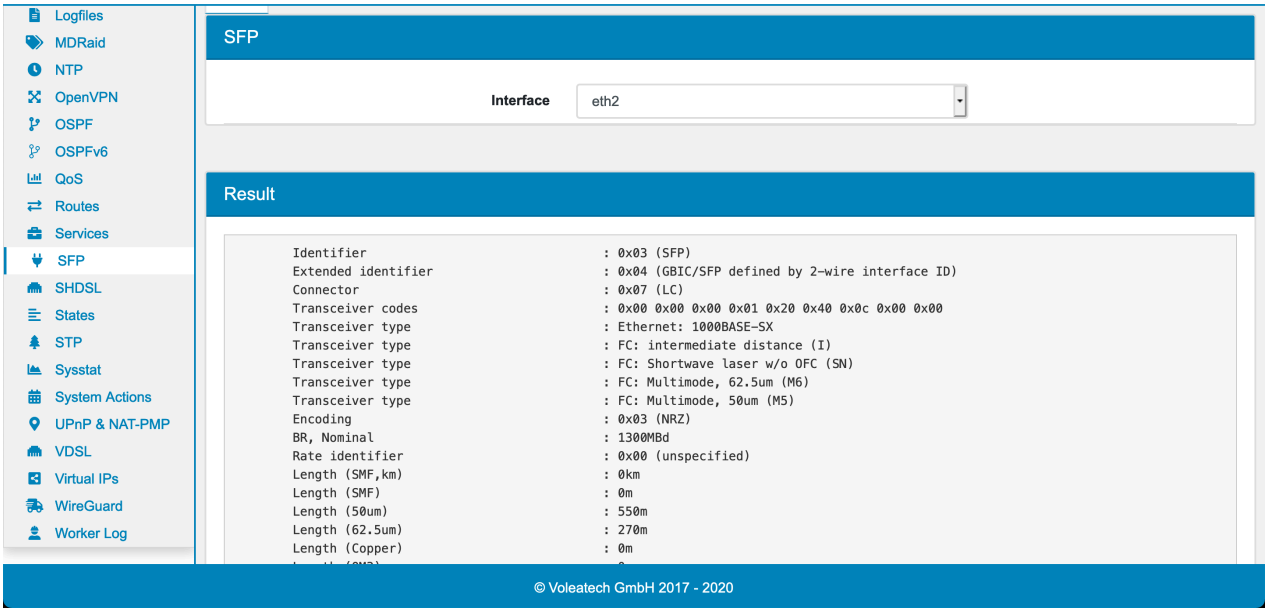


Figure 341: Sfp result

Here you see the Small Form-factor Pluggable (SFP) information. To view the information for the interface, select an interface from the list.

The list prefilters the interfaces. To display the information, select the appropriate SFP interface.

SHDSL

This page displays SHDSL interface settings and link status.

Find the SHDSL diagnostics under **Diagnostics > Interfaces > SHDSL**.

Here you see the settings and the status of the SHDSL setup.

To change the SHDSL setup, navigate to **Interfaces > Assign > SHDSL**. For more information, see the *SHDSL settings on page 115*.

Output

The diagnostics page shows the active configuration, modem connection status, and connection data including PAM, speed, SNRM, and LATN.



Figure 342: Current settings

You also see performance counters such as CRC errors. `NE` represents the local errors, and `FE` the foreign errors, if available.

The performance counters include:

- **SNRM**: Signal to Noise Margin
- **LATN**: Line Attenuation
- **PBO**: Power Back-off
- **ES**: Errored Seconds
- **SES**: Severe ES
- **AUS**: UnAvailable Seconds
- **LOSW**: Loss Of SyncWord
- **CRCa**: Cyclic Redundancy Check

Clear Stats helps you reset the performance counters.

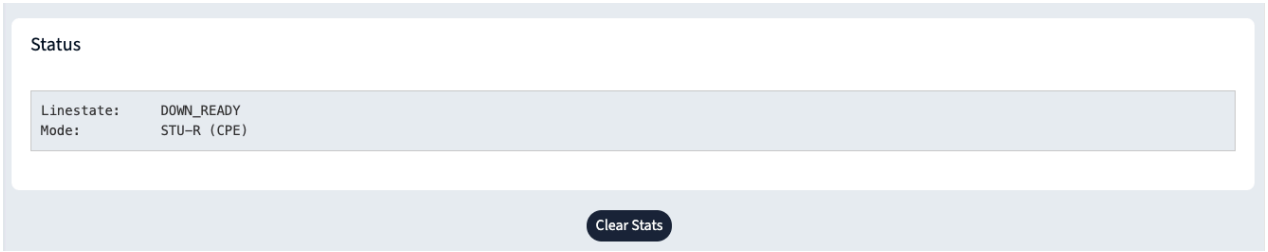


Figure 343: Status

States

This page displays active firewall connection states and HA synchronization entries.

Find the state diagnostics under **Diagnostics > Firewall > States**.

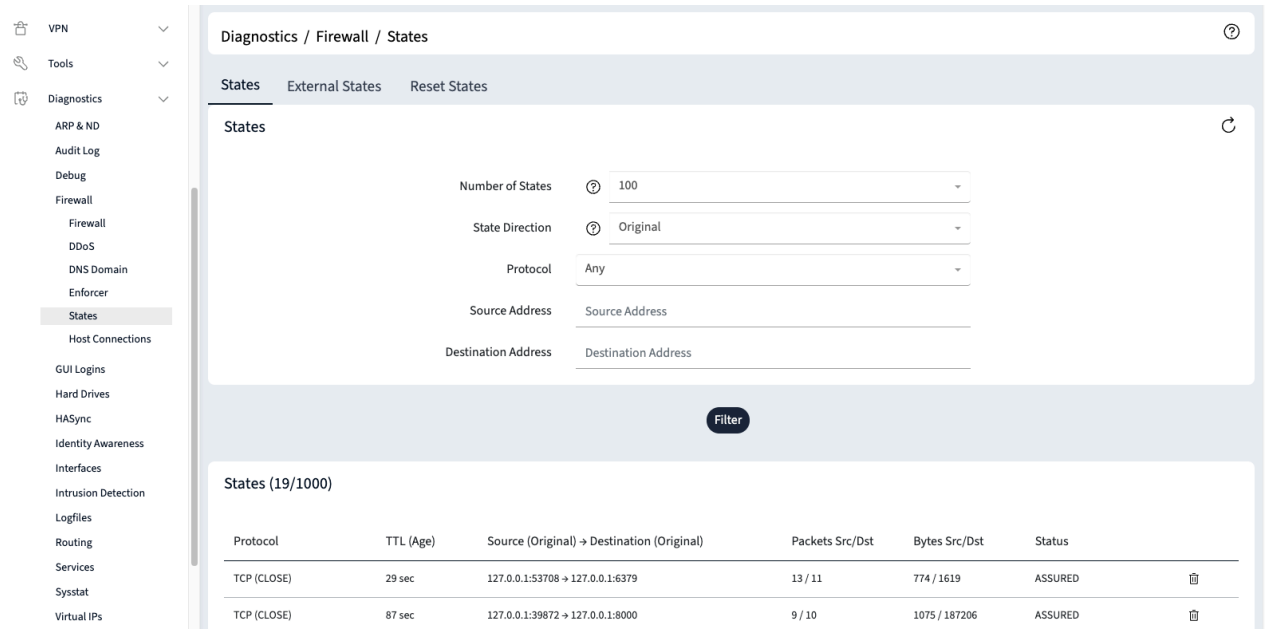


Figure 344: States

Here you see the network **States**. If **States Sync** is enabled in **High Availability**, you also see the **External States**.

External States are the states that the system synchronizes from the other firewalls. The states remain in the **External States** status until the firewall takes over the master role on VIPs. The system then merges these states into the normal **States**.

Each entry shows which **Protocol** the entry uses, for example TCP, UDP, or ICMP.

The **TTL** (time to live) value determines the lifespan of the state.

The **Source** and **Destination** columns show the IP address. If you use some kind of address translation, like NAT, TBF displays the **Original** address in parentheses.

You see how many packets the system processes within a state by **Packets Src/Dst** and **Bytes Src/Dst**. This information shows how much data is going through an open state.

Lastly, TBF displays the **Status**. To delete the network state, select the **Delete** symbol.

You can filter the states by port and IP address in the original or reply direction. By default, TBF shows the first 10,000 states. You can show more states.

CAUTION: Showing the states takes a long time if you have more than 100,000 state entries.

To reset the states, select **Reset all States**. This action terminates the active connections. Reestablishing the connections takes some time. This operation causes the browser session to become unresponsive temporarily. In this case, refresh the page to continue.

State actions

Each state has actions connected to the state on the right side of the state entry. You can delete a state.

If you associate a state with a firewall rule, you can navigate directly to the rule via the **Rule** symbol.

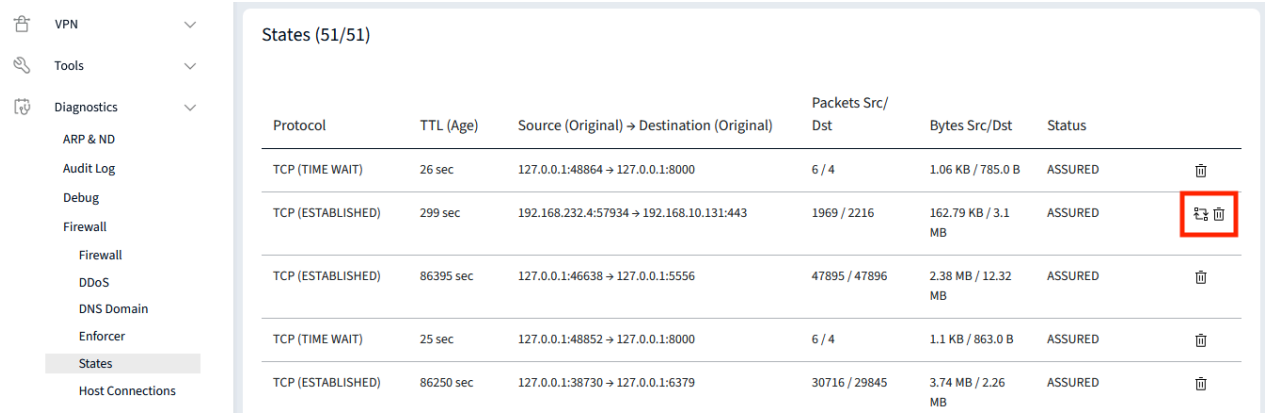


Figure 345: States actions

STP

This page displays Spanning Tree Protocol (STP) bridge, tree, and port information.

Find the STP diagnostics under **Diagnostics > Interfaces > STP**.

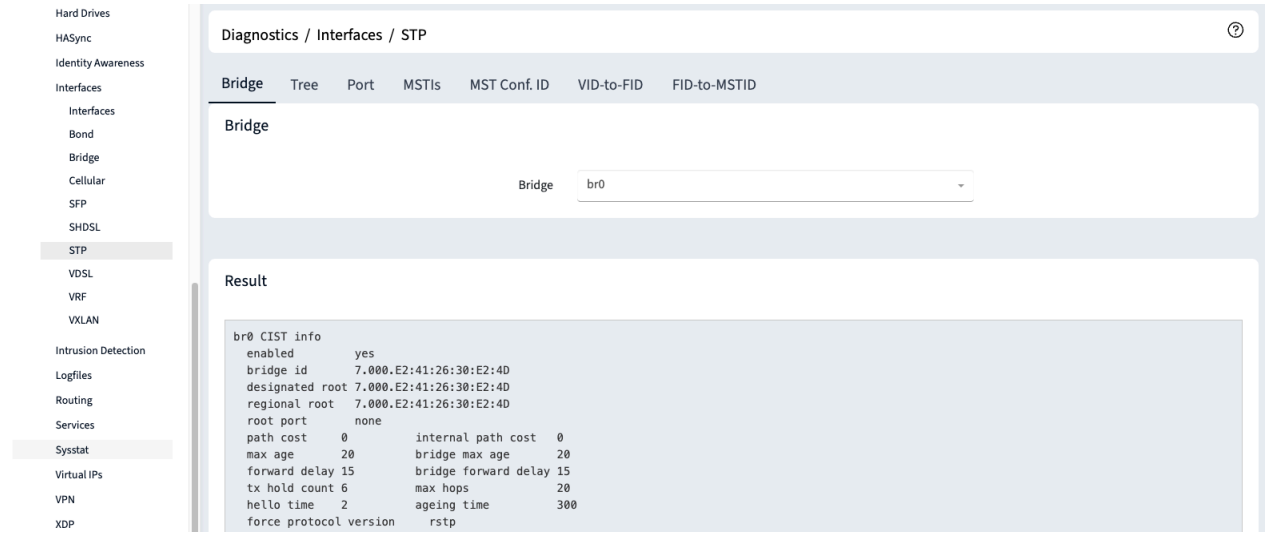


Figure 346: STP result

Here you see the Spanning Tree Protocol (STP) information organized into the following categories:

- Bridge
- Tree
- Port
- MSTIs
- MST Conf. ID
- VID-to-FID
- FID-to-MSTID

To view the STP information, select a bridge from the list.

To change the bridge STP settings, navigate to **Interfaces > Assign > Bridges**. For more information, see *bridge* on page 99.

Sysstat

This page displays CPU, memory, I/O, and network usage statistics as graphs.

Find the sysstat diagnostics under **Diagnostics > Sysstat**.

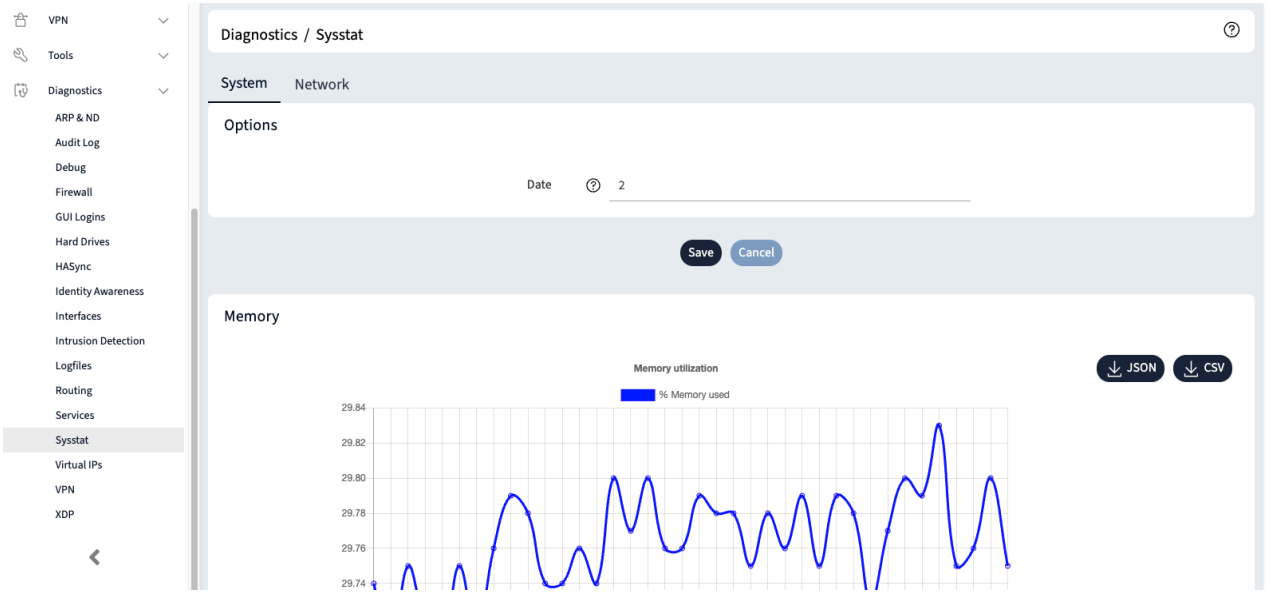


Figure 347: Options and memory

Here you see informational graphics that show the usage statistics for CPU, memory, I/O, and network resources.

