mGuard Configuration Examples

Configuration Examples

UM EN MGUARD CONFIG

PHOENIX CONTACT
INSPIRING INNOVATIONS

Configuration Examples

mGuard Configuration Examples

UM EN MGUARD CONFIG, Revision 01 2019-03-01

This user manual is valid for mGuard security appliances.

108392_en_01

PHOENIX CONTACT GmbH & Co. KG • Flachsmarktstraße 8 • 32825 Blomberg • Germany
phoenixcontact.com

Table of contents

1 Safety instructions

Read this user manual carefully and keep it for future reference.

1.1 Labeling of warning notes

This symbol together with the **NOTE** signal word alerts the reader to a situation which may cause damage or malfunction to the device, hardware/software, or surrounding property.

Here you will find additional information or detailed sources of information.

1.2 Qualification of users

The use of products described in this user manual is oriented exclusively to:

– Qualified electricians or persons instructed by them. The users must be familiar with the relevant safety concepts of automation technology as well as applicable standards and other regulations.
– Qualified application programmers and software engineers. The users must be familiar with the relevant safety concepts of automation technology as well as applicable standards and other regulations.

1.3 Note on the usage of Application Notes

The provided Application Notes are a free service from Phoenix Contact. The examples and solutions shown are not customer-specific solutions, but general support for typical application scenarios. The Application Notes are not binding and do not claim to be complete.

A quality check of the Application Notes takes place but is not comparable with the quality assurance of commercial products. Errors, functional and performance deficiencies cannot be excluded.

To avoid malfunctions/misconfigurations and associated damage, the proper and safe use of the product/software is the sole responsibility of the customer and must comply with the applicable regulations. The customer must check the function of the examples described and adapt them to the individual, customer-specific requirements of the system or application scenario.

The IP settings in the Application Notes have been chosen as examples. In a real network scenario, these IP settings must always be adjusted to avoid address conflicts.

The information in the Application Notes is checked regularly. If corrections are necessary, they will be included in the subsequent revision. Users will not be notified.

2 Creating additional internal/external routes

Document ID: 108409_en_00
Document designation: AH EN MGUARD ADDITIONAL INT ROUTES

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes how to use additional internal routes to enable access from one network to another.

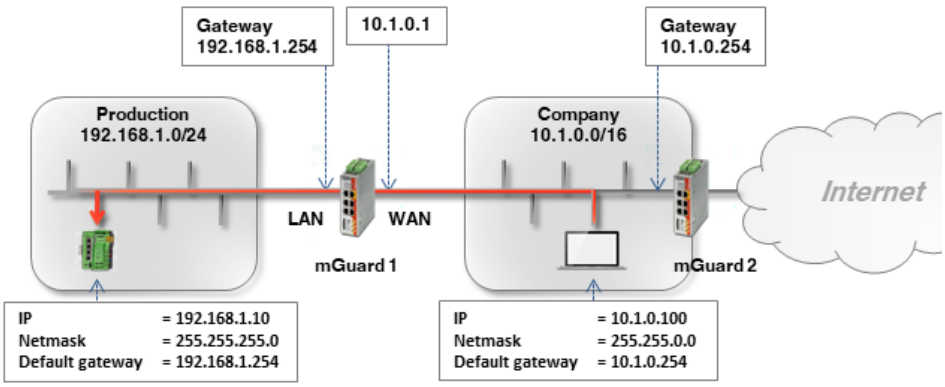
Use of additional external routes is along the same lines as internal routes and will not be described separately.

2.1 Introduction

If packets in the internal network of the gateway (*mGuard 2*) are to be sent to an IP address in another network (external or DMZ), the gateway must know which router or gateway it should use to forward these packets. *Additional Internal Routes* can therefore be specified in the gateway (*mGuard 2*). (Further options are described in Section 4 and 5.)

2.2 Example

The web interface of a machine controller (PLC) in the production network is to be accessed from the company network.



The PLC (192.168.1.10) and the office computer (10.1.0.100) are not in the same network. The office computer sends packets intended for the PLC to its default gateway (*mGuard 2*: 10.1.0.254).

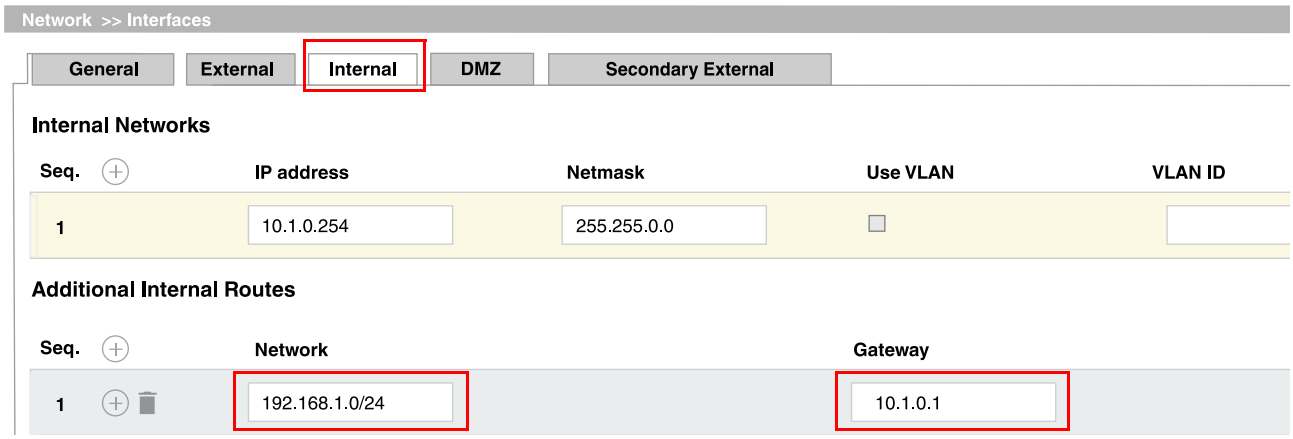
This gateway now needs to know where it should forward the packet to. This is specified by adding additional internal routes.

An additional route must be configured on the default gateway (*mGuard 2*: 10.1.0.254) of the office computer. This route specifies *mGuard 1* (10.1.0.1) as gateway and the production network (192.168.1.0.0/24) as destination network. *mGuard 1* acts as the router that connects the two networks.

2.3 Procedure

If the default gateway in the company network is an mGuard device (*mGuard 2* in the *Router* network mode), proceed as follows:

1. Log into the default gateway web interface (*mGuard 2*) in the company network (LAN interface at 10.1.0.254).
2. Go to **Network >> Interfaces >> Internal**.
3. Create an **additional internal route** to the production network (network: 192.168.1.0/24 via gateway 10.1.0.1):



4. Clients in the company network send packets intended for the network 192.168.1.0/24 via their standard gateway (*mGuard 2*) to *mGuard 1*.

Result

Clients in the company network can reach the PLC in the production network via its real IP address:

– Web browser: http://192.168.1.10
– Ping: 192.168.1.10

	The incoming rules of the *mGuard 1* firewall must allow corresponding requests.

Advantages

– The PLC can be reached directly via its real IP address.
– There is no need to change the network configuration of the office computer and other clients in the company network.

Disadvantages

– Additional routes have to be configured on the gateway.

3 Using network address translation (1:1 NAT)

Document ID: 108407_en_00
Document designation: AH EN MGUARD NAT

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes the basic use of 1:1 NAT. A description of how to access two internal networks from an external network as well as how to access an external network from an internal network is provided.

3.1 Introduction

Using NAT (*Network Address Translation*), the address information in data packets is replaced with other address information or overwritten in order to be able to connect different networks together.

mGuard devices support the NAT procedures: *IP masquerading* and *1:1 NAT*. Use of NAT in VPN connections is also possible (see Section 13).

IP masquerading

With *IP masquerading* enabled, the mGuard device masks the IP address of senders, e.g. from the production network (= *internal network*) with its own external IP address.

1:1 NAT

1:1 NAT maps the IP addresses of a *Real network* to IP addresses of a *Virtual network*. Devices in the *Real network* can therefore be accessed directly via their assigned (*mapped*) IP addresses from the *Virtual network*.

Depending on the netmask specified in the 1:1 NAT configuration, the entire *Real network* or corresponding subnets can be mapped to the *Virtual network*.

3.2 Important information on the use of NAT

1:1 NAT is not supported in the *Stealth* network mode.

The IP addresses specified under "*Virtual network*" must be free. They must not be assigned to other devices, because an IP address conflict would otherwise occur in the *"Virtual network"*. This is even the case if a device corresponding to an IP address in the specified "*Virtual network*" does not exist at all in the "*Real network*".