You can download the data for each category in the following formats:

- JSON
- CSV

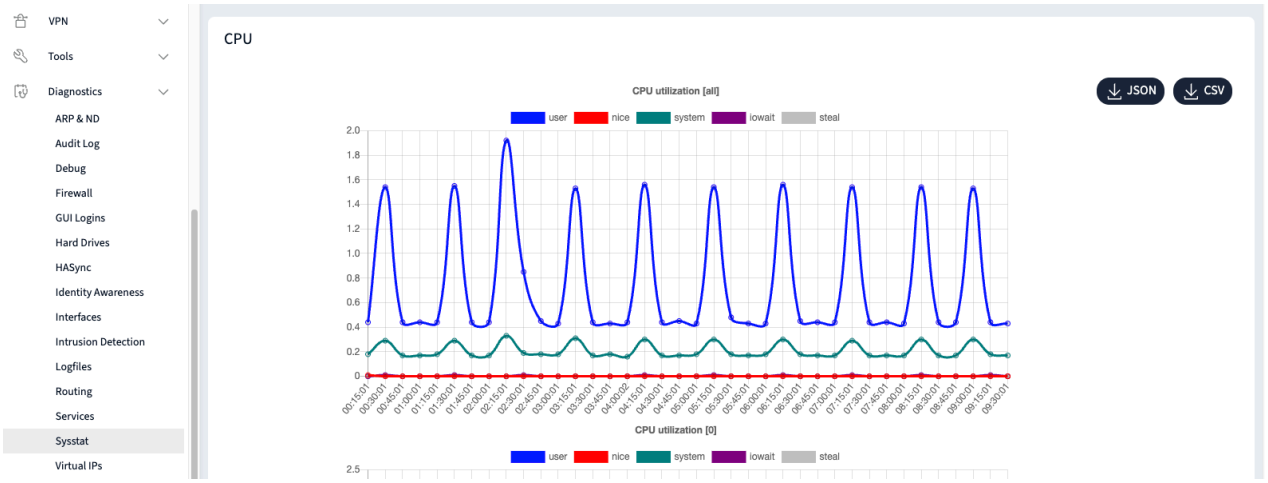


Figure 348: CPU utilization

You can select a date to display in the graphs. TBF maintains data in rotation for one month per date. You can select any date between the first and the last day of the month.

When the current month has not yet reached a particular day, TBF automatically fills the gap with data from the previous month.

On the **Network** tab, you can also show data for the network interfaces or filter the view to show data for a single interface.

System actions

This page displays scheduled system actions with their last execution and next runtime.

Find the system action diagnostics under **Diagnostics > Debug > System Actions**.

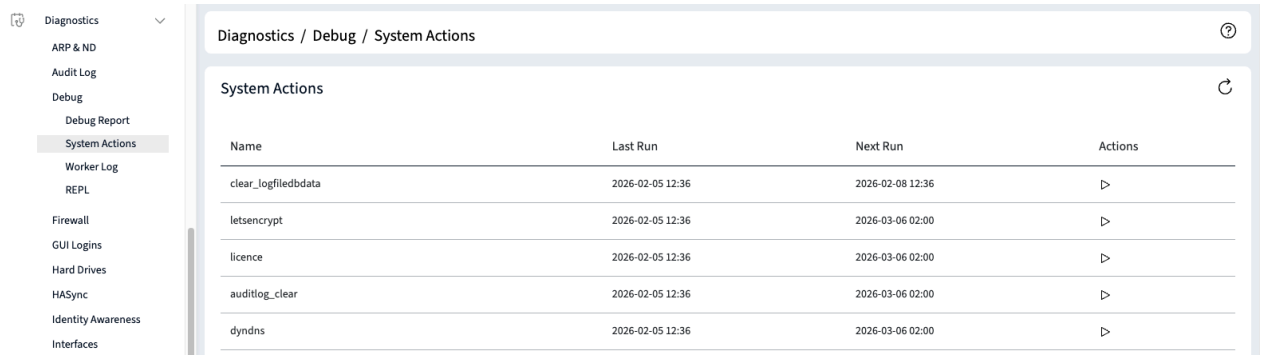


Figure 349: System actions

System actions function similarly to cron jobs. The key difference is that the TBF management process executes the system actions. On this page, you can manually trigger these actions or view the last execution time and next scheduled runtime of the actions.

UPnP and NAT-PMP

This page displays UPnP and NAT-PMP port forwarding entries created automatically by clients.

Find the UPnP NAT diagnostics under **Diagnostics > Services > UPNPNAT**.

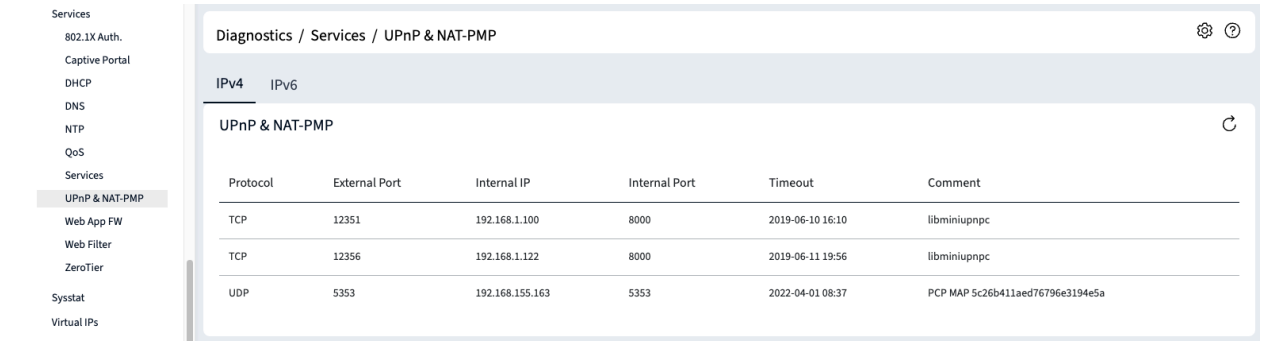


Figure 350: UPnP & NAT-PMP

Here you see the UPnP NAT entries that the firewall automatically creates. Each entry represents a port forward to a client.

TBF organizes the entries into two categories:

- IPv4
- IPv6

VDSL

This page displays VDSL interface settings and line status data.

Note: This section contains screenshots from the former VT Air GUI. These will be updated in a future version.

Find the VDSL diagnostics under **Diagnostics > Interfaces > VDSL**.

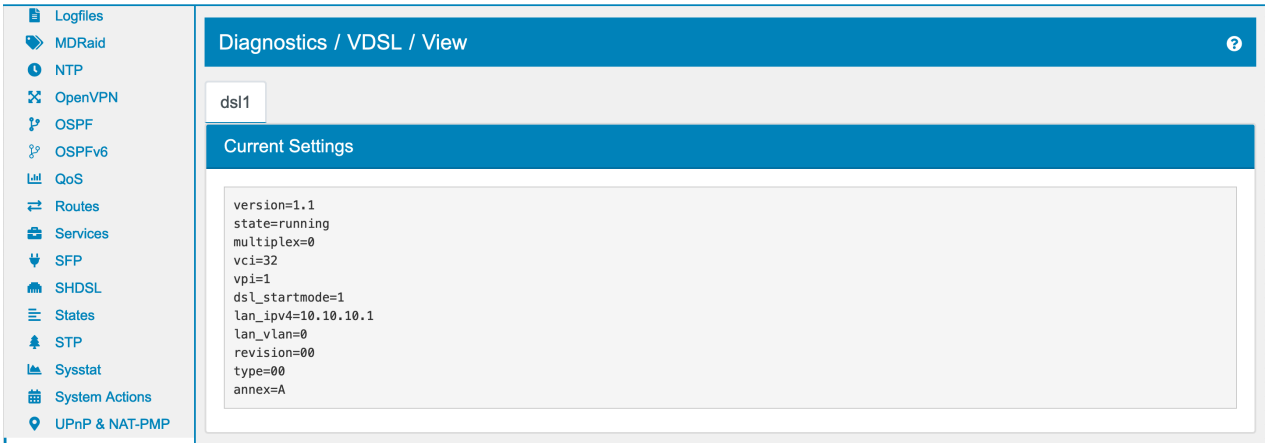


Figure 351: Current settings

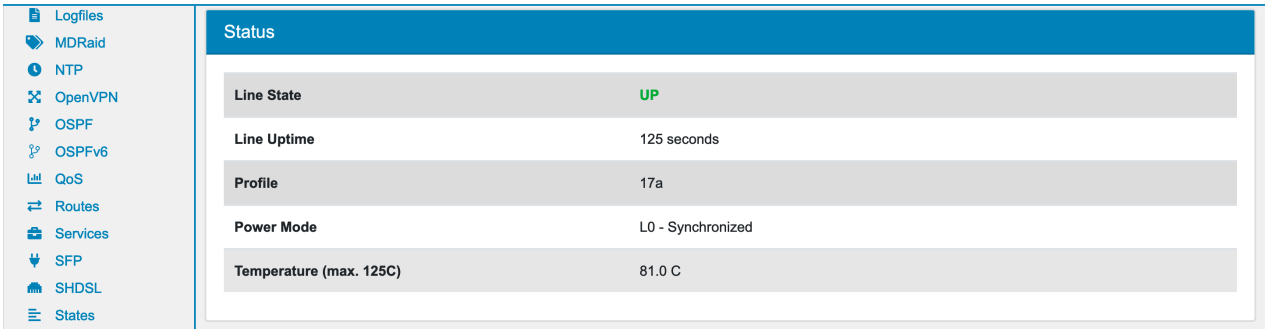


Figure 352: VDSL status

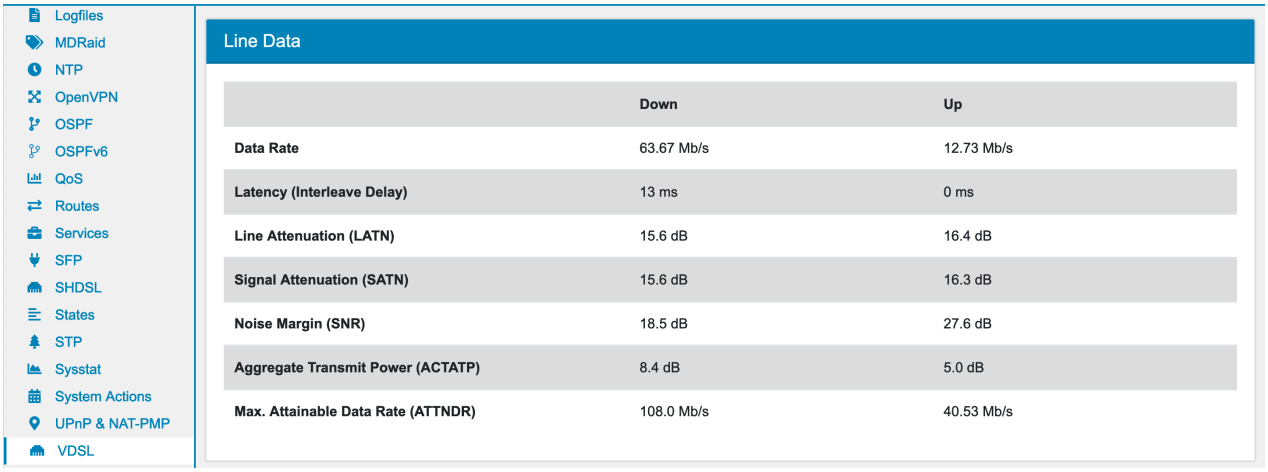


Figure 353: VDSL line data

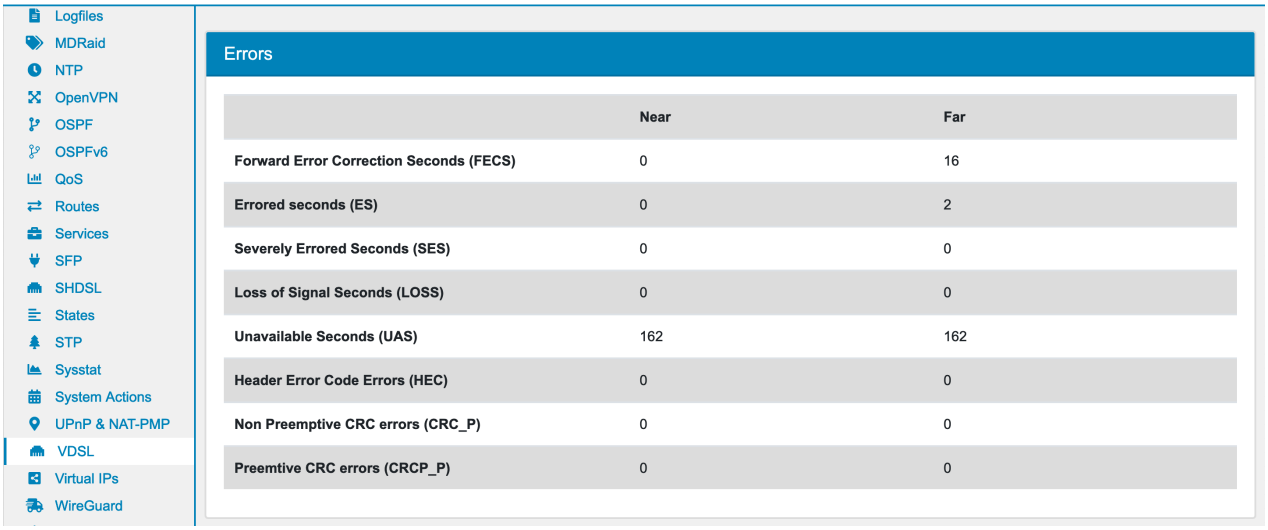


Figure 354: VDSL errors

Here you see the settings and operational status of the VDSL setup.

To change the VDSL setup, navigate to **Interfaces > Assign > VDSL**. For more information, see the *VDSL settings* on page 119.

Current Settings shows the setup of the VDSL modem.

Status shows the following information:

- Line State
- Line Uptime
- Profile
- Power Mode
- Temperature

Line Data shows the following information:

- Data Rate
- Latency (Interleave Delay)
- Line Attenuation (LATN)
- Signal Attenuation (SATN)
- Noise Margin (SNR)
- Aggregate Transmit Power (ACTATP)
- Max. Attainable Data Rate (ATTNDR)

Errors shows the following information:

- Forward Error Correction Seconds (FECS)
- Errored seconds (ES)
- Severely Errored Seconds (SES)
- Loss of Signal Seconds (LOSS)
- Unavailable Seconds (UAS)
- Header Error Code Errors (HEC)
- Non Preemptive CRC errors (CRC_P)
- Preemptive CRC errors (CRCP_P)

Virtual IPs

This page displays VRRP virtual IP address status and configuration.

Find the virtual IP address diagnostics under **Diagnostics > Virtual IPs**.

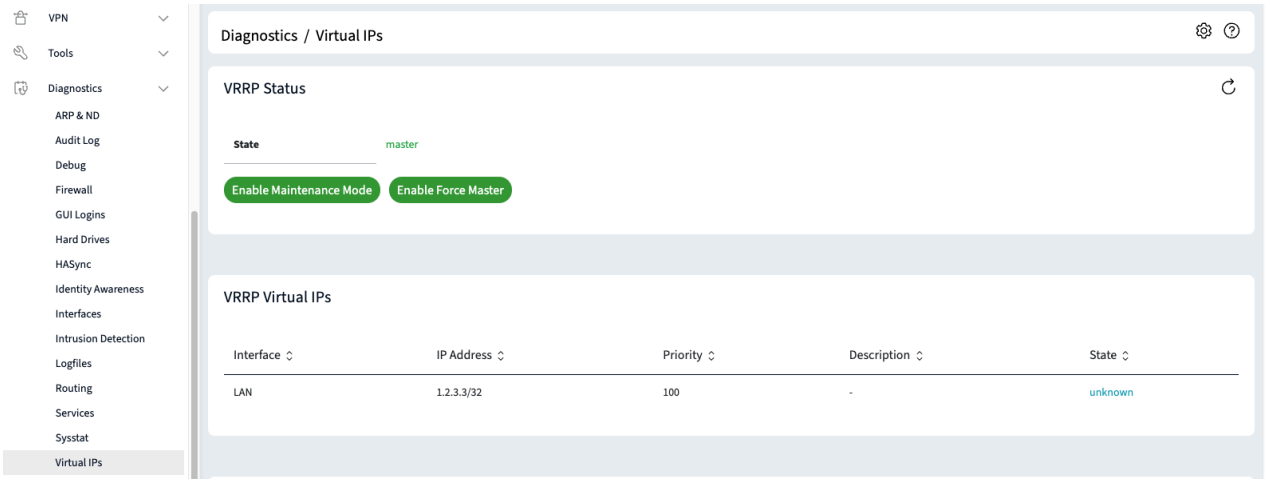


Figure 355: VRRP status and VRRP virtual IPS

Here you see the virtual IP addresses setup settings and status.