With 1:1 NAT, the *network part* of an IP address is rewritten (*mapped*) and the *host part* usually remains unchanged. The network part of the IP address is prescribed by the specified netmask.

The same netmask that is used by the *Virtual network* must not be used at the same time to map the *Real network* to the virtual location. In this case, the mGuard would respond to all ARP requests from the *Virtual network*, therefore rendering it unusable.

The specified netmask must be smaller than that used by the *Virtual network*.

If access is to be limited, corresponding firewall rules must be created.

3.3 Example 1: Mapping IP addresses (1:1 NAT)

3.3.1 Individual devices in the production network are to be accessed from the company network

Individual devices in two production networks (with the same network settings) are to be accessible from the company network via 1:1 NAT.

To do this, the *real* IP address of a client in the production network is rewritten (*mapped)* as a *virtual* IP address in the company network. The assigned client in the production network can be accessed directly via this *virtual* IP address.

(If access is to be limited, corresponding firewall rules must be created.)

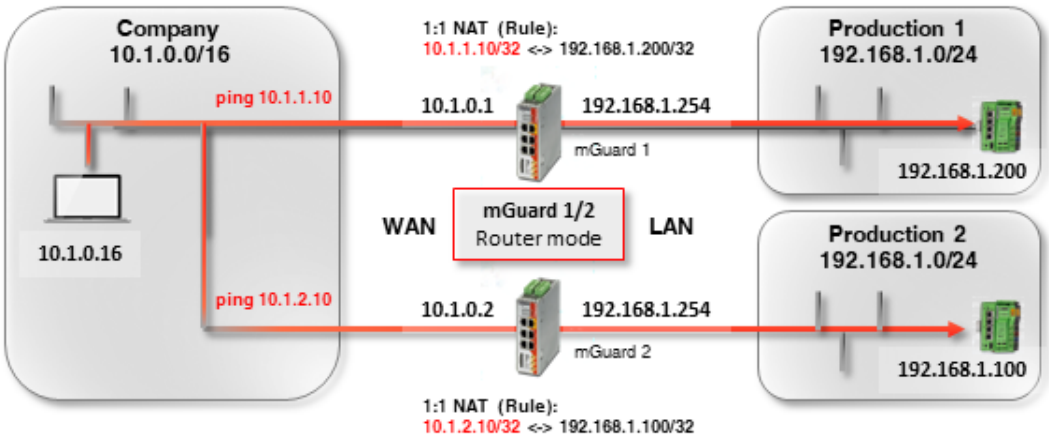


Figure 3-1 1:1 NAT rule: accessing individual IP addresses in the production network from the company network

The ARP *daemon* on the mGuard device will respond to ARP requests sent to the assigned IP addresses in the *Virtual network*. No IP changes may therefore be made in the *Virtual network*.

Table 3-1 Example rules for 1:1 NAT with the netmask 32 (IP address mapping)

Real network	Virtual network	Netmask	Assigned IP addresses
192.168.1.200	10.1.1.10	**32**	192.168.1.200 <-> 10.1.1.10

3.3.2 mGuard device settings

To allow access to devices in the production network from the company network using 1:1 NAT, proceed as follows:

1. Log into the *mGuard 1* web interface.
2. Go to **Network >> NAT**.
3. Configure the 1:1 NAT rules in accordance with Figure 3-2.

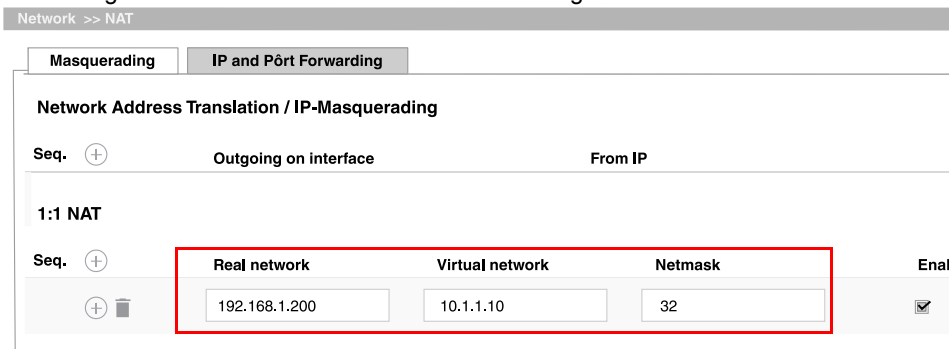


Figure 3-2 *mGuard 1*: Accessing production 1 (IP addresses)

1. Log in to the *mGuard 2* web interface.
2. Go to **Network >> NAT**.
3. Configure the 1:1 NAT rules in accordance with Figure 3-4.

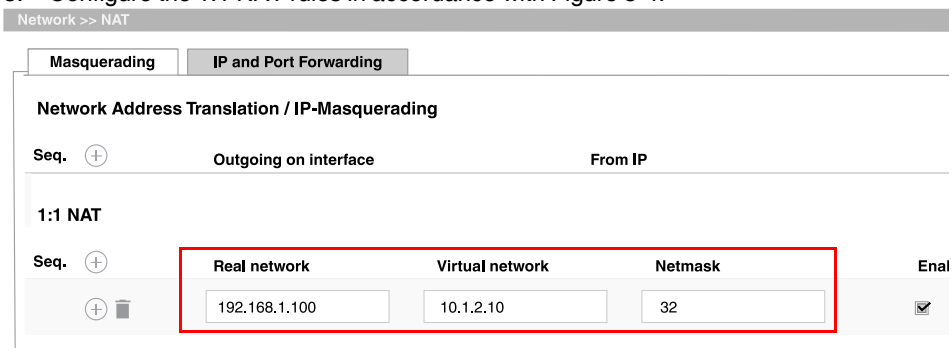


Figure 3-3 *mGuard 2*: Accessing production 2 (IP addresses)

Result

Network packets sent from the company network to the *virtual* IP address 10.1.1.10 are forwarded to the *real* IP address 192.168.1.200 in the production network 1.

Network packets from the company network to the *virtual* IP address 10.1.2.10 are forwarded to the *real* IP address 192.168.1.100 in the production network 1 via mGuard 2.

3.4 Example 2: Mapping networks (1:1 NAT)

3.4.1 The entire production network is to be accessed from the company network

Two production networks with the same network settings are the be accessed from the company network via 1:1 NAT.

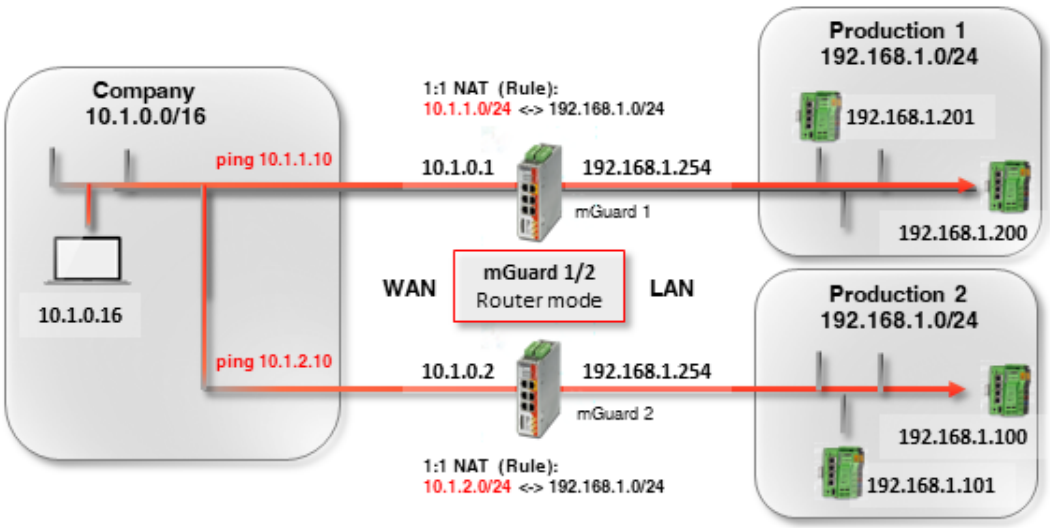


Figure 3-4 1:1 NAT rule: Accessing the entire production network from the company network

The two mGuard devices have external IP addresses that belong to the external company network (10.1.0.1 and 10.1.0.2).

Systems of **production location 1** are to be accessed from the company network via the *Virtual* network **10.1.1.0/24** and systems of **production location 2** are to be accessed via the *Virtual* network **10.1.2.0/24** using 1:1 NAT.

Real clients in the company network may not use an IP address from the *virtual* networks.