At the page top, TBF displays the **VRRP Status**. There are two buttons:

- One to enable or disable **Maintenance Mode**
- Another to enable or disable **Force Master**

Note: Enable only **Maintenance Mode** or **Force Master** at one time, not both simultaneously.

View the **VRRP Virtual IPs** and **Alias Virtual IPs** along with the current system status.

To change the virtual IP address setup, navigate to **General > Virtual IPs**. For more information, see *VRRP* on page 205.

Maintenance mode

When you activate maintenance mode, TBF automatically sets the IP address VRRP ID to 1, which is the lowest ID possible. As a result, TBF becomes a slave when any other device is available in the network that also has the VRRP IP addresses set up. This also means that TBF remains the master when TBF is the **ONLY** device left that has the VRRP IP addresses set up.

The maintenance mode option remains active even after system reboots until you manually disable the option.

Use maintenance mode when you need to:

- Perform system maintenance on TBF
- Test high availability failover functionality
- Install updates

Force master

When you activate the force master mode, TBF automatically sets the IP address VRRP ID to 255, which is the highest ID possible. As a result, TBF becomes master when any other device is available in the network that also has the VRRP IP addresses set up.

The force master mode option remains active even after system reboots until you manually disable the option.

Use force master mode to force TBF to take over the master role during emergencies or when there are detected problems with the current master device.

VRF

This page displays VRF instance details and associated routing information.

Find the VRF diagnostics under **Diagnostics > Interfaces > VRF**.



Figure 356: VRF result

Here you see the VRF information. To view the VRF information, select a VRF from the list.

To change the VRF settings, navigate to **Interfaces > Assign > VRF**. For more information, see *VRF* on page 121.

VXLAN

This page displays VXLAN forwarding table entries.

Find the VXLAN diagnostics under **Diagnostics > Interfaces > VXLAN**.

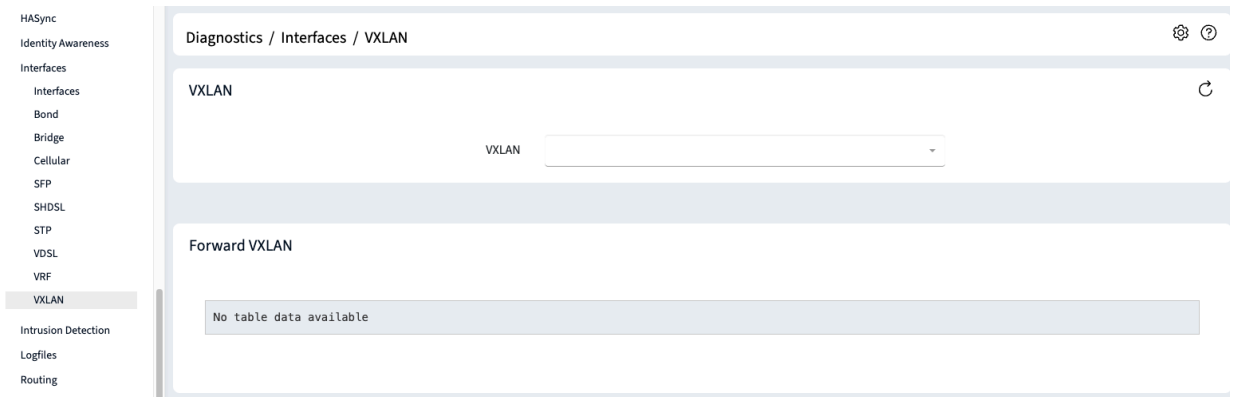


Figure 357: VXLAN

Here you see the VXLAN forward entry information. To view the VXLAN information, select a VXLAN from the list.

To change the VXLAN settings, navigate to **Interfaces > Assign > VXLAN**. For more information, see *VXLAN* on page 122.

Web filter

This page displays web filter logs and statistics including ClamAV and ICAP data.

Find the web filter diagnostics under **Diagnostics > Services > Web Filter**.

Here you see the web filter setup information. This information is organized into:

- Access log
- ClamAV log
- ICAP log
- ICAP access
- ICAP statistics

Note: The page automatically updates the information without user input.

To change the web filter setup, navigate to **Services > Web Filter**. For more information see *web filter* on page 301.

Access Log shows real-time information for users who access the proxy server, including details for request and response statuses. To filter the access log entries, use the search field.

Clamav Log displays the ClamAV log file content.

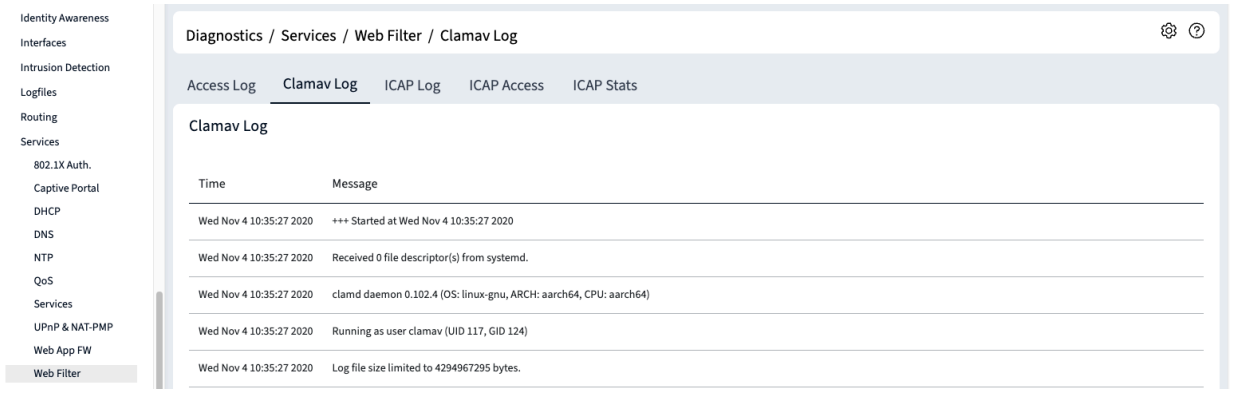


Figure 358: Clamav log

ICAP Log shows the contents of the ICAP log file, which includes virus detection alerts.

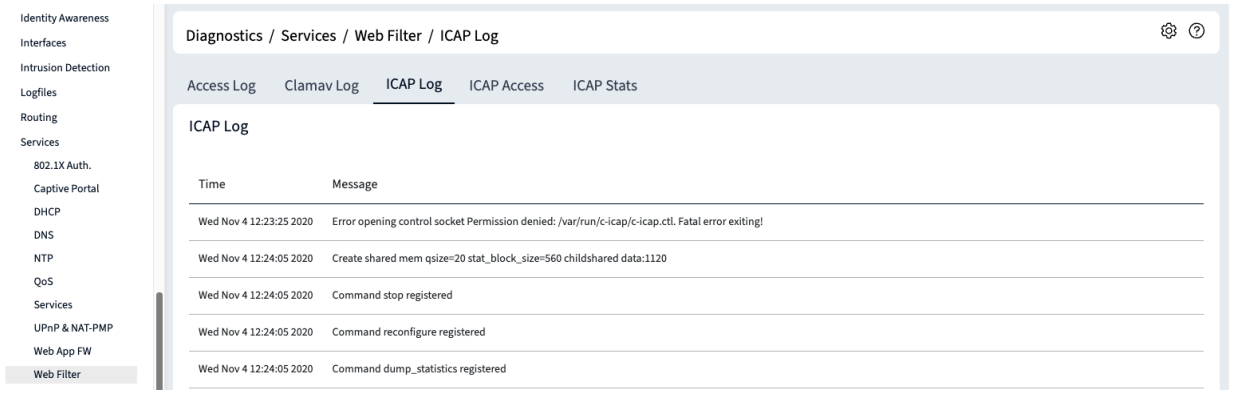


Figure 359: Icap log

ICAP Stats shows the ICAP statistics, which include:

- Running servers statistics
- General statistics
- Service info statistics
- Service echo statistics
- Service squidclamav statistics

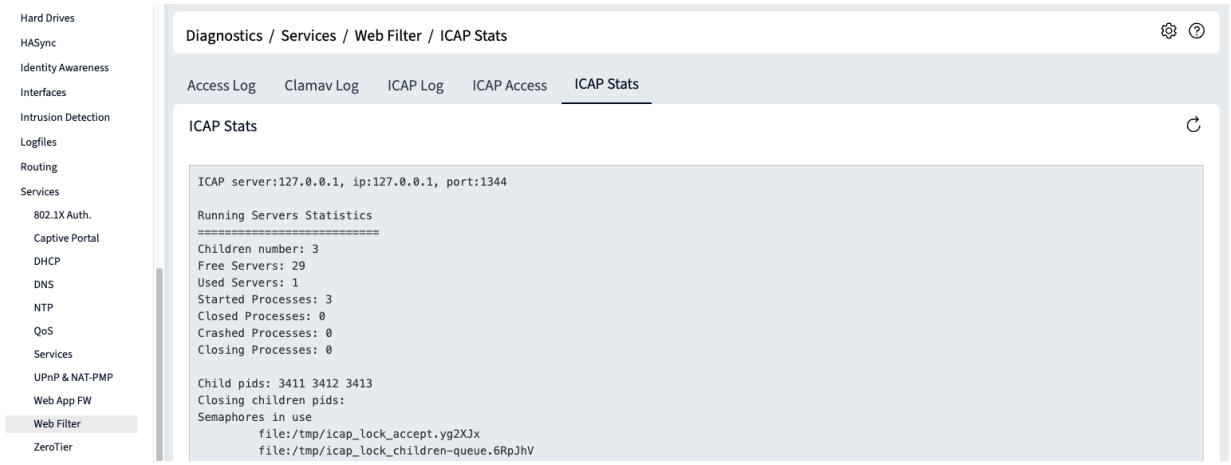


Figure 360: Icap stats

Web application firewall

This page displays web application firewall overview, dashboard, and audit logs.

Find the web application firewall diagnostics under **Diagnostics > Services > Web App FW**.

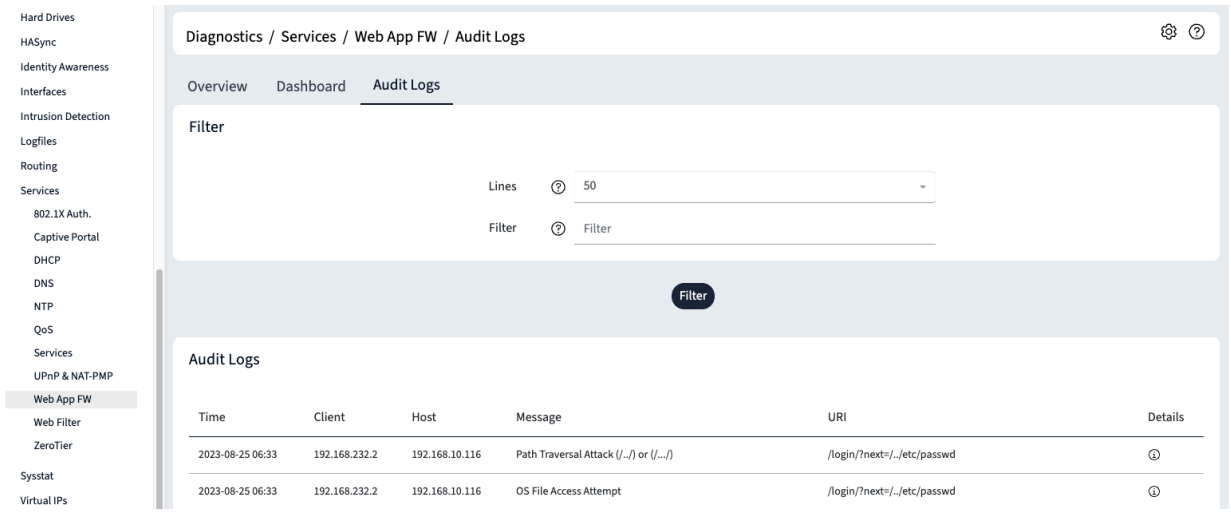


Figure 361: Audit logs

There are three tabs:

- **Overview**: Displays general information, including the rules version in use.
- **Dashboard**: Shows web application firewall information organized by country and origin.
- **Audit Logs**: Shows detailed information for each rule that triggers.

Each log entry includes information about time, client, host, message, and URI.

Additionally, select the **Details** symbol on the right to access more detailed information for each entry.

Dashboard

The dashboard is a convenient way to visualize the event data patterns.

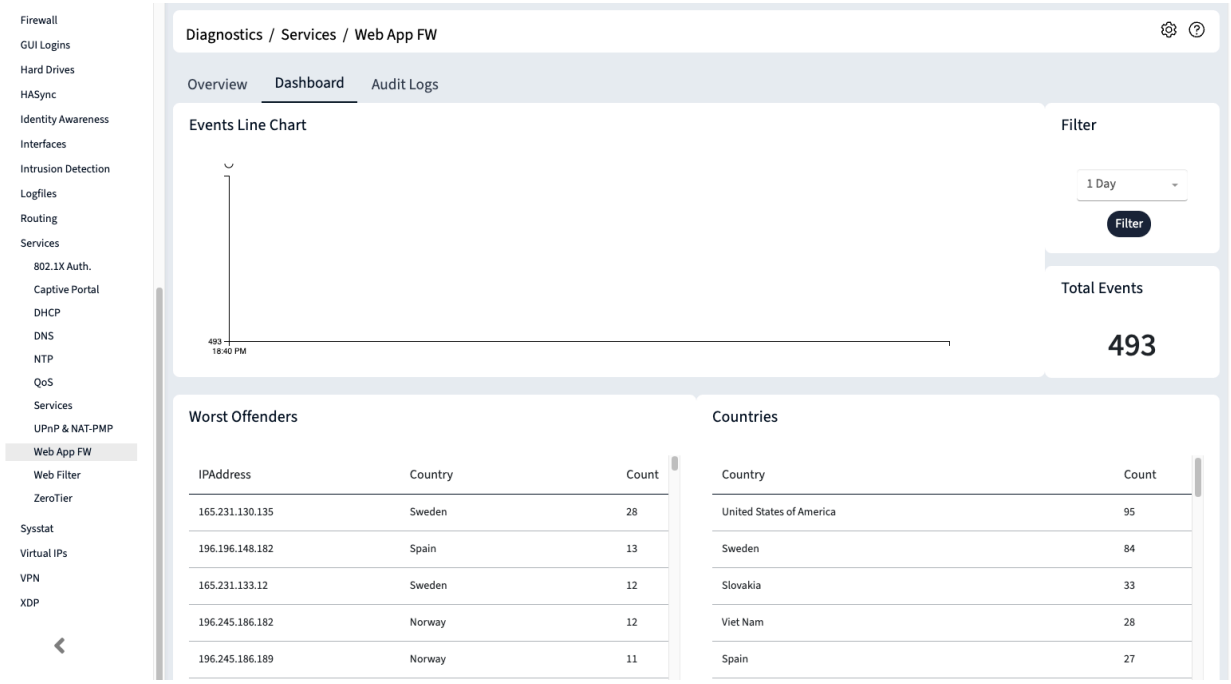


Figure 362: WAF dashboard

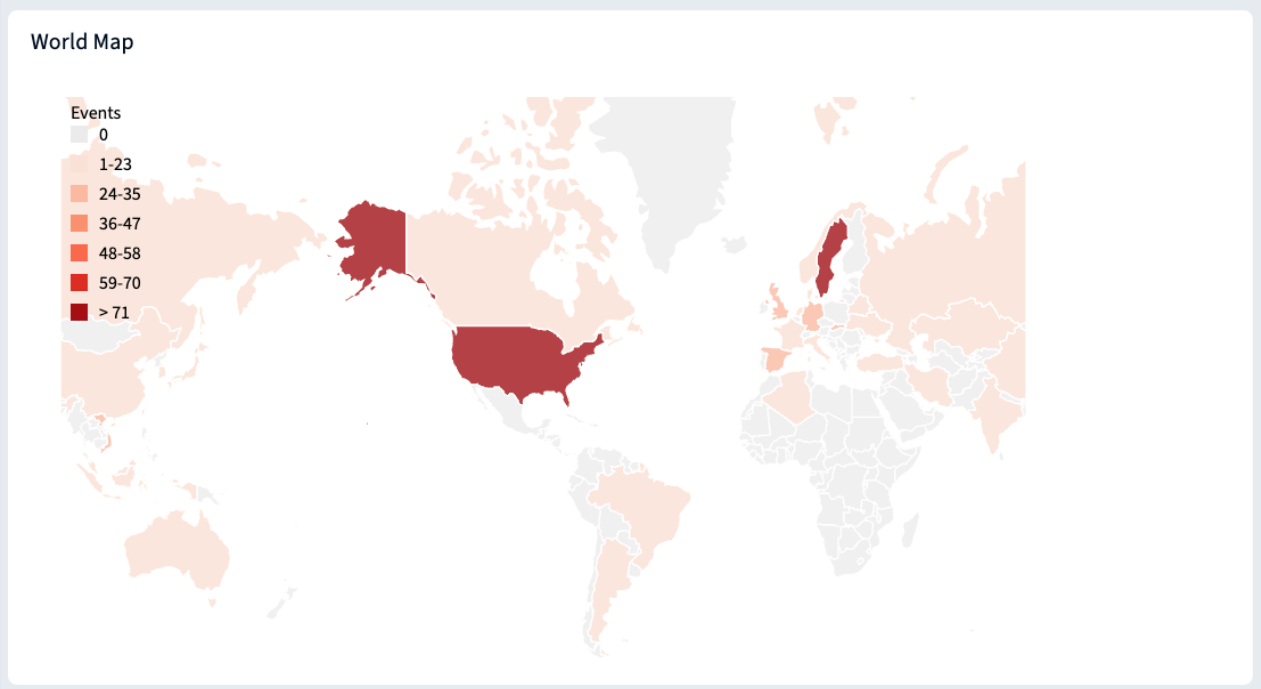


Figure 363: Dashboard world map

Note: Displaying data in the dashboard requires log file analysis. TBF disables this feature by default because of performance considerations. You can enable the feature in the *settings* on page 63.

WireGuard

This page displays WireGuard tunnel configuration and peer status.

Find the WireGuard diagnostics under **Diagnostics > VPN > WireGuard**.

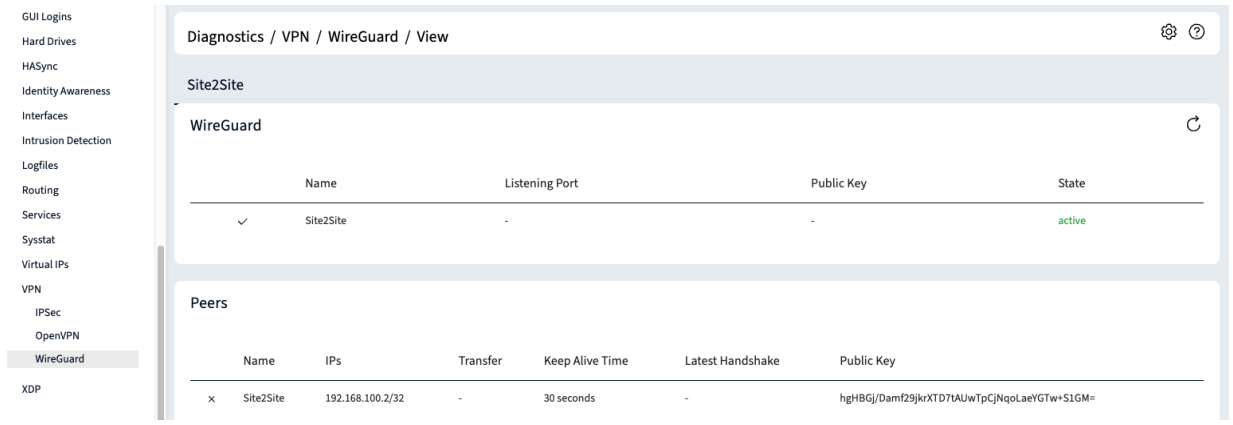


Figure 364: WireGuard

Here you see the WireGuard configuration information.

To change the WireGuard setup, navigate to **VPN > WireGuard**. For more information, see *WireGuard* on page 352.

TBF displays the set up WireGuard instances. The following information displays:

- **Name** shows the name of the selected WireGuard.
- **Listening Port** is the port on which the system receives connections.
- **Public Key** is a shortened string of the public key.
- **State** shows the current operational status of the service.

The next table shows the peers. The peer information includes:

- The first column displays a symbol that indicates whether the peer is **Connected** or **Not Connected**.
- **IPs** are the IP addresses that the peer can use.
- **Transfer** shows how much data this peer has received and transmitted.
- **Keep Alive Time** shows in seconds how long the connection remains alive.
- **Public Key** is a shortened string of the public key.

Worker log

This page displays background worker process execution logs for debugging.

Find the worker log under **Diagnostics > Debug > Worker Log**.

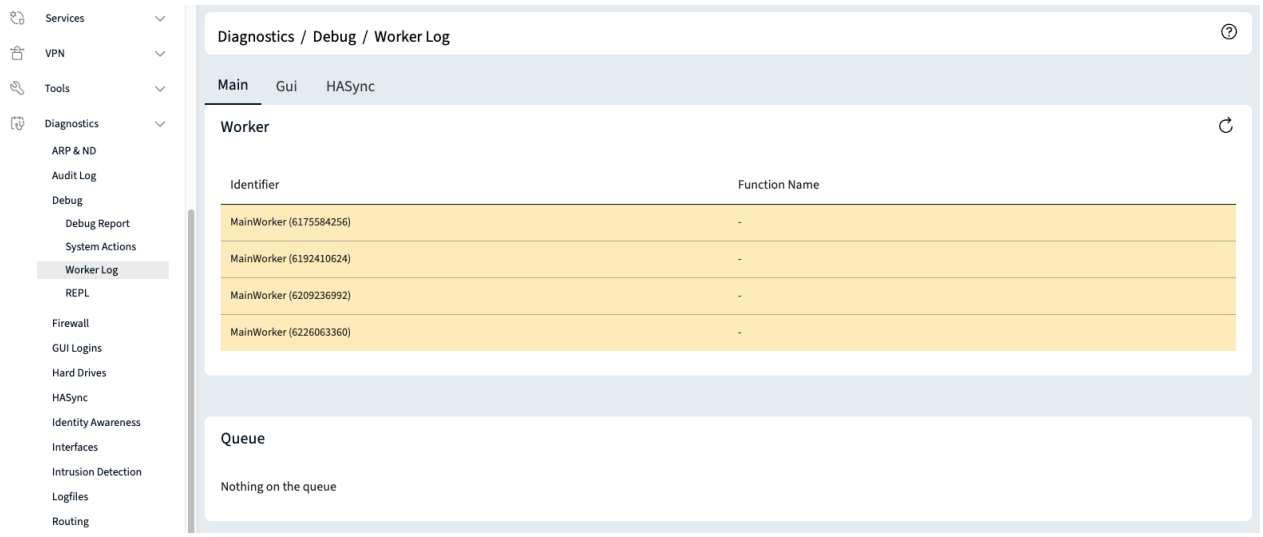


Figure 365: Worker log

The worker log displays information for the background worker process that handles functions and processes information within the system. The worker log shows the functions that execute and provides valuable debugging information for TBF.

XDP

This page displays XDP acceleration status and allows temporary enable or disable.

Find the XDP diagnostics under **Diagnostics > XDP**.

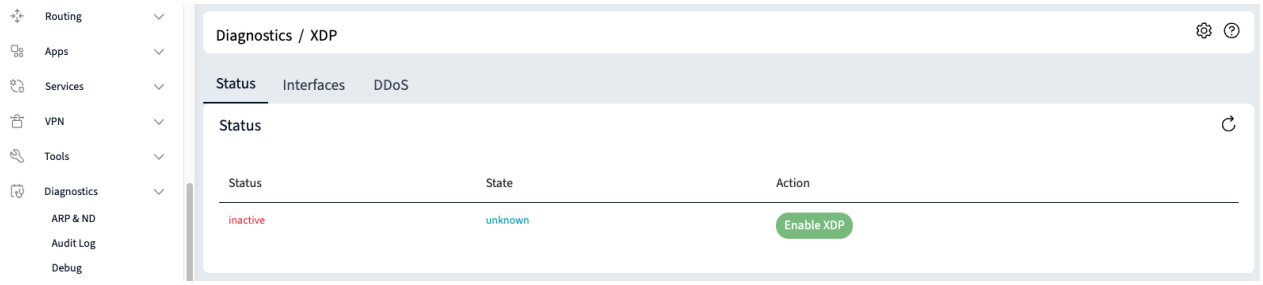


Figure 366: XDP status

Here you see the status of the XDP setup, and enable or disable XDP temporarily. When you disable XDP, TBF preserves the general XDP configuration. The preserved configuration is useful for testing or debugging purposes.

To permanently enable or disable the XDP setup, go to **System > Settings > Advanced**. For more information, see *XDP* on page 20.

Interfaces

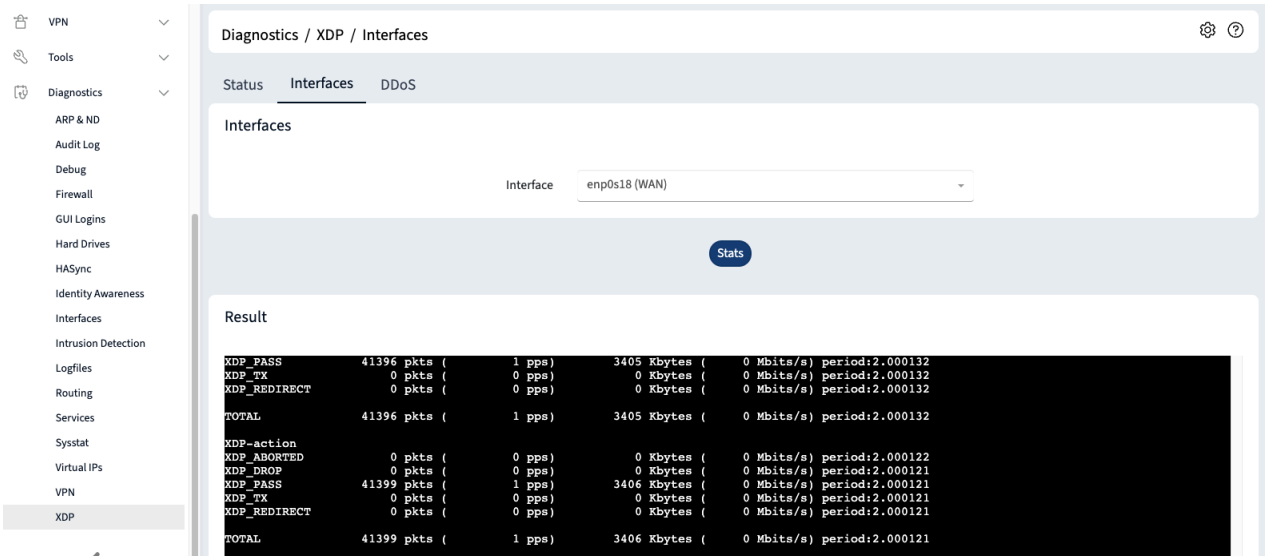


Figure 367: XDP result

Here you see statistics for interfaces that support XDP. The statistics show how many packets the system forwards or passes on to the normal firewall.

DDoS

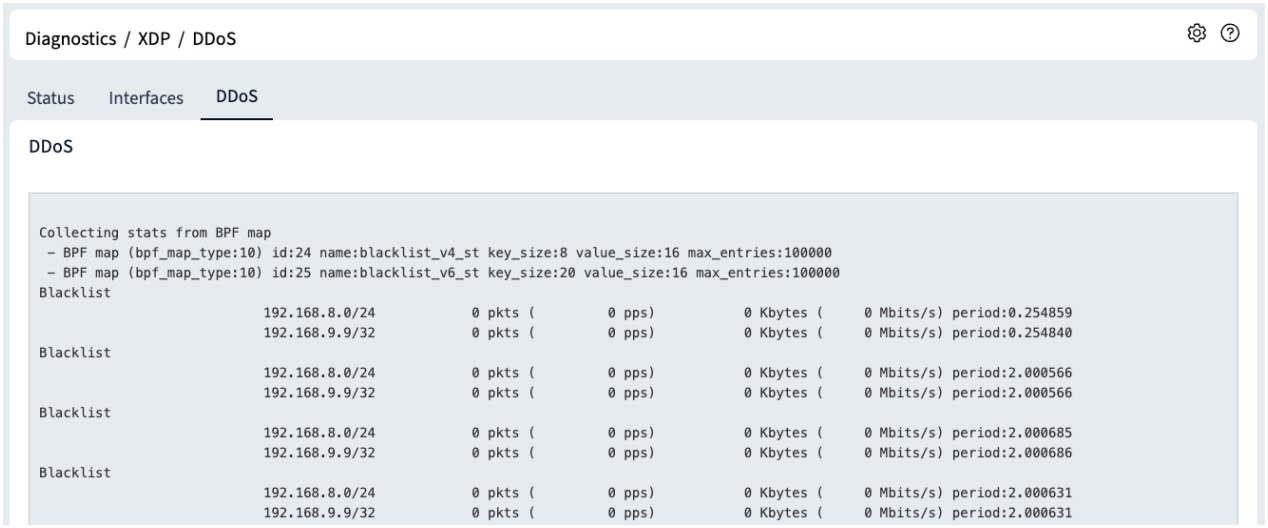


Figure 368: DDOS

Here you see statistics for the XDP DDoS protection functionality. XDP DDoS protection enables efficient packet dropping at near line-rate speeds. This packet dropping helps to protect the firewall from being overwhelmed by high-volume attacks. TBF automatically loads the firewall DDoS rules.

ZeroTier

This page displays ZeroTier network membership, node info, and peer connections.

Find the ZeroTier diagnostics under **Diagnostics > Services > ZeroTier**.

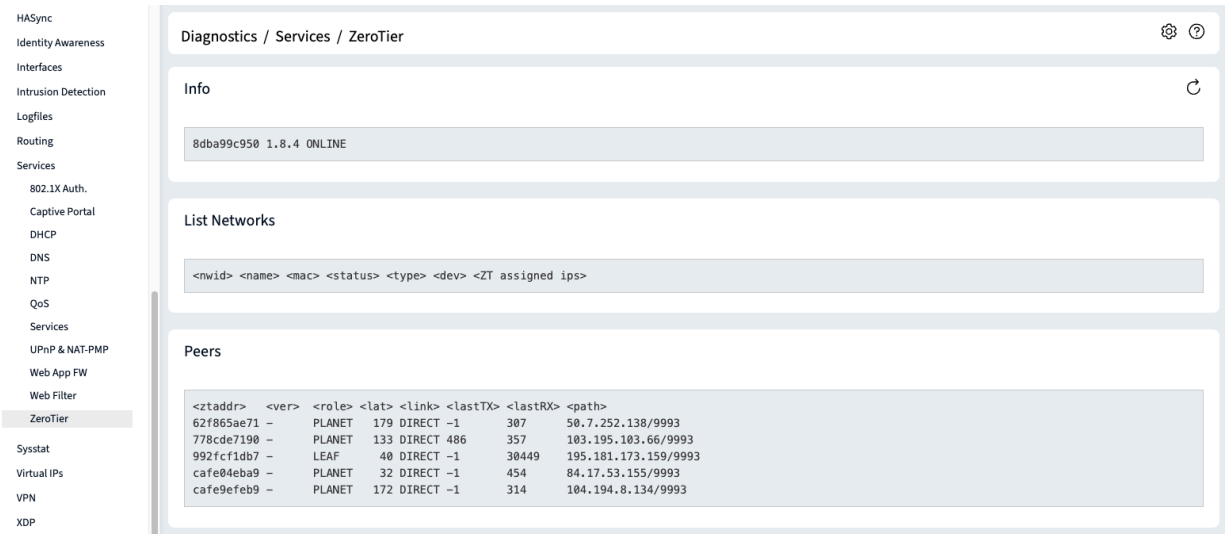


Figure 369: ZeroTier

The page displays three main sections:

- Info
- List networks
- Peers

To change the ZeroTier setup, navigate to **Services > ZeroTier**. For more information, see *ZeroTier* on page 320.