Table 3-2 Examples of rules for 1:1 NAT with different netmasks and resulting assignments

Real network	Virtual network	Netmask	Assigned IP addresses
192.168.1.0	10.1.0.0	**24**	192.168.1.0 <-> 10.1.0.0 192.168.1.1 <-> 10.1.0.1 … 192.168.1.254 <-> 10.1.0.254 192.168.1.255 <-> 10.1.0.255

The respective ARP daemon on the two mGuard routers ensure that clients in the external network know where to send packets addressed to the networks 10.1.1.0/24 and 10.1.2.0/24.

3.4.2 mGuard device settings

To make the production network accessible from the company network using 1:1 NAT, proceed as follows:

1. Log into the *mGuard 1* web interface.
2. Go to **Network >> NAT**.
3. Configure the 1:1 NAT rules in accordance with Figure 3-5.

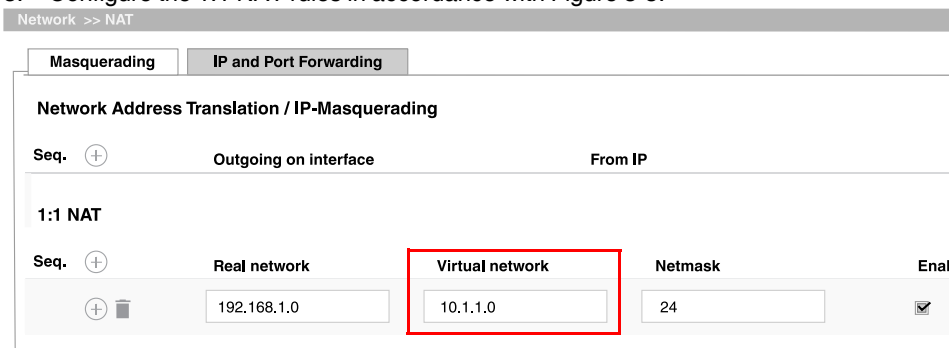


Figure 3-5 *mGuard 1*: Accessing production 1 (networks)

1. Log in to the *mGuard 2* web interface.
2. Go to **Network >> NAT**.
3. Configure the 1:1 NAT rules in accordance with Figure 3-6.

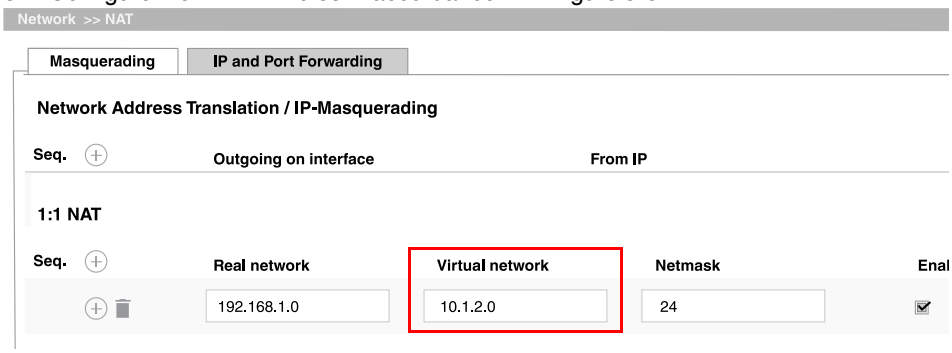


Figure 3-6 *mGuard 2*: Accessing production 2 (networks)

Result

The client 192.168.1.200 in production location 1 can be accessed from the external network via the IP address 10.1.1.200. Client 192.168.1.201 can be accessed via 10.1.1.201.

The client 192.168.1.10 in production location 2 can be accessed via the IP address 10.1.2.10 from the external network; the client 192.168.1.11 can be accessed via the IP address 10.1.2.11, etc.

Clients in production location 2 can in principle also be accessed from production location 1 via their *virtual* IP addresses (10.1.2.0/24), and vice versa.

4 Accessing internal networks (additional routes | IP and port forwarding | 1:1 NAT)

Document ID: 108406_en_00
Document designation: AH EN MGUARD NETWORK SEGMENT 1

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes the use of the mGuard device as a router that connects two networks (internal and external network). The internal network is to be accessed from the external network.

The following procedures are described:

– Option 1: additional internal routes
– Option 2: IP and port forwarding
– Option 3: Network Address Translation (1:1 NAT)

4.1 Introduction

In the "Router" network mode (*Router mode*), an mGuard device can be used to connect two networks. The firewall and VPN security functions are also available (depending on license).

With certain models, a demilitarized zone (DMZ) can be connected via the additional DMZ interface as an option.

4.1.1 Example

The production network (= *internal network*) and the company network (= *external network*) are connected via an mGuard router.

The web interface of a machine controller (PLC) in the production network is to be accessed from the company network. The controller should also respond to a ping request sent to it.

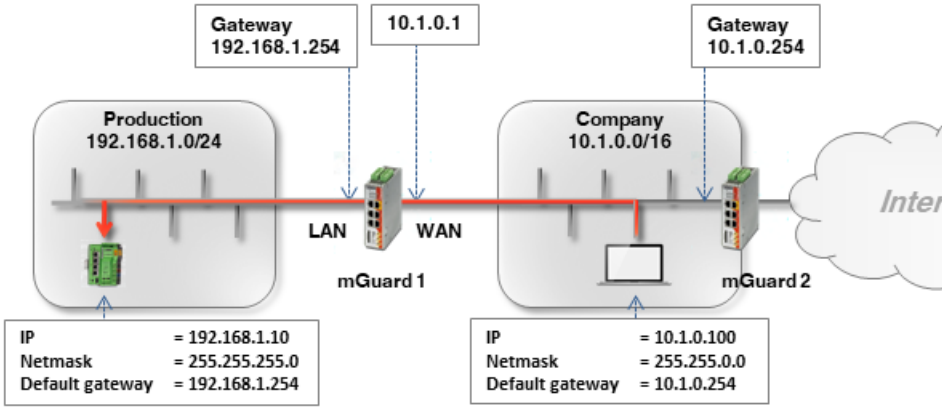


Figure 4-1 Client and mGuard router network settings

The two networks can be connected in various ways:

– Option 1: additional internal routes
– Option 2: IP and port forwarding
– Option 3: Network Address Translation (1:1 NAT)

4.1.2 Procedure

1. Configure the WAN and LAN interface of the router (*mGuard 1*)
2. Configure firewall rules
3. Configure network settings in accordance with option 1, 2, or 3

4.2 mGuard router network settings

To enable network traffic between the two networks, the external interface (= WAN port) and the internal interface (= LAN port) of the *mGuard 1* router must be configured in all options and assigned at least one IP address.

Ensure that the clients in the production and company network are configured in accordance with their network.

The internal IP address of *mGuard 1* must be configured as the default gateway (192.168.1.254) for clients in the production network (PLCs).

The internal IP address of *mGuard 2* must be configured as the default gateway (10.1.0.254) for clients in the company network.

To install *mGuard 1* as the router between the company network (WAN) 10.1.0.0/16 and the production network (LAN) 192.168.1.0.0/24, proceed as follows:

1. Log in to the *mGuard 1* web interface (192.168.1.254).
2. Go to **Network >> Interfaces**.
3. *General* tab: select the **network mode** *Router* and the **router mode** *Static*.
4. *Internal* tab: select 192.168.1.254 as the internal IP address (netmask 255.255.255.0).
5. *External* tab: select 10.1.0.1 as the external IP address (netmask 255.255.0.0).

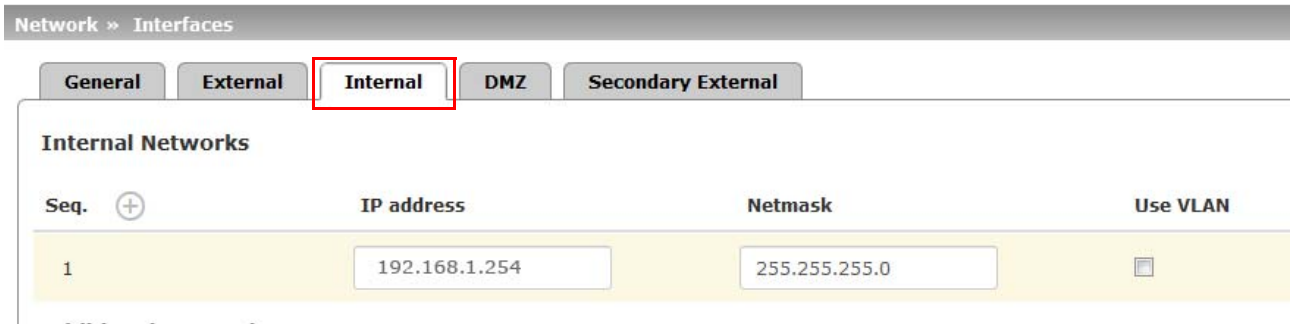


Figure 4-2 Internal interface

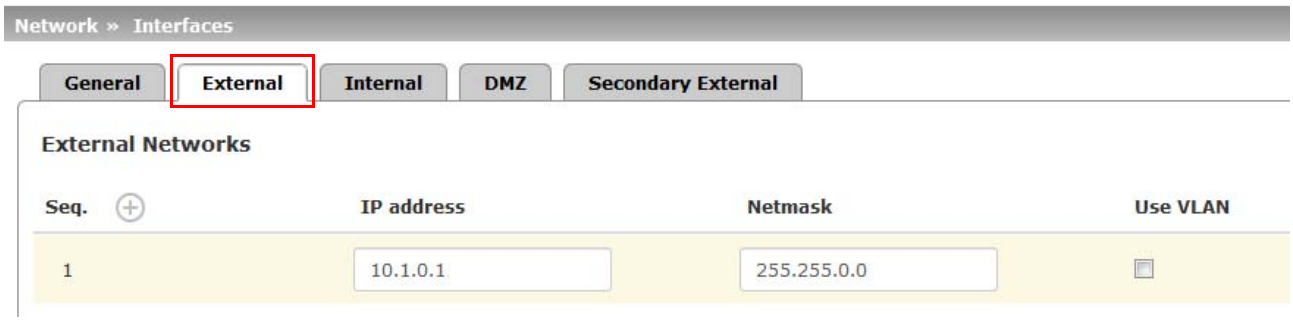


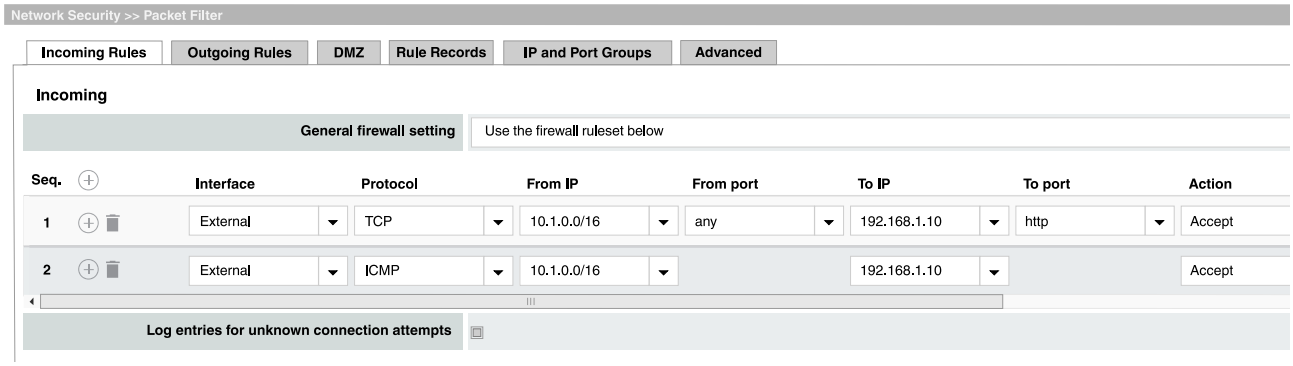
Figure 4-3 External interface

4.3 Configuring firewall rules

mGuard 1 is to be configured to allow the HTTP access to the web interface of the PLC (192.168.1.10) from the company network (= external network: 10.1.0.0/16). In addition, it should also be possible to "*ping*" the controller (ICMP request).

Proceed as follows:

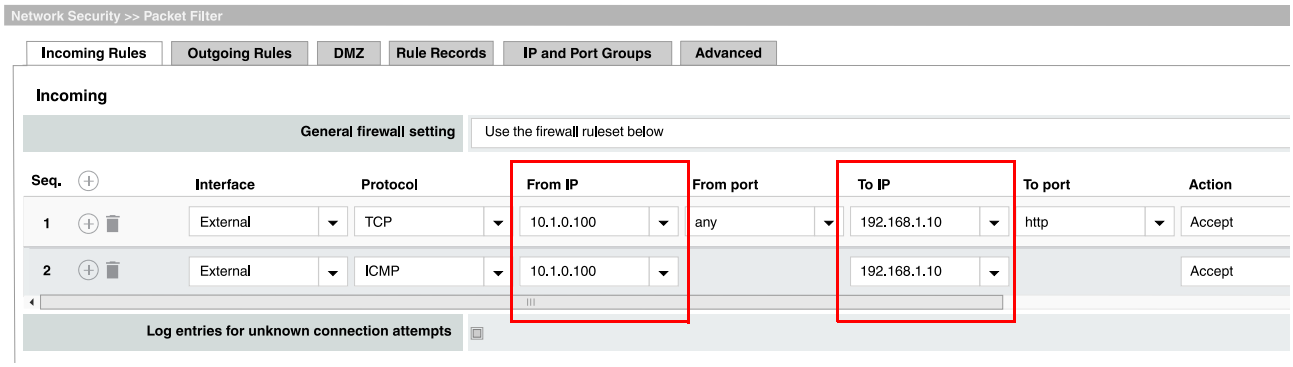
1. Log in to the *mGuard 1* web interface (192.168.1.254).
2. Go to **Network Security >> Packet Filter >> Incoming Rules**.
3. Select "*Use the firewall ruleset below*" under **General firewall setting**.
4. Create two firewall rules as follows:



Result

The firewall rules allow incoming TCP packets to the HTTP port and incoming ICMP packets from the company network to the IP address of the PCL. All other packets are rejected by the firewall.

As an option, the **From IP** and **To IP** fields can also be used to limit access to certain clients (e.g. from **10.1.0.100** to **192.168.1.10**).



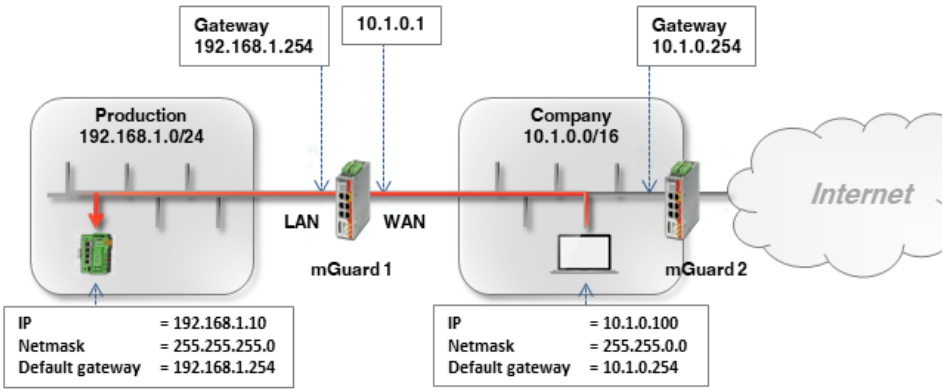
4.4 Network settings in accordance with option 1, 2, and 3

4.4.1 Option 1: additional internal routes on the gateway

The PLC (192.168.1.10) and the office computer (10.1.0.100) are not in the same network. The office computer sends packets intended for the PLC to its default gateway (*mGuard 2*: 10.1.0.254).

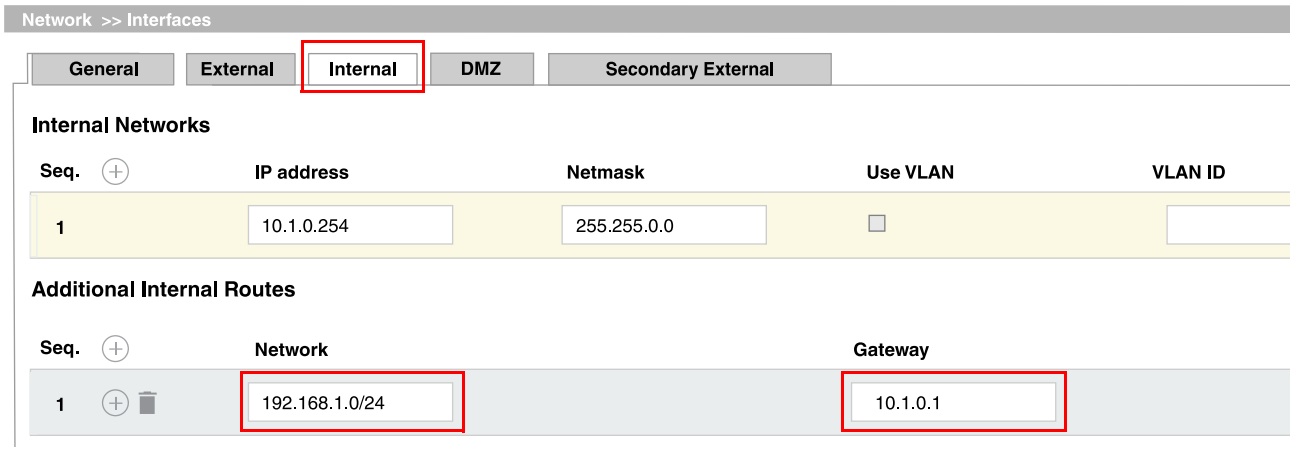
This gateway now needs to know where it should forward the packet to. This is specified by adding additional internal routes:

An additional route must be configured on the default gateway (*mGuard 2*: 10.1.0.254) of the office computer. This route specifies *mGuard 1* (10.1.0.1) as gateway and the production network (192.168.1.0.0/24) as destination network. *mGuard 1* acts as the router that connects the two networks.