21. Tools

To monitor, troubleshoot, and manage the system, access diagnostic, maintenance, and administrative tools.

This section contains access to various diagnostic, maintenance, and administrative utilities that help you monitor system performance, troubleshoot network issues, and manage system data.

Diagnostic tools

To test network connectivity and monitor system performance, use these tools:

- *ARPing* on page 440: Test reachability via MAC address using ARP messages
- *Ping* on page 445: Test host reachability using ICMP messages
- *Traceroute* on page 451: Trace network path to destination
- *Iftop* on page 443: Monitor network traffic and bandwidth usage
- *Top* on page 451: Display running processes and system resource usage
- *Tcpdump* on page 447: Capture and analyze network packets
- *LLDP* on page 443: View Link Layer Discovery Protocol information

Maintenance tools

For system maintenance and management, use these tools:

- *Backup & restore* on page 441: Create and restore system backups
- *Patch* on page 444: Apply system updates and patches
- *Factory defaults* on page 442: Reset the system to factory settings
- *Restart* on page 446: Restart system services or the entire system
- *Shutdown* on page 447: Safely shut down the system

Administrative tools

For system administration and configuration, use these tools:

- *Web console* on page 441: Access system console through web interface
- *File manager* on page 442: Manage system files and directories
- *Script* on page 446: Execute custom scripts and commands
- *Setup wizard* on page 447: Initial system configuration wizard
- *Templates* on page 450: Manage configuration templates
- *Wake on LAN* on page 452: Wake up network devices remotely

The tools display real-time feedback and results through the web interface. Most diagnostic tools display continuously updated information while they run.

ARPing

Test host reachability with MAC addresses. TBF sends Address Resolution Protocol (ARP) messages to the layer 2 network.

Find the ARPing tool under **Tools > ARPing**.

With this tool, test the reachability of a host via a MAC address. The tool sends an ARP message from the system to the entire layer 2 network and receives a response when the target MAC address is reachable.

The browser displays the information in real time.

Perform the following steps:

1. Enter the MAC address of the destination.
2. As needed, set the following options:
   - Number of pings
   - Source interface to use. The default is `auto`.
3. To start the test, select **ARPing**.

Backup & restore

Manage system data protection through backup creation and restoration processes.

Find the backup and restore settings under **Tools > Backup & Restore**.

Here you back up and restore the data of TBF.

The backup and restore functionality consists of 2 main operations:

- *Backup* on page 441: Create system data backups
- *Restore* on page 441: Restore system data from backups

Backup

To help protect against data loss and to enable system recovery, create backups of system data.

The backup functionality lets you download a file with the name `backup.json`. This file is in the JSON format and it contains the data of TBF. Before you create a backup, you can enter a password. The restore process also requires the password.

TBF automatically creates a backup every day at 02:00. To find the backup, look in `/var/lib/vtair/backup/`.

Note: If the web interface displays passwords unencrypted, the backup also stores them unencrypted. TBF and the backup store the user passwords encrypted.

Restore

Restore system data from backups that you created previously to recover the system configuration.

The restore functionality allows you to restore a backup that you previously created with TBF. If you entered a password for the backup, then enter it here as well.

Note: Do not modify the backup file. This action helps provide a working restore of the backup.

The GUI displays the restore process after you upload the backup file. Watch the restore process until the end. The GUI redirects you after the restore process completes. When the restore process finishes, TBF does not yet apply the backup settings. Map the interfaces of the restored device to the backup and apply the settings as needed.

Before you apply the restored settings, TBF does not apply any backup settings. You use passwords from the restored backup to sign in to the GUI.

Web console

This page describes the web-based console for direct system access.

Find the web console under **Tools > Console**.

This tool opens an interactive system console that allows you to operate directly on the system through a browser. The console allows you to access the shell interface as `root`.

WARNING: The web console is a Linux bash shell with administrator privileges. Proceed with care when using it.

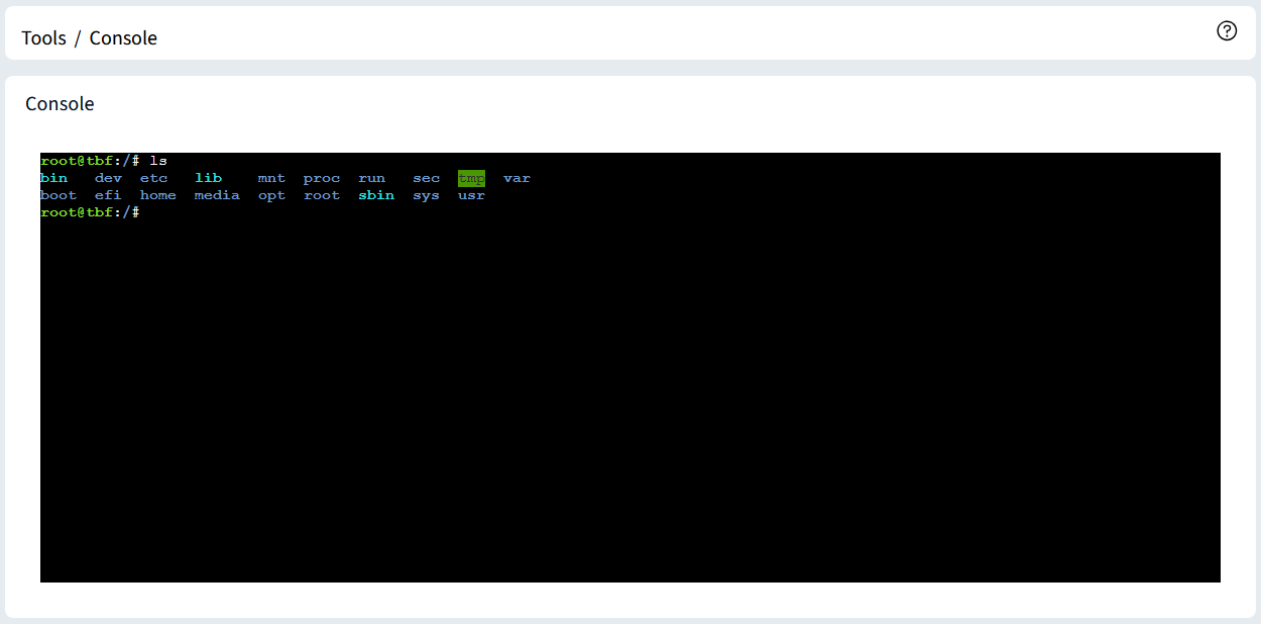


Figure 370: Console

Use *standard UNIX commands* to navigate and control the web console of the TBF device.

Factory defaults

Reset TBF to its original configuration, to restore the default settings and to return the device to a factory-new state.

Prerequisites:

Before you proceed, create a backup. You lose created and modified data in TBF.

Find the factory defaults option under **Tools > Factory Defaults**.

Perform the following steps:

1. Navigate to **Tools > Factory Defaults**.
2. To restore factory defaults, select the **Confirm** button.

   The device restores the factory defaults and reboots automatically.

After the reboot, TBF returns to its factory state with the default settings and resets the created and modified data. You can then start fresh with a clean configuration.

The `LAN` interface is accessible again at the default IP address `192.168.1.1`.

File manager

The file manager tool allows you to upload and download files to and from TBF.

Find the file manager tool under **Tools > File Manager**.

This tool lets you upload and download files to and from TBF.

Use the file manager

To navigate directories and to upload or download files, use the file manager.

To transfer files between a computer and TBF, use the file manager.

Perform the following steps:

1. To navigate through directories, select a folder.
2. To upload a file, select the **Upload** button. Select a file.

   TBF uploads the file to the currently open directory.
3. To download a file, select the file.

   TBF downloads the file to the computer.

Iftop

The iftop tool monitors network traffic and displays real-time bandwidth usage for network connections on a specified interface.

Find the iftop tool under **Tools > Iftop**.

The iftop tool displays a continuously updated list of network connections and their bandwidth usage. The information appears in real time in the browser and allows you to monitor network activity as it occurs.

The iftop tool displays network connections between hosts and shows both inbound and outbound traffic. Specify which network interface to monitor. This makes it useful for systems with multiple network interfaces. The display shows the source and destination of network traffic along with the current data transfer rates.

The iftop interface presents network traffic data in an organized format that updates automatically. This real-time monitoring capability helps you to:

- Identify network usage patterns
- Detect high-bandwidth connections
- Troubleshoot network performance issues

LLDP

Link Layer Discovery Protocol (LLDP) displays information about neighboring network devices that support the LLDP standard.

Find the LLDP tool under **Tools > LLDP**.

Enable or disable LLDP. TBF disables it by default.



Figure 371: LLDP

LLDP shows you information about other devices on the network that also use LLDP.

Patch

The patch tool applies targeted updates to specific system components without a complete system software update.

Find the patch tool under **Tools > Patch**.

The patch tool lets you apply patches to specific parts of the system without a complete system software update.

Apply and delete patches.

The status can be one of the following:

- Not applied yet
- Was unsuccessful to apply
- Successfully applied

Patch creation settings

These configuration settings are available when you create a new patch.

When you create a new patch, you have several options:

Description

This field describes the patch.

Patch Content

This field contains the content of the patch.

Path Strip Count

This setting specifies the number of leading components to strip from the patch file name.

Base Directory

This setting specifies the base directory for the patch. The default is `/`.

Ignore Whitespace

This setting ignores whitespaces in the patch content.

Ping

Ping is a network diagnostic tool that tests host reachability. The tool sends Internet Control Message Protocol (ICMP) echo request messages and measures response times.

This fundamental network diagnostic tool helps you:

- Test network connectivity between a device and a remote host
- Measure round-trip time for packets
- Identify packet loss issues
- Verify that firewall rules work correctly

The ping tool sends packets at regular intervals and displays the results in real time. The results show whether each packet reaches its destination and how long the round trip takes.

Find the ping tool under **Tools > Ping**. The browser displays the information in real time.

Use ping

Use the ping tool to test host reachability. The system sends ICMP messages.

Find the ping tool under **Tools > Ping**.

The browser shows the information in real time.

Perform the following steps:

1. Enter the hostname or IP address of the destination.
2. Set the following options as needed:
   - IP protocol v4 or v6
   - Number of pings
   - Source address. The default is `auto`.
3. To start the test, select **Ping**.

The ping results appear in real time in the browser.

For examples of successful and unsuccessful ping results, see the *ping examples* on page 445.

Ping helps you determine the reachability of a specific host or IP address. When you have a new rule configured that should block a specific website, for example, test the reachability and thus the effectiveness of the rule with ping.

Note: You do not need to visit the website to confirm that a rule is operational.

Ping examples

This topic shows examples of successful and unsuccessful ping results.

Successful ping result

A successful ping result looks like this:

```
PING www.wikipedia.com (91.198.174.192) 56(84) bytes of data.
64 bytes from text-lb.esams.wikimedia.org (91.198.174.192): icmp_seq=1 ttl=56 time=27.9 ms
64 bytes from text-lb.esams.wikimedia.org (91.198.174.192): icmp_seq=2 ttl=56 time=30.6 ms
64 bytes from text-lb.esams.wikimedia.org (91.198.174.192): icmp_seq=3 ttl=56 time=24.4 ms

--- www.wikipedia.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 24.441/27.682/30.699/2.566 ms
```

Unsuccessful ping result

An unsuccessful ping result looks like this:



Figure 372: Unsuccessful ping result

Restart

To reload services and apply configuration changes, restart TBF

Find the restart option under **Tools > Restart**.

Perform the following steps:

1. Navigate to **Tools > Restart**.
2. To restart TBF, select the **Restart** button.

   The web interface does not work during this time.

TBF restarts and the services reload with any pending configuration changes.

Related information
Shutdown on page 447

Script

The script tool saves custom script code to specific files on the system. System backups include the scripts.

Find the script tool under **Tools > Script**.

The script tool lets you save the code of a script to a specific file on the system. This way, the TBF backup includes the script.

Settings

File Path

This field specifies the absolute file path of the script.

Note: If you specify a file path that already exists in the system, then you cannot overwrite an existing file.

Script Content

This field contains the content of the script.

Description

This field describes the script.

Setup wizard

To set the general options, WAN interface, LAN interface, and administrator password, access and use the setup wizard.

Find the setup wizard under **Tools > Setup Wizard**.

Set up TBF with the same setup wizard that appears the first time you log in to TBF.

Perform the following steps:

1. Navigate to **Tools > Setup Wizard**.
2. Set the following options across the wizard pages:
   - General options
   - WAN interface
   - LAN interface
   - New administrator password

You can cancel the dialog and revisit it later.

To change options, go to the *settings* on page 352 and the corresponding interface pages.

Shutdown

To help prevent data loss and help provide proper system closure, shut down TBF.

Find the shutdown tool under **Tools > Shutdown**.

The web interface does not work during this time.

Perform the following steps:

To shut down TBF, select the **Shutdown** button.

TBF shuts down and stops the services and processes.

Note: When you want to restart TBF instead, use the *restart* on page 446 tool.

TCPDump

TCPDump is a packet analyzer tool that displays TCP/IP and other network packets in real time through a browser.

Find the TCPDump tool under **Tools > TCPDump**.

This tool is a packet analyzer that displays TCP/IP and other packets that the network transmits and receives.

The information appears in real time in the browser.

Note: To download the trace afterward, enable the **Save to File** setting **before** you start TCPDump.

The output looks like this:

```
tcpdump: listening on any, link-type LINUX_SLL (Linux cooked), capture size 262144 bytes
12:51:21.060482 IP (tos 0x0, ttl 64, id 3784, offset 0, flags [DF], proto TCP (6), length 162)
    vtair.localhost.hq.voleatech.com.8000 > IT1.hq.voleatech.com.57318: Flags [P.], cksum 0x92ec
 (incorrect -> 0xbd6a), seq 2584698705:2584698827, ack 1786006290, win 120, length 122
12:51:21.100800 IP (tos 0x0, ttl 127, id 26583, offset 0, flags [DF], proto TCP (6), length 40)
    IT1.hq.voleatech.com.57318 > vtair.localhost.hq.voleatech.com.8000: Flags [.], cksum 0x66f0
 (correct), seq 1, ack 122, win 2052, length 0
12:51:21.365874 IP (tos 0x0, ttl 64, id 46612, offset 0, flags [DF], proto ICMP (1), length 28)
    vtair.localhost.hq.voleatech.com > 192.168.10.1: ICMP echo request, id 0, seq 17114, length 8
12:51:21.365959 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto ICMP (1), length 28)
    192.168.10.1 > vtair.localhost.hq.voleatech.com: ICMP echo reply, id 0, seq 17114, length 8
12:51:21.384065 ARP, Ethernet (len 6), IPv4 (len 4), Request who-has 192.168.10.143 tell
 pfSensedev.dev.hq.voleatech.com, length 46
12:51:21.396593 IP (tos 0x0, ttl 64, id 14672, offset 0, flags [DF], proto TCP (6), length 91)
```

With TCPDump, analyze the communication on the network in great detail. If you are only interested in the communication of a specific host, then specify its IP address in the filter section. This way, the system provides the packets that this host sends and receives.

If you are only interested in the traffic of a specific host that relates to a specific port or protocol, then also specify the port number in the filter section. This way, customize the TCPDump request and get a much cleaner result that is a lot easier to interpret.