If the default gateway in the company network is an mGuard device (in this case *mGuard 2*), proceed as follows:

1. Log into the default gateway web interface (*mGuard 2*) in the company network (LAN interface at 10.1.0.254).
2. Go to **Network >> Interfaces >> Internal**.
3. Create an **additional internal route** to the production network (network: 192.168.1.0/24 via gateway 10.1.0.1):



4. Clients in the company network send packets intended for the network 192.168.1.0/24 via their standard gateway (*mGuard 2*) to *mGuard 1*.

Result

Clients in the company network can now reach the PLC in the production network via its real IP address:

– Web browser: http://192.168.1.10
– Ping: 192.168.1.10

Advantages

– The PLC can be reached directly via its real IP address.
– There is no need to change the network configuration of the office computer and other clients in the company network.

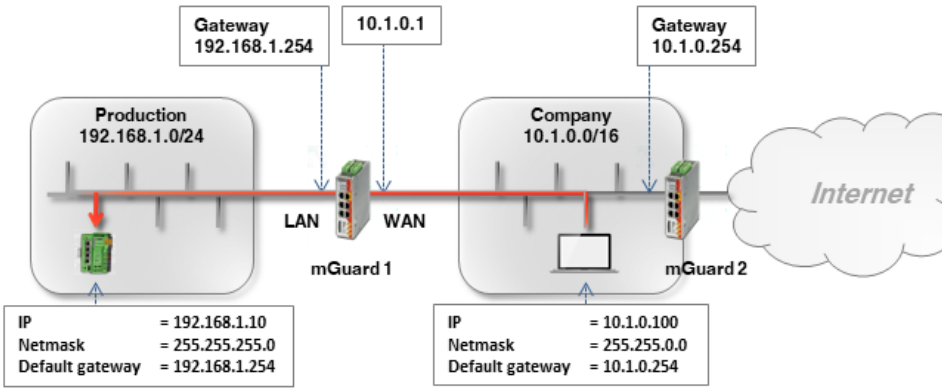
Disadvantages

– Additional routes have to be configured on the gateway.

4.4.2 Option 2: IP and port forwarding

With IP and port forwarding, the IP address and port number is in the header of the incoming data packets is rewritten so that the data packets sent to the external IP address of *mGuard 1* are forwarded to a chosen IP address and/or port number in the internal network.

The PLC (192.168.1.10) is not in the same network as the requesting office computer (10.1.0.100).



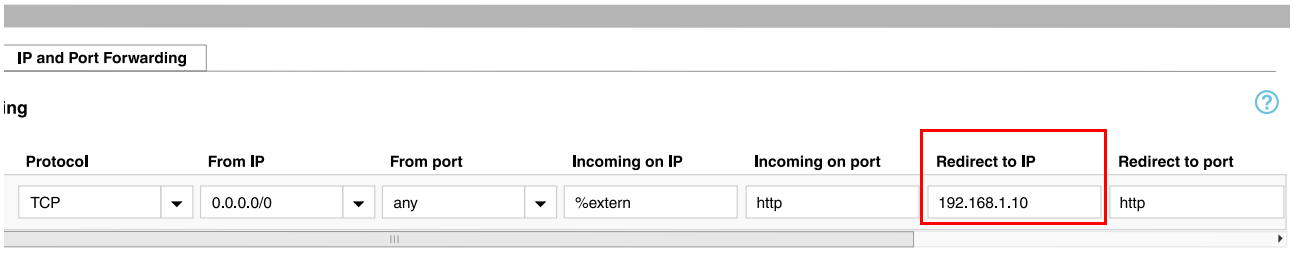
Network packets sent from the company network (WAN) to *mGuard 1* that are intended for its external IP address are rewritten so that they are forwarded to the IP address of the PLC in the production network (LAN). Along with the IP address, the port to which the packet is addressed can also be rewritten with a chosen port.

IP and port forwarding can only be used for the network protocols TCP, UDP and GRE. ICMP is not supported. A *ping* to the PLC is therefore not possible with this option.

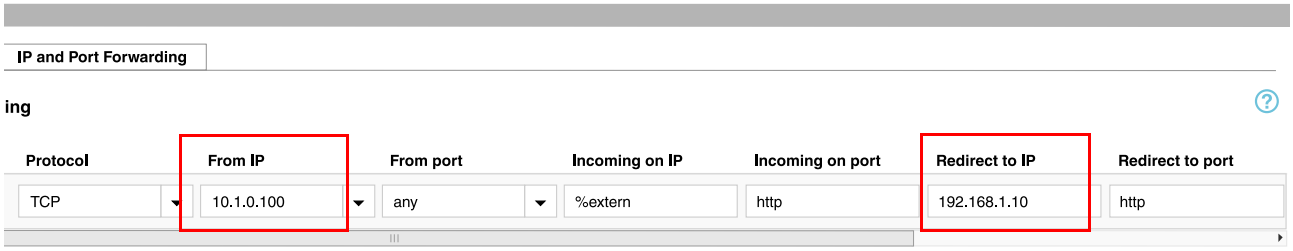
NOTE: If a rule for IP and port forwarding applies to a packet, it is immediately forwarded to the specified destination. Any existing firewall rules that have been configured via **Network Security >> Packet Filter** are not taken into consideration.

Proceed as follows:

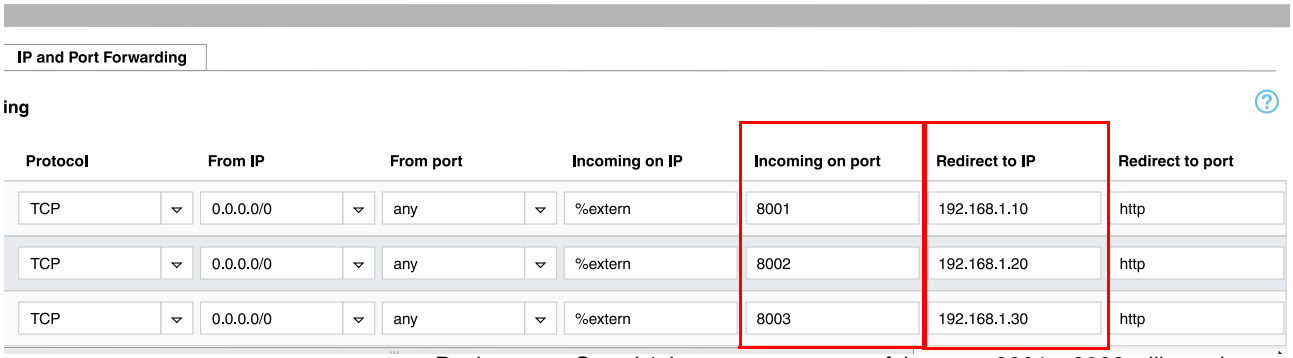
1. Log in to the *mGuard 1* web interface (LAN interface at 192.168.1.254).
2. Go to **Network >> NAT >> IP and Port Forwarding**.
3. Create a rule with the following configuration:



4. **Optional**:
   – With the *From IP* and *From port* fields, the rule can be restricted to certain sender addresses (e.g. a particular computer in the company network: 10.1.0.100) or networks, as well as to certain ports.



– The external IP address of the mGuard can also be specified in the field *Incoming on IP*.
  If the variable **%extern** is used when several static IP addresses are used for the WAN interface, this entry only applies to the first IP address on the list.
  The variable **%extern** is to be used if the mGuard IP address can be changed dynamically so that a particular external IP address cannot be specified.
– In our example, only requests to port 80 (*http*) are forwarded to the destination address and the destination port.
– In order to be able to reach several clients in the destination network using IP and port forwarding, the following configuration can be used:



Packets at *mGuard 1* that are sent to one of the ports 8001 – 8003 will now be forwarded to port 80 (*http*) of the corresponding IP addresses (e.g. 192.168.1.10).