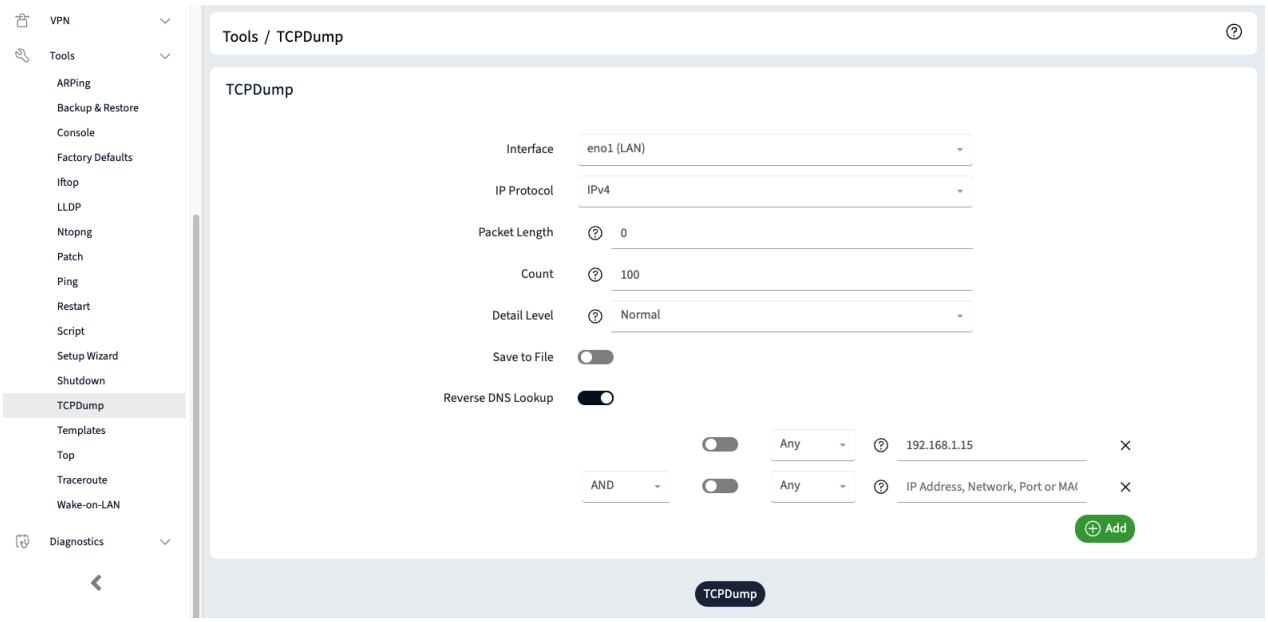


Figure 373: Tcpdump

Related topics

- *TCPDump settings* on page 448: Configuration parameters and filtering options
- *Troubleshooting via TCPDump* on page 449: Console commands for network troubleshooting

TCPDump settings

This topic describes the settings and filtering options for the TCPDump packet analyzer tool.

The TCPDump tool provides several configuration options to customize packet capture and analysis.

Settings

IP Protocol

This setting specifies the interface and the IP protocol, either IPv4 or IPv6.

Count

This field determines after how many packets the capture stops.

Packet Length

This field specifies the number of bytes captured for each packet. The default is `0`, which captures everything.

Detail Level

This setting determines the level of detail for the capture. The default is **Normal**.

Filtering options

When you do not use the IPsec interface, add additional filters by:

- Protocol
- IP address
- Port
- MAC address

Connect the filters with logical operators such as:

- `AND`
- `OR`
- `NOT`

Troubleshooting via TCPDump

This topic describes how to use TCPDump console commands to capture and analyze network packets for troubleshooting sporadic detected connection errors.

You can also access the TCPDump tool via the console. To do so, log in to the TBF device via SSH or via a serial connection. For more information, see *console access* on page 55.

This helps find sporadic connection or detected network errors that you cannot reproduce right away.

Perform the following steps:

1. To access the shell after you log in, enter `shell`.
2. Enter the command:

   ```
   nohup tcpdump -i INT -w /var/tmp/packetcapture.pcap -W 3 -C 10 -n host IPADDRESS and port PORT &
   ```

   Where:

   - `INT` is the physical interface, for example, `eth1`.
   - `IPADDRESS` is the IP address of the host you want to observe the connection with.
   - `PORT` is the port of the connection you want to observe.

   This creates a maximum of 3 rotating files with 10 MB. You can analyze the files later.
3. To end the process, enter:

   ```
   pkill tcpdump
   ```
4. To view the results afterward, load the file into Wireshark or enter:

   ```
   tcpdump -r /var/tmp/packetcapture.pcap0 -n -vvvv -XX
   ```

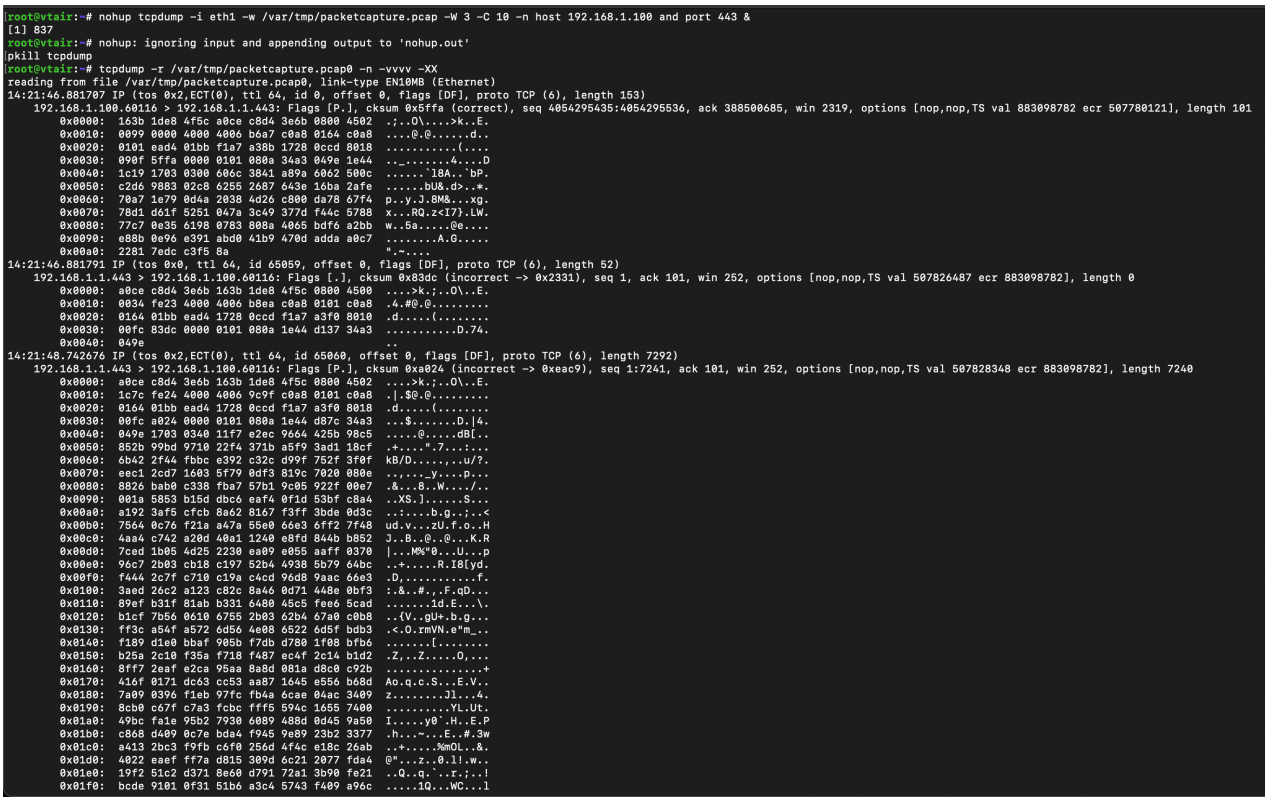


Figure 374: Tcpdump

Templates

Use the templates tool to create a system setup with predefined configuration options for base security, intrusion protection, web filtering, and VPN services.

Find the templates tool under **Tools > Templates**.

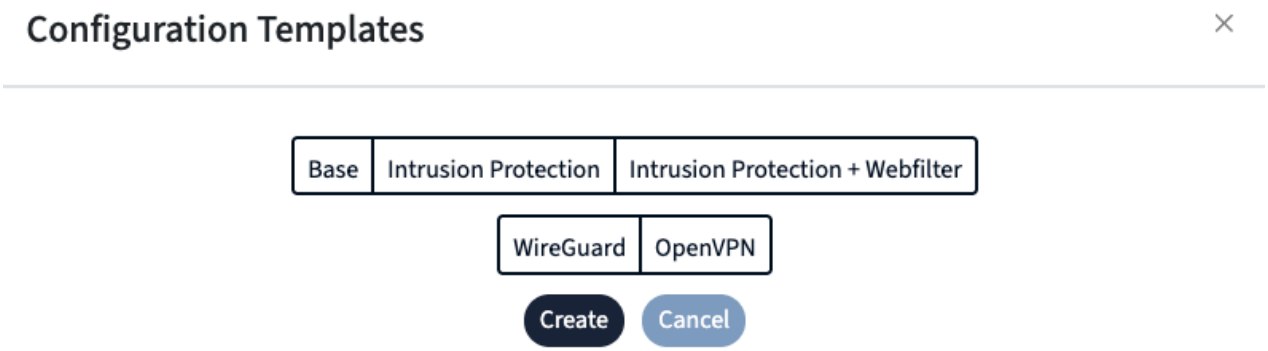


Figure 375: Configuration templates

The templates tool helps you create a TBF setup. Select from the following configuration options:

- **Base**
- **Intrusion Protection**
- **Intrusion Protection + Webfilter**

Additionally, select a VPN setup option:

- **WireGuard**
- **OpenVPN**
- Both options

TBF automatically generates a basic configuration for the selected categories.

Note: You can revert this setup by using this tool again and deselecting the options to remove.

Top

Top is a system monitoring tool that displays real-time CPU and memory utilization information.

Find the top tool under **Tools > Top**.

The result is an ordered list of running processes that updates continuously and shows which processes consume the most system resources.

The browser displays the information in real time.

Traceroute

Traceroute is a network diagnostic tool that displays the route that packets take to reach a destination and measures transit delays across network paths.

The traceroute tool traces the path that packets take from a device to a destination host. The system sends packets with incrementally increasing time-to-live (TTL) values. Each router along the path decrements the TTL and returns an ICMP `time exceeded` message when the TTL reaches 0. This message reveals the identity and response time of the router.

This diagnostic tool helps you:

- Identify the network path to a destination
- Measure delay at each hop along the route
- Locate network bottlenecks or detected failures
- Troubleshoot routing issues
- Verify network topology changes

Traceroute is useful when you diagnose slow network connections, identify where packets drop, or understand the network infrastructure between a device and a remote host.

Find the traceroute tool under **Tools > Traceroute**. The browser displays the information in real time.

To use the traceroute tool, see *use traceroute* on page 451.

Use traceroute

To display the route and measure delay across a network path to a destination, use the traceroute tool.

Find the traceroute tool under **Tools > Traceroute**.

The browser shows the information in real time.

Perform the following steps:

1. Enter the hostname or IP address of the destination.
2. Set the following options as needed:
   - IP protocol v4 or v6
   - Source address. The default is auto.
   - Maximum TTL (time-to-live): This option represents the maximum number of network hops
   - ICMP ECHO instead of UDP datagrams
3. To start the trace, select **Traceroute**.

The traceroute results appear in real time in the browser and show each hop along the network path with response times.

A successful result looks like this:

```
traceroute to www.wikipedia.de (134.119.24.29), 30 hops max, 60 byte packets
1  HSI-KBW-46-237-206-193.hsi.kabel-badenwuerttemberg.de (46.237.206.193)  5.683 ms  10.471 ms  10.579
 ms
2  ip-81-210-148-240.hsi17.unitymediagroup.de (81.210.148.240)  10.256 ms  10.266 ms  10.228 ms
3  de-fra01b-rc1-ae37-0.aorta.net (84.116.191.173)  16.927 ms  17.086 ms  17.079 ms
4  de-fra01b-ri2-ae30-0.aorta.net (84.116.134.166)  16.781 ms de-fra01b-ri2-ae29-0.aorta.net
 (84.116.134.162)  16.956 ms  16.722 ms
5  213.46.177.10 (213.46.177.10)  16.767 ms  28.383 ms  26.085 ms
6  ffm-bb3-link.telia.net (62.115.120.1)  26.078 ms ffm-bb4-link.telia.net (62.115.120.7)  20.464 ms
 16.672 ms
7  koln-b1-link.telia.net (80.91.247.247)  19.419 ms  104.602 ms  104.479 ms
8  ae0.cr-nashira.cgn4.core.heg.com (62.115.33.242)  39.964 ms  21.363 ms  21.355 ms
9  ae0.sr-jake.cgn1.dcn.heg.com (87.230.114.222)  21.317 ms  21.275 ms  21.227 ms
10  wikipedia.de (134.119.24.29)  22.021 ms  21.997 ms  20.675 ms
```

To identify network issues such as high latency at specific hops, packet loss, or detected routing errors, use traceroute results. Each line shows a hop number, hostname or IP address, and 3 response time measurements in milliseconds.

Wake-on-LAN

To enable or wake up network devices remotely, use the Wake-on-LAN tool. TBF sends network messages.

Find the Wake-on-LAN tool under **Tools > Wake-on-LAN**.

This tool enables or wakes up a host via a network message.

Perform the following steps:

1. Enter the **MAC Address** of the destination device.
2. When the destination is on another network, optionally enter the **IP Address**

   The default is `255.255.255.255`. If the destination is on another network, then specify an IP address.
3. To start the wake process, select **Wake-on-LAN**.

Wake-on-LAN devices

This topic describes how to save and manage devices for convenient Wake-on-LAN operations and store their MAC addresses, IP addresses, and descriptions.

You can also save devices with their **MAC Address**, **IP Address**, and a **Description**.

Under **Actions**, you can manually start Wake-on-LAN for each device.

22. Configuration examples

This topic covers real-world configuration examples for common network scenarios and use cases.

This section provides practical configuration examples for common network scenarios. Each example includes step-by-step instructions and best practices.

Configuration examples:

- *Windows updates* on page 453: Allow only Windows updates through the firewall
- *IPv6 multi-WAN* on page 453: Set up IPv6 with multiple internet service providers
- *Custom scripts* on page 454: Execute custom actions on system events

Windows updates

To allow only Windows updates through the firewall, create a DNS network object with specific Microsoft domains.

To allow only Windows updates, use a DNS network object that includes the following DNS names:

Perform the following steps:

1. Create a DNS network object with the following DNS names:
   - windowsupdate.microsoft.com (Exact Match)
   - *.windowsupdate.microsoft.com (Direct Subdomains)
   - *.update.microsoft.com (Direct Subdomains)
   - *.windowsupdate.com (Direct Subdomains)
   - download.windowsupdate.com (Exact Match)
   - download.microsoft.com (Exact Match)
   - *.download.windowsupdate.com (Direct Subdomains)
   - wustat.windows.com (Exact Match)
   - ntservicepack.microsoft.com (Exact Match)
   - go.microsoft.com (Exact Match)
   - dl.delivery.mp.microsoft.com (Exact Match)
2. Use the network object as the destination in the firewall rule.
3. Disable IPS for the rule in the advanced settings.

IPv6 multi-WAN

Set up IPv6 multi-WAN when TBF connects to multiple internet service providers (ISPs) that each provide an IPv6 address or IPv6 network address range.

Set up IPv6 multi-WAN when TBF connects to multiple ISPs that each provide an IPv6 address or IPv6 network address range.

For implementation details, see *Network Prefix Translation (NPTv6)* on page 453.

Network Prefix Translation (NPTv6)

IPv6-to-IPv6 Network Prefix Translation (NPTv6) enables IPv6 networks to achieve address independence, similar to Network Address Translation (NAT) in IPv4 networks.

IPv6-to-IPv6 NPTv6 is a specification that allows IPv6 networks to achieve address independence. NPTv6 works similarly to how NAT functions in IPv4 networks.

To set up IPv6 multi-WAN, you can follow the same process as IPv4 multi-WAN. The key difference is that to enable Network Prefix Translation (NPT), you define a network range in Source Network Address Translation (SNAT).

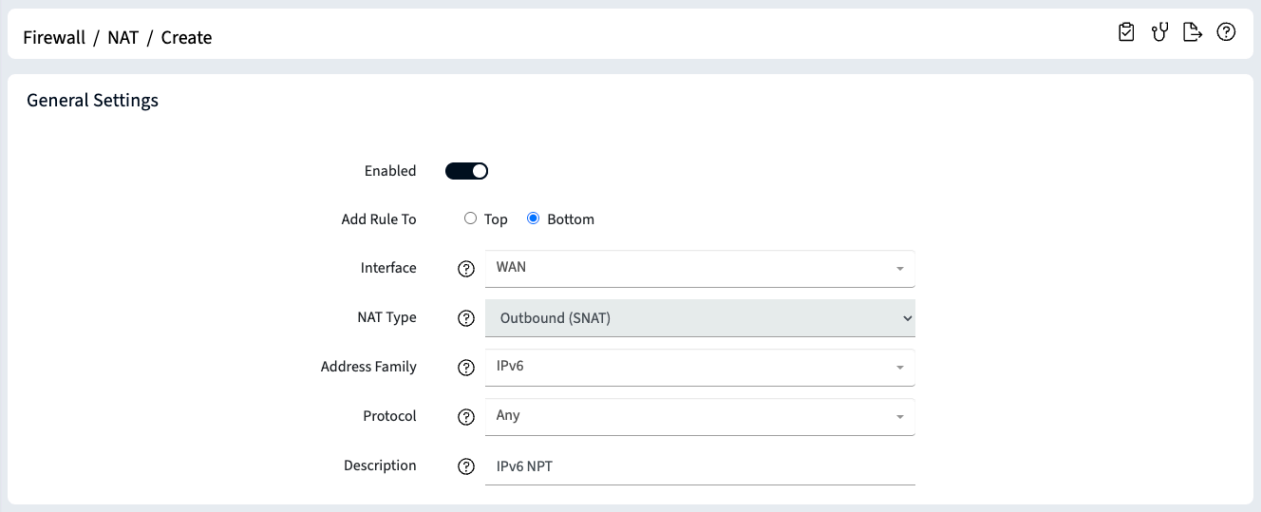


Figure 376: General settings

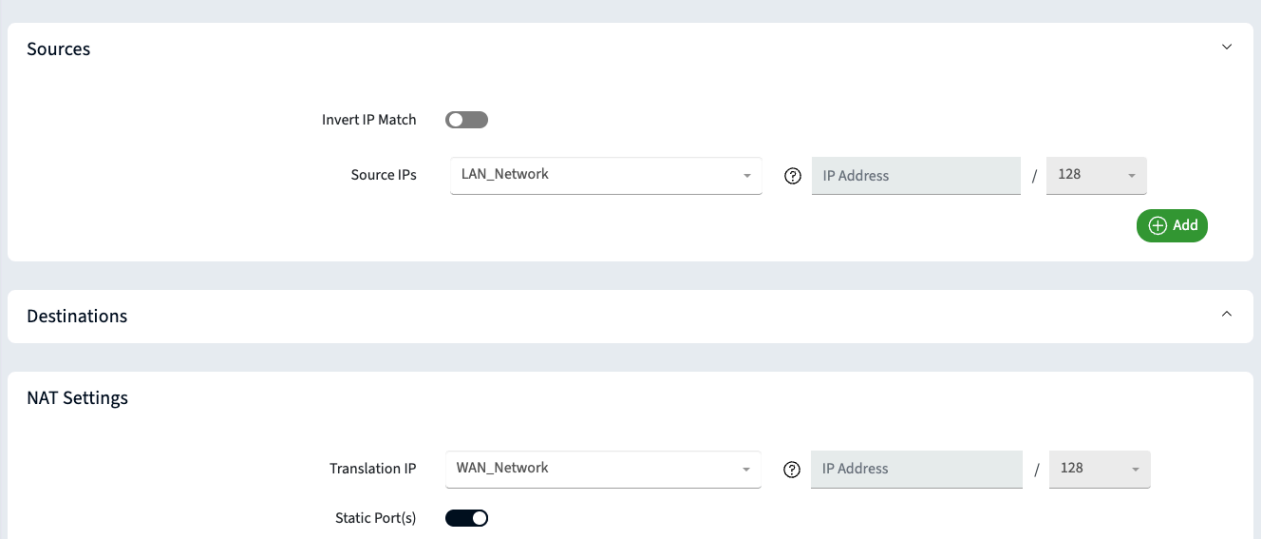


Figure 377: Sources, destinations, and NAT settings

To create a second SNAT rule for `WAN2`, you can follow the same configuration pattern.

To complete the *multi-gateway setup* on page 239, you add the `WAN2` gateway to the **Main** routing table in *routing tables* on page 211.

Custom scripts

To execute actions when system events occur, set up custom scripts.

TBF calls custom scripts on events that happen on the system to execute custom actions.

The following types of notifications are available:

- *System notification* on page 455: Scripts triggered by route, IP address, or link events
- *Gateway change notification* on page 455: Scripts triggered by gateway events

System notification

System notifications trigger custom scripts when routes, IP addresses, or link events occur.

System notifications include the following:

- New routes
- Deleted routes
- IP addresses events
- Link-up events
- Link-down events

TBF calls scripts in the folder `/etc/scripts/syschange`, and passes the event data along as environment variables.

Gateway change notification

Gateway change notifications trigger custom scripts when gateway events occur.

The gateway change notification calls scripts in the folder `/etc/scripts/gatewaychange`, and passes the event data along as environment variables.

23. API

TBF has a REST API for automated configuration management and system integration.

The API enables programmatic access to the device configuration options. The access allows you to automate management tasks across multiple devices and integrate the system with existing network management tools.

The API has the same functionality as the GUI through standardized REST endpoints. The endpoints support `GET`, `POST`, and `DELETE` operations for complete configuration control.

API documentation

Complete API reference documentation is available at *https://docs.vtair.de/api/*.

Getting started

To begin to use the API, start with the general overview and authentication setup:

- *General API overview* on page 456: Introduction to REST API capabilities and usage
- *Token authentication* on page 458: Set up API access that assists in establishing security
- *API token management* on page 81: Manage API tokens for automated access

API usage guides

Learn how to interact with the API with different methods:

- *Console access* on page 457: Command line API access
- *GET requests* on page 457: Retrieve configuration data
- *POST requests* on page 458: Update configuration settings

General

TBF comes with a REST API to automate configuration changes in considerable networks and to save time.

Instead of accessing the GUI of potentially hundreds of devices, you can enter the configuration changes once and send them to a whole list of IP addresses.

The REST API contains the same configuration options as the GUI in a text-based form. You can access the REST API via the command line, a web browser or specialized programs that read and write to REST APIs.

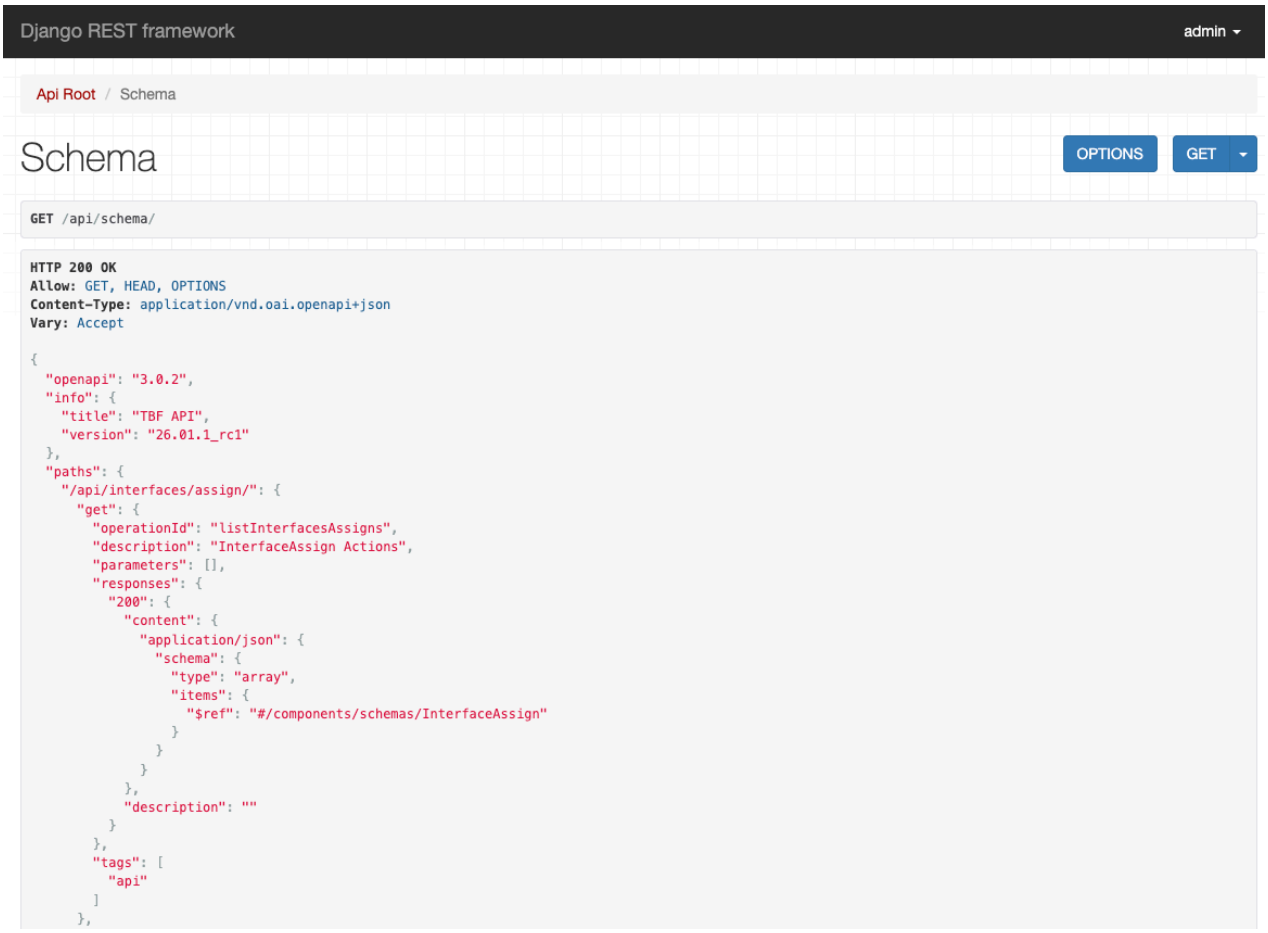


Figure 378: Rest API schema

Via `GET`, `POST` and `DELETE` commands, you can alter the contents of the individual fields. To load the fields within that endpoint, you can execute a `GET` request. For example, you can display TBF users with the request `GET /api/user/`.

When you access the API via a web browser, you can execute `POST` commands directly from the browser window.

The following topics contain detailed information about accessing and using the API:

- *Access the API via a console* on page 457
- *GET requests* on page 457
- *POST requests* on page 458
- *API token* on page 458

Access the API via a console

You can access the REST API through command-line interfaces on Mac, Linux, or Windows systems.

You can access the REST API via the command line, a web browser or specialized programs that read and write to REST APIs. This section covers command line access methods.

GET requests

This topic describes how to use `GET` requests to retrieve data from API endpoints with curl commands.

Mac or Linux GET requests

On a Mac or Linux device, you can enter:

```
curl -k -u USER:PASSWORD https://IPADDRESS/api/user/
```

The command accesses the user data of the TBF device that you reach under the `IPADDRESS` with the user credentials `USER` and `PASSWORD`. The command reads back the configuration string of the users you have set up on the system.

The `-k` option makes the command line ignore that the system does not trust the SSL connection to the device.

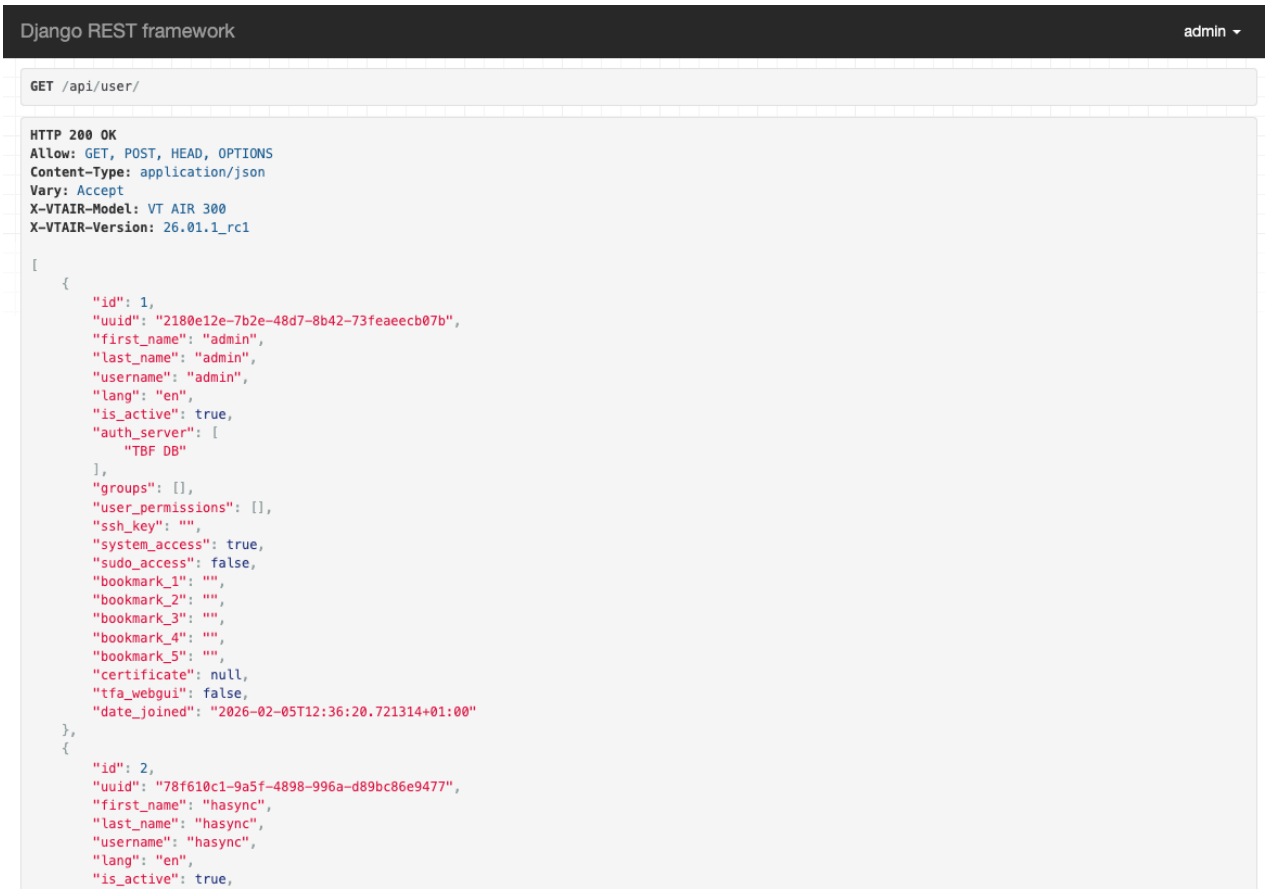


Figure 379: Rest API user data response

POST requests

To write data to a device with curl commands and data parameters, use `POST` requests.

Write data with POST requests

To write data to a device, enter:

```
curl -k -u USERS:PASSWORD -d 'DATA' https://IPADDRESS/api/user/
```

The `DATA` string includes the fields you want to fill separated by `&`. The following example is shortened for readability:

```
'uuid=null&first_name=John&lastname=Doe&username=johnDoe&lang=en'
```

API token

Each user has an API token that they use instead of a username and password for authentication.

Each user has an API token that they use instead of the username and the password. For this, the format of the authorization header is `Token <ACCESS_TOKEN>`:

```
curl -k -H "Authorization: Token <ACCESS_TOKEN>" https://IPADDRESS/api/user/
```

24. Commands

Command-line tools and utilities are available for system management, diagnostics, and network testing.