Result

All or (optional) only certain clients in the company network can reach the PLC in the production network via the following IP address:

– Web browser: http://10.1.0.1 (= mGuard device)
– *Ping*: Not possible.

Advantages

– Easy to configure for a small number of destinations.

Disadvantages

– Only port-based protocols (UPD/TCP) can be forwarded (*ping* not possible).
– The destination client (PLC) is accessed via the external IP address of the mGuard device and not via its real IP address.

– If several clients (machine controllers) in the production network are to be reached via the same port, a type of mapping table must be maintained in order to know which port is to be used to access a particular client (e.g. http://10.1.0.1:8001 for 192.168.1.10 or http://10.1.0.1:8002 for 192.168.1.20). This can easily lead to confusion.

For further information, also refer to mGuard firmware user manual.

4.4.3 Option 3: 1:1 NAT

With 1:1 NAT, a **real network** (e.g. the internal production network) is mapped to a **virtual network** via the mGuard. (In our example, the virtual network is part of the external company network.)

The mGuard thus assigns IP addresses of the real network to specific IP addresses of the virtual network. If packets are sent to these virtual IP addresses, mGuard forwards these to the real IP addresses.

Depending on the application, the real and virtual networks can be LAN, WAN or DMZ networks.

Depending on the subnet mask specified in the 1:1 NAT configuration, the subnets of the **real network** can also be mapped in the **virtual network**.

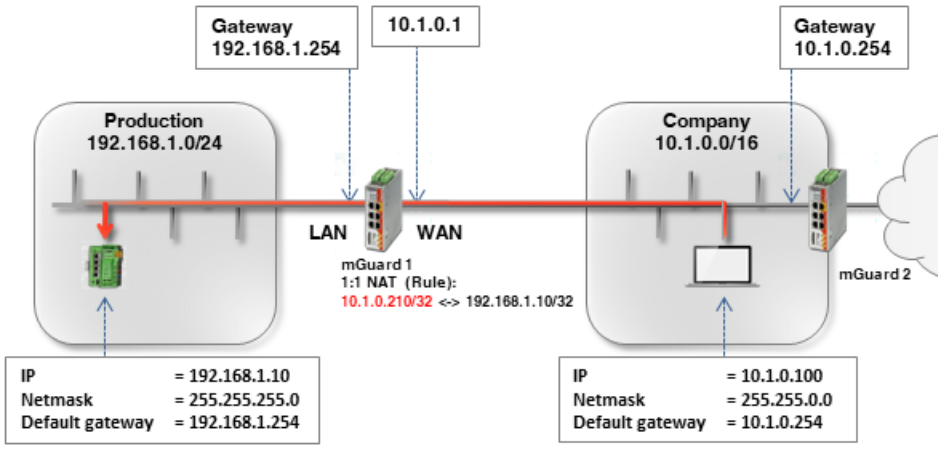
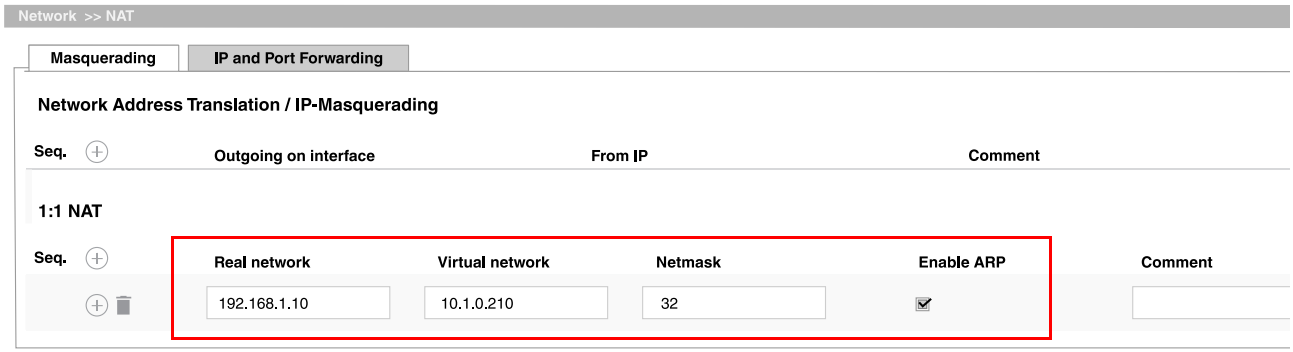


Table 4-1 Examples of rules for 1:1 NAT with different netmasks and the resulting assignments

Real network	Virtual network	Netmask	Assigned IP addresses
192.168.1.10	10.1.0.210	**32**	192.168.1.10 <-> 10.1.0.210

To make the PLC accessible to all clients in the company network, proceed as follows:

1. Log in to the *mGuard 1* web interface (LAN interface at 192.168.1.254).
2. Go to **Network >> NAT >> Masquerading**.
3. In the section 1:1 NAT, create a rule with the following configuration:



4. Packets that are sent to the IP address 10.1.0.210 in the company network are now forwarded to the IP address 192.168.1.10.

NOTE: The IP addresses specified in *Virtual network* must be free. They may not be assigned to other devices or used in any way, because otherwise an IP-address conflict would occur in the *Virtual network*. This even applies when no device exists in the *Real network* for one or more IP addresses from the specified *Virtual network*.

The PLC can now be accessed from the company network via the following IP address:

– Web browser: http://10.1.0.210
– Ping: 10.1.0.210

Advantages

– No changes in the production network are necessary.
– Each client in the production network can be accessed via a *virtual* IP address of the company network.
– The PLC can be accessed via protocols and ports in accordance with the rules specified for the incoming firewall.
– The integration of further network segments (e.g. different production units) into the company network is also possible using an mGuard device in each of the segments to be integrated. Some or all of these networks can use the same internal network settings (e.g. 192.168.1.0.0/24).

  Broadly speaking: if, for example, the (virtual) external network has a subnet mask of 16 and the systems in this network only use IP addresses in the range 10.1.0.1 – 10.1.0.254, the networks 10.1.1.0/24, 10.1.2.0/24, 10.1.3.0/24 can be used to map the (real) internal networks to IP addresses of the (virtual) external network.

Disadvantages

A sufficient number of unused virtual network IP addresses is necessary to be able to perform the mapping.

5 Accessing external networks (IP masquerading | 1:1 NAT)

Document ID: 108408_en_01
Document designation: AH EN MGUARD NETWORK SEGMENT 2

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes the use of the mGuard device as a router that connects two networks (internal and external network). The external network is to be reached from the internal network.

The following procedures are described:

– Option 1: NAT masking (IP masquerading)
– Option 2: 1:1 NAT

5.1 Introduction

In the "Router" network mode (*Router mode*), an mGuard device can be used to connect two networks. The firewall and VPN security functions are also available (depending on license).

With certain models, a demilitarized zone (DMZ) can be connected via the additional DMZ interface as an option.

5.1.1 Example

The production network (= *internal network*) and the company network (= *external network*) are connected via an mGuard router.

A server in the company network is to be accessed from the production network.

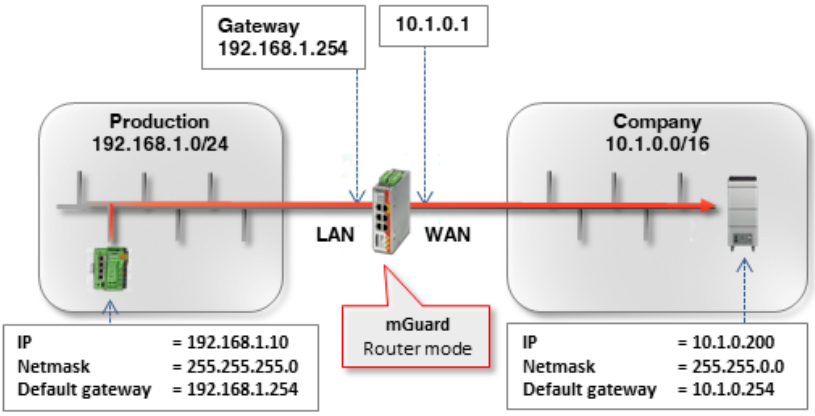


Figure 5-1 Client and mGuard router network settings

The two networks can be connected in various ways:

– Option 1: **Masking / IP masquerading**
– Option 2: **1:1 NAT**

5.1.2 Procedure

1. Configure the WAN and LAN interface of the router (*mGuard 1*)
2. Configure firewall rules
3. Configure network settings in accordance with option 1 or 2

5.2 mGuard router network settings

To enable network traffic between the two networks, the external interface (= WAN port) and the internal interface (= LAN port) of the *mGuard 1* router must be configured in all options and assigned at least one IP address.