TBF offers various command-line tools for different purposes:

- *Speedtest* on page 459: Network speed testing tool
- *Command-line tools* on page 459: System maintenance and update tools
- *XDP command-line tools* on page 460: XDP offloader management tools

Each tool has specific functionality to manage different aspects of the system through the command-line interface.

Speedtest

The speedtest command-line tool tests internet bandwidth with speedtest.net.

Description

This tool is a command-line interface to test internet bandwidth with *speedtest.net*.

Usage

```
speedtest.py [-h] [--no-download] [--no-upload] [--single] [--bytes] [--share] [--simple] [--csv]
 [--csv-delimiter CSV_DELIMITER] [--csv-header] [--json] [--list]
               [--server SERVER] [--exclude EXCLUDE] [--mini MINI] [--source SOURCE] [--timeout
 TIMEOUT] [--secure] [--no-pre-allocate] [--version]
```

Options

```
optional arguments:
-h, --help            show this help message and exit
--no-download         Do not perform download test
--no-upload           Do not perform upload test
--single              Only use a single connection instead of multiple. This simulates a typical file
 transfer.
--bytes               Display values in bytes instead of bits. Does not affect the image generated by
 --share, nor output from --json or --csv
--share               Generate and provide a URL to the speedtest.net share results image, not displayed
 with --csv
--simple              Suppress verbose output, only show basic information
--csv                 Suppress verbose output, only show basic information in CSV format. Speeds listed
 in bit/s and not affected by --bytes
--csv-delimiter CSV_DELIMITER
                        Single character delimiter to use in CSV output. Default ","
--csv-header          Print CSV headers
--json                Suppress verbose output, only show basic information in JSON format. Speeds listed
 in bit/s and not affected by --bytes
--list                Display a list of speedtest.net servers sorted by distance
--server SERVER       Specify a server ID to test against. Can be supplied multiple times
--exclude EXCLUDE     Exclude a server from selection. Can be supplied multiple times
--mini MINI           URL of the Speedtest Mini server
--source SOURCE       Source IP address to bind to
--timeout TIMEOUT     HTTP timeout in seconds. Default 10
--secure              Use HTTPS instead of HTTP when communicating with speedtest.net operated servers
--no-pre-allocate     Do not pre allocate upload data. Pre allocation is enabled by default to improve
 upload performance. To support systems with insufficient memory, use this
                        option to avoid a MemoryError
--version             Show the version number and exit
```

Command-line tools

System maintenance and update tools are available through the command-line interface.

TBF has several command-line tools for maintenance, updates, and diagnostics:

- *logcleanup* on page 460: Clean up log files that are too large
- *check-routes* on page 460: Check routes and gateways

logcleanup

logcleanup is a command-line tool to clean up log files that are too large.

TBF runs this tool periodically.

Usage

```
logcleanup.py
```

check-routes

check-routes is a command-line tool to check routes and gateways.

Usage

```
check-routes
```

XDP command-line tools

These command-line tools manage and show options of the XDP offloader feature.

Use XDP command-line tools to manage and show options of the XDP offloader feature.

The following XDP tools are available:

- *xdp_vtair_user* on page 460: Tool to attach the XDP program and to set the blacklist
- *xdp_stats* on page 461: Tool to read out the statistics of XDP

xdp_vtair_user

xdp_vtair_user attaches the XDP program and sets the blacklist.

Usage

```
xdp_vtair_user [options]
```

Options

```
DOCUMENTATION:
 XDP flowoffload
- Allows selecting BPF section --progsec name to XDP-attach to --dev

Required options:
-d, --dev <ifname>         Operate on device <ifname>

Other options:
-h, --help                 Show help
-F, --force                Force install, replacing existing program on interface
-U, --unload               Unload XDP program instead of loading
-M, --reload-maps          Recreate pinned maps
-B, --blacklist            Reload Blacklist
-b, --blacklistread        Read and output Blacklist
-O, --option               Option Setting mode required for enable or disable
-E, --enable               Enable XDP Forward mode on loaded XDP program
-D, --disable              Disable XDP Forward mode on loaded XDP program
    --filename <file>      Load program from <file>
    --progsec <section>    Load program in <section> of the ELF file
```

xdp_stats

xdp_stats reads out the statistics of XDP.

Usage

`xdp_stats [options]`

Options

```
DOCUMENTATION:
 XDP stats program
- Finding xdp_stats_map via --dev name info

Required options:
-d, --dev <ifname>         Operate on device <ifname>

Other options:
-h, --help                 Show help
-b, --blacklist            Blacklist
-C, --clear                Clear all values in xdp_stats_map or blacklist
```

25. Roadmap

This page lists planned features and development priorities.

- **MPTCP**: Multipath TCP offers multiple connections between two hosts and also adjusts the send rate and balances the degree of congestion within each subflow to improve throughput or resource utilization. We want to explore the ability to connect two TBFs with MPTCP for SDWAN.

- **Dynamic charting framework**: A dynamic charting framework would allow us to chart arbitrary data from the OS that can be read like states, CPU utilization, and so on.

- **Web filter connection to intrusion protection**: We want to forward decrypted web filter traffic to the intrusion protection engine. This would allow us to also check encrypted web traffic that is decrypted by the web filter against the IPS ruleset.

26. Changelog

This page lists new features and changes for each TBF release.

View each release for new features, improvements, bug fixes, and other changes.

TBF 26.10

This topic lists fixes and improvements included in TBF 26.10.

New features

- The **Update** view has been redesigned with separate online and offline update flows. Version checks and image installation now use `tbf-swupdate` (SWUpdate) instead of `apt`.
- Vendor-specific IEC104 Type IDs (128–255) are now seeded for the enforcer in the NETCONF model.
- Addon activation states are now saved in the NETCONF database.
- The Addon API now exposes a readiness status. Enable requests support a synchronous wait option, while the default asynchronous behavior remains unchanged.
- The WebUI factory reset now triggers the on-device factory-reset mechanism on TBF.

Improvements

- Input-chain firewall rules now respect configured rate limits.
- SNMPv3 users now support SHA-2 authentication hash variants.
- The system-time REST API responses are now documented, and input validation has been strengthened.

Bug fixes

- DNS host overrides no longer leave a duplicate automatically generated hostname record.
- The Modbus quantity limit is no longer included in the generated enforcer (DPI) configuration.
- HAProxy stick-table rate limiting works correctly again.
- Interfaces now restore autonegotiation after fixed link settings are cleared.
- Detected proxy upstream connection errors now return a clear response in the iframe instead of an unhandled detected server error.
- The HAProxy diagnostics page and disabled-backend handling have been corrected.
- Audit-log actor information is now serialized correctly in diagnostics.
- The ENIP enforcer configuration no longer produces an invalid class ID value.
- Enabling an enforcer now reloads the main `nftables` ruleset.
- Interface diagnostics now include DHCP lease information.
- Certificate imports are now atomic and pre-validated.
- OpenVPN PKCS#12 export works correctly.
- Incompatible Docker image architectures are rejected when creating app containers.
- Punctuation is allowed in Docker environment values.
- Stale physical interfaces no longer survive a configuration restore.
- Bridge firewall rules without an interface are rejected.
- A protected-delete operation that was unsuccessful no longer affects related configuration.
- Firewall rule expiry dates display correctly when editing under a non-English locale.
- GeoIP classification is fixed for aliases named `_GEO_IP_`.
- The virtual IP interface-address check no longer reports spurious detected errors.

Security

- Firewall log-analysis queries now use validated and parameterized input to help prevent SQL injection.
- Unauthenticated captive-portal clients can no longer bypass captive-portal enforcement through web-proxy DNAT.
- User input in the WAF ruleset template is correctly HTML-escaped.
- Syslog CA validation is now conditional, and the `rsyslog` TLS CA configuration is corrected.
- LDAP over TLS enforces TLS 1.2.
- Configured SNMPv3 users require authentication and privacy (`authPriv`).
- Redirect targets are validated to reject unauthorized cross-site redirects across the web interface and REST API.
- LLDP advertisements are minimized to reduce information disclosure, and LLDP is no longer advertised on WAN interfaces.

- Logout now helps protect against cross-site requests.
- Diagnostic console sessions now help protect against cross-site hijacking.
- State-changing copy and convert actions require a POST request and retain CSRF protection.
- OpenVPN and IPsec diagnostic action and log input are validated.
- PKCS#12 certificate export works with passwords containing special characters.
- A configuration save or restore that was unsuccessful no longer leaves the system in a partial state (fail-closed).
- TOTP device secrets are no longer exposed through the TOTP device API.

Infrastructure and documentation

- The VT AIR Portal connector has been removed.
- `nftables` reloads use idle CPU scheduling to reduce their impact on other system activity.

TBF 26.07

This topic lists fixes and improvements included in TBF 26.07.

New features

- Add-on API readiness and synchronous operations: add-ons now expose a `ready` status, and enable/disable requests support a synchronous `wait` option. The default asynchronous behavior is unchanged.
- Learning Mode data REST API for creating firewall rules from learned traffic, including the appropriate bridge firewall rule for traffic learned on bridge interfaces.
- Configurable IPv4 Reverse Path Filtering (uRPF) with **Off**, **Strict**, and **Loose** modes in the GUI, REST API, and NETCONF/YANG model.
- Administrators can mark the Setup Wizard as completed through the settings command REST API.
- License information now includes the license end date and device ID in the license view, dashboard, and REST API.

Improvements

- Firewall and enforcer validation now rejects unsupported **Any** source, destination, and port matches, and enforces the required Modbus destination port.
- Disabled Modbus rules are excluded from the generated DPI configuration, and the Modbus quantity limit is no longer emitted. Register address ranges remain enforced.
- Periodic DNS hostname changes now reliably invalidate caches and update generated `nftables` rules. Stale GeoIP entries are removed when rules are deleted or disabled.
- Route-change custom-script events are queued and batched so that no events are lost during bursts.
- Add-on enable/disable operations are more consistent and no longer leave services stopped or remove services after a rapid re-enable.
- NETCONF XML backups are correctly detected, and VT AIR JSON backups are automatically converted to TBF NETCONF XML on NETCONF-enabled systems.

Bug fixes

- Unicast VRRP advertisements now use the configured synchronization interface, while virtual IP addresses remain bound to their configured interface.
- Disabled HAProxy backends are no longer referenced in generated frontend directives.
- The HAProxy diagnostics page now displays a message when the service is disabled, instead of returning a detected error.
- HAProxy unavailable proxy upstreams now return a clear `502` response instead of a detected server error.
- Bridge firewall rules preserve connection-tracking marks, and creating a disabled firewall rule with an assigned enforcer correctly triggers the enforcer update.
- DNS host overrides no longer produce duplicate automatically generated hostname records.
- Multiple-use captive portal vouchers now display their actual usage and availability.
- Learning Mode no longer crashes when the firewall event table is empty, and repeated event-log entries are deduplicated.

  Note: A permission-controlled action is available to clear the event log.

- OpenVPN preserves explicit `udp4` and `tcp4` settings for hostname-based remote endpoints.
- Certificate chain validation correctly handles CA bundles that contain multiple PEM certificates. Let's Encrypt full chains retain all intermediate and root certificates, and intermediate certificates are included during HA synchronization.
- Encrypted backup filenames retain the `.encrypted` suffix, and encrypted XML backups restore correctly.
- Namespace-qualified NETCONF migration data is handled correctly, which helps prevent duplicate IDs and aliases during migration.
- WAF rule updates work correctly when the target rule directory already exists.

Security

- Firewall diagnostic SQL queries now use validated and parameterized input to help prevent SQL injection.
- Unauthenticated captive-portal clients can no longer bypass captive-portal enforcement through web-proxy DNAT.
- REST API CSRF handling is corrected for deployments that use dynamically assigned IP addresses, while retaining CSRF validation.
- Authenticated API write requests received while the system is starting now return HTTP `503` with `Retry-After: 5`.

Note: Clients should retry after initialization completes.

Infrastructure and documentation

- The in-application manual has moved from locally built Sphinx documentation to a pre-built DITA manual for TBF. You can navigate the documentation within the existing modal.
- The online documentation link now points to the Hirschmann documentation portal: *https://www.doc.hirschmann.com/index.html*.
- `nftables` reloads use idle CPU scheduling to reduce their impact on other system activity.
- PostgreSQL locale settings are corrected for Yocto-based systems.
- Audit-log routing and rotation are corrected, including audit logging in staging mode.

TBF 26.04

This topic lists fixes and improvements included in TBF 26.04.

New features

- Learning Mode data REST API for firewall
- Custom 502 error page during service startup (Nginx)
- `TTY` timeout for console login sessions
- Container app management improvements
- Online activation improvements with centralized cache and UI refresh
- Comments on bridge firewall rules

Bug fixes

- Enforcer traffic no longer bypassed by flow offloading
- Bridge rules now save marks to `conntrack` (enforcer persistence)
- Netconf data loss on stale connection reconnect
- Netconf m2m signal firing when netconf disabled
- `BPF` map file descriptor leak
- Licence tier detection (IPS feature name mismatch)
- Container create form `KeyError`
- Firewall `nftables` queue syntax on single-CPU systems
- `Conntrack` zone handling for localhost/firewall traffic
- Netconf atomic unlock on exception
- Fix migration crash on namespaced startup config

Infrastructure

- Getty `systemd` restart hardening
- `XDP` disabled for TBF 240/241 models
- Dashboard hides `-dirty` suffix from U-Boot version
- Netconf atomic decorator for NETCONF bulking in views

TBF 26.01

This topic lists fixes and improvements included in TBF 26.01.

Firewall

- Fix route rule generation and double OR in prerouting
- Add firewall traffic to custom `ct` zone
- Allow all firewall rules to enable log files

VPN/IPsec

- Fix phase 1 fallback check
- Fix HASync settings URL

OpenVPN

- Remove unsupported ciphers and digests
- Fix `remote` directive protocol for IPv4 addresses

Certificates

- Accept PKCS#8 private keys in Let's Encrypt validation
- Allow import with key length 3072
- Fix `save_vtair_cert` for NETCONF backend

User management

- Fix Linux password hashing for new users
- Fix TOTP device creation to generate key when none provided
- Fix Linux user rename to also move home directory

Interfaces

- Fix `HWInterface` reassignment during `force_clean`
- Add OS version check for IAF-240 bridge configuration
- Check `ethtool` subattributes for fixed offload settings

Portal/UI

- Show renewal link when days left are displayed
- Fix expiration display string formatting
- Show BeldenOS DISTRO version in dashboard
- Improve form submission and update page UX
- Separate search and notification into individual nav items

NETCONF backend

- Harden retry, reconnect, and tree merge logic
- Add `netconf_atomic` bulk wrapper for batch operations
- Fix deletion with `SET_NULL` and improve process lock timeout
- Fix `FieldTracker` crash during dill deserialization

Gateway monitoring

- Fix gateway check reload on state flush

Captive portal

- Optimize duplicate interface validation query

Suricata

- Fix config warnings

Backup/restore

- Fix restoring TBF backup via UI
- Add `multipart/form-data` support to backup restore API

Misc

- Fix debug report crash from incorrect database dump
- Bulk permission assignment instead of per-item add

Technical support

Technical questions

For technical questions, please contact any Hirschmann dealer in your area or Hirschmann directly.

You find the addresses of our partners on the Internet at *www.belden.com*.

For technical support, visit *my.belden.com*. This site also includes a free of charge knowledge base and a software download section.

Technical Documents

The current manuals and operating instructions for Belden products are available at *doc.hirschmann.com*.

Customer Innovation Center

The Customer Innovation Center is ahead of its competitors on three counts with its complete range of innovative services:

- Consulting incorporates comprehensive technical advice, from system evaluation through network planning to project planning.
- Training offers you an introduction to the basics, product briefing and user training with certification. You find the training courses on technology and products currently available at *www.belden.com/solutions/customer-innovation-center*.
- Support ranges from the first installation through the standby service to maintenance concepts.

With the Customer Innovation Center, you decide against any compromise in any case. Our client-customized package leaves you free to choose the service components you want to use.

HIRSCHMANN

A BELDEN BRAND