Ensure that the clients in the production and company network are configured in accordance with their network.

The internal IP address of *mGuard 1* must be configured as the default gateway (192.168.1.254) for clients in the production network (PLCs).

The internal IP address of *mGuard 2* must be configured as the default gateway (10.1.0.254) for clients in the company network.

To install *mGuard 1* as the router between the company network (WAN) 10.1.0.0/16 and production network (LAN) 192.168.1.0/24, proceed as follows:

1. Log in to the *mGuard 1* web interface (192.168.1.254).
2. Go to **Network >> Interfaces**.
3. *General* tab: select the **network mode** *Router* and the **router mode** *Static*.
4. *Internal* tab: select 192.168.1.254 as the internal IP address.
5. *External* tab: select 10.1.0.1 as the external IP address.

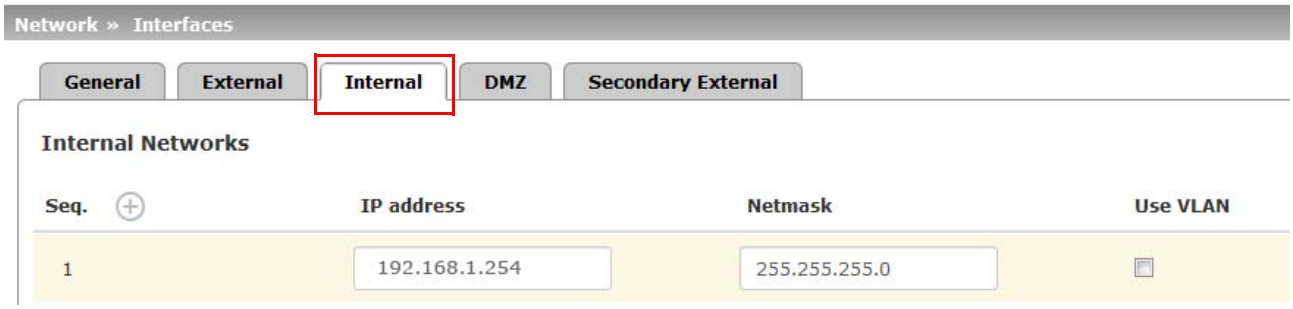


Figure 5-2 Internal interface

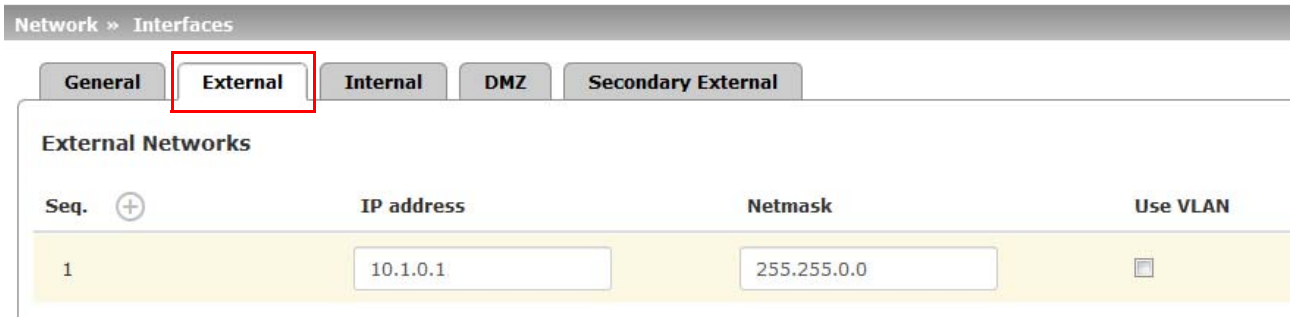


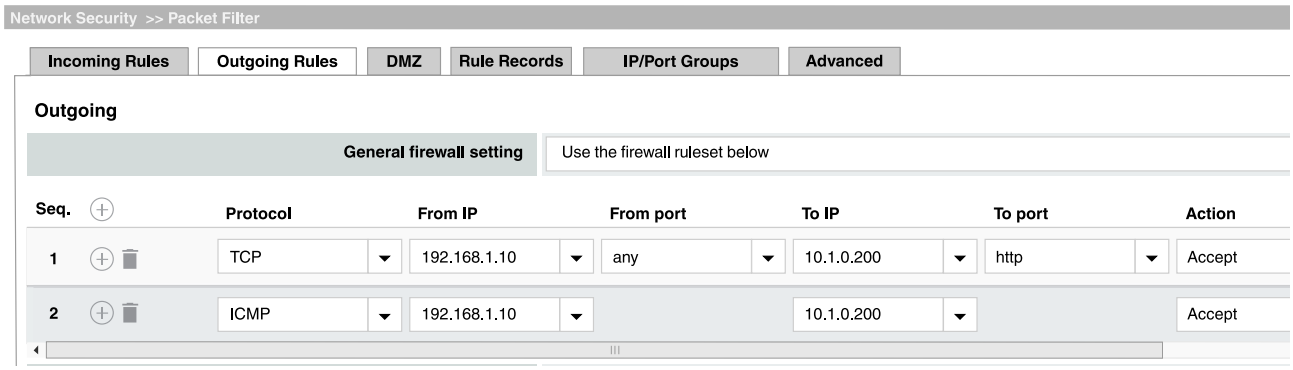
Figure 5-3 External interface

5.3 Configure firewall rules

mGuard 1 is to be configured so as only to allow a particular client from the production network (192.168.1.10) to access the web server (10.1.0.200) in the company network. Apart from that, it should also be possible to "*ping*" the web server (ICMP request).

Proceed as follows:

1. Log in to the *mGuard 1* web interface (192.168.1.254).
2. Go to **Network Security >> Packet Filter >> Outgoing Rules**.
3. Select "*Use the firewall ruleset below*"under **General firewall setting.**
4. Create two firewall rules as follows:



Result

The firewall rules allow outgoing TCP packets to the HTTP port and outgoing ICMP packets. All other packets are rejected by the firewall. The fields **From IP** and **To IP** specify which IP address (server) can be accessed from which IP address (client).

5.4 Network settings in accordance with option 1 and 2

5.4.1 Option 1: masking / IP masquerading

The mGuard device masks the IP addresses of senders from the production network (= *internal network*) with its own external IP address.

This means that the mGuard replaces the IP address of the sender (192.168.1.10) in the data packets with its external IP address (10.1.0.1).

When the packets arrive at the destination server (10.1.0.200), the IP address of the sender (mGuard: 10.1.0.1) is in the same network, and the server sends the response back to the mGuard directly. The mGuard reverses the NAT changes and forwards the response to the original sender (192.168.1.10).

To make the server in the company network accessible to the client in the production network, proceed as follows:

1. Log in to the mGuard web interface (LAN interface at 192.168.1.254).
2. Go to **Network >> NAT >> Masquerading**.
3. In the section *Network Address Translation / IP-Masquerading*, specify a rule with the following configuration:

4. **Optional**: You can also specify all IPs (0.0.0.0/0) in the *From IP* field if you want to enable IP masquerading for all clients in the production network. The access limitation must then be regulated via the firewall settings.

Result

The mGuard router replaces the IP address of packets sent from the client (192.168.1.10) in the production network to the IP address of the server in the company network (10.1.0.210) with its own external IP address and forwards them.

The server in the company network can be reached by the client via its real IP address:

– Web browser: http://10.1.0.200
– Ping: 10.1.0.200

Advantages

– No changes in the production network are necessary.
– Each client in the production network can reach all destinations in the company network via their real IP addresses.
– The destinations in the company network can be accessed via protocols and ports in accordance with the specified firewall rules (outgoing rules).

5.4.2 Option 2: 1:1 NAT

With 1:1 NAT, a **real network** (e.g. the external company network) is mapped to a **virtual network** via the mGuard. (In our example, the *Virtual network* is a part of the internal production network.)

The mGuard thus assigns IP addresses of the real network to specific IP addresses of the virtual network. If packets are sent to these virtual IP addresses, mGuard forwards these to the real IP addresses.

Depending on the application, the real and virtual networks can be LAN, WAN or DMZ networks.

Depending on the subnet mask specified in the 1:1 NAT configuration, the subnets of the **real network** can also be mapped in the **virtual network**.

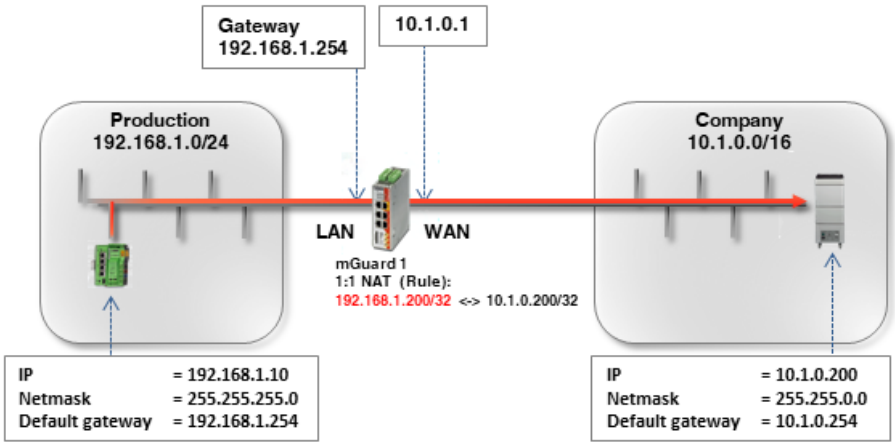
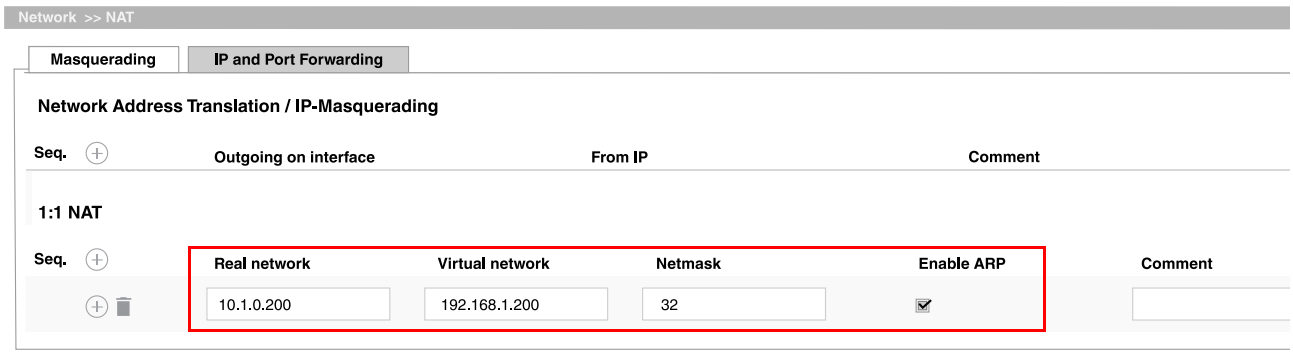


Table 5-1 Examples of rules for 1:1 NAT with different netmasks and the resulting assignments

Real network	Virtual network	Netmask	Assigned IP addresses
10.1.0.200	192.168.1.200	**32**	10.1.0.100 <-> 192.168.1.200

To make the server in the company network accessible to the client in the production network, proceed as follows:

1. Log in to the mGuard web interface (LAN interface at 192.168.1.254).
2. Go to **Network >> NAT >> Masquerading**.
3. In the section *1:1 NAT*, create a rule with the following configuration:



4. Packets that are sent to the IP address 192.168.1.200 in the production network are now forwarded to the IP address 10.1.0.200.

NOTE: The IP addresses specified in *Virtual network* must be free. They may not be assigned to other devices or used in any way, because otherwise an IP-address conflict would occur in the *Virtual network*. This even applies when no device exists in the *Real network* for one or more IP addresses from the specified *Virtual network*.

The server in the company network can now be accessed via the following IP address:

– Web browser: http://192.168.1.200
– Ping: 192.168.1.200

Advantages

– No changes are necessary in the company network.
– Each client in the company network can be accessed via a *virtual* address of the production network.
– The destinations in the company network can be accessed via protocols and ports in accordance with the rules specified in the incoming firewall.

Disadvantages

A sufficient number of unused virtual network IP addresses is necessary to be able to perform the mapping.

6 mGuard firewall properties and possible applications

Document ID: 108405_en_00
Document designation: AH EN MGUARD FIREWALL

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes the fundamental functions of the mGuard firewall, as well as possible applications.

6.1 Stateful packet inspection firewall

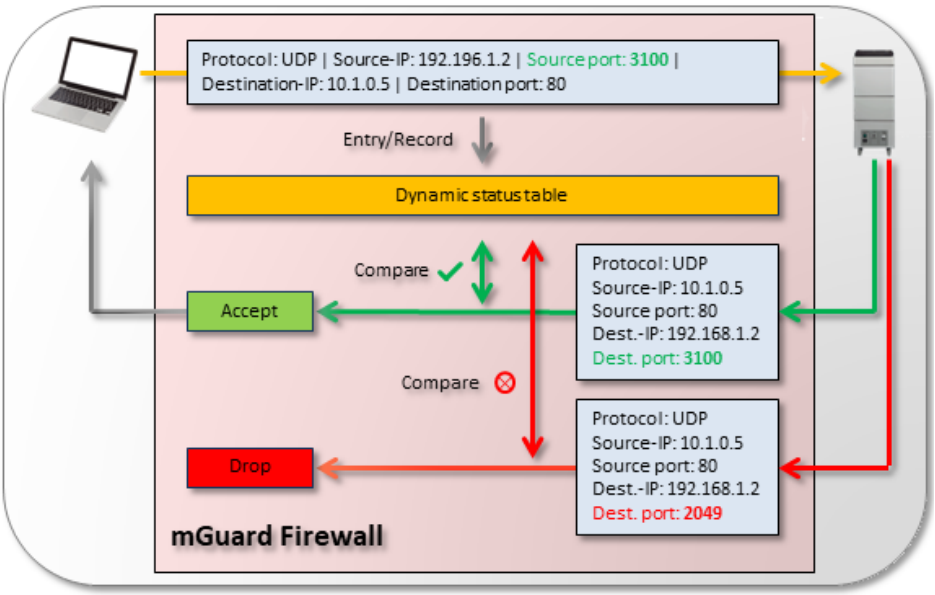


Figure 6-1 If incoming or outgoing packets pass the mGuard firewall (orange arrow), their properties (e.g. protocol, source IP/port, destination IP/port) are stored in a dynamic status table. The properties of the expected response packet are also stored in order that this also passes through the firewall. Response packets are then compared with the values in the status table. If the packets correspond to the dynamically entered values of the status table, they are accepted (green arrow). If they do not match, they are rejected (red arrow).

The mGuard firewall functions as a dynamic packet filter (*Stateful Packet Inspection Firewall*) that analyzes incoming and outgoing network packets in accordance with configured rules.

Dynamic packet filtering allows response packets to pass through the incoming firewall automatically if they can be clearly assigned to the request that previously passed through the outgoing firewall.

It is therefore not necessary in principle to configure incoming rules to accept responses to outgoing requests. An incoming rule could in fact be configured so that all incoming packets are rejected. Incoming responses to requests would still be accepted.

6.2 Static firewall

Static firewall rules are used to control access to the basis of networks (IP addresses, protocols, and ports).

These rules are static and always enabled for the selected interfaces once they have been created. This means that certain devices/networks can communicate with one another.

(**Example**: see Section 4.3, “Configuring firewall rules”)

6.3 Dynamically enabled firewall (via firewall rule records)

Firewall rules that are summarized in firewall rule records can be enabled and disabled dynamically. Enabling and disabling can be carried out

– via web interface,
– via text message (SMS),
– via switch/button,
– by establishing a VPN connection.

As with static firewall rules, access to the basis of networks (IP addresses, protocols, and ports) is controlled. The rules, however, are only enabled when necessary.

(**Example "firewall rule record"**: see Section 8, “Using firewall rule records”)

6.4 User firewall

The user firewall enables user-specific firewall rules that only apply for defined firewall users or user groups to be defined. User firewall rules have priority over firewall rules configured elsewhere (e.g. *Incoming/Outgoing Rules*) and override these where applicable.

Access to the destination is not allowed on the basis of statically configured firewall rules, but dynamically after the firewall user logs on using the user firewall rules assigned to the firewall user.

A user firewall rule comes into effect when a firewall user assigned to the rule logs on via the web interface of the mGuard device. Authentication is performed via the internal database or a RADIUS server.

(**Example**: see Section 9, “Using the user firewall to enable access to an external network”)

7 Frequently occurring errors when creating firewall rules

Document ID: 108403_en_00
Document designation: AH EN MGUARD FIREWALL MISCONFIG

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes common errors when creating firewall rules (e.g. incorrect sequence, incorrect source port).

7.1 Introduction

The mGuard firewall functions as a dynamic packet filter that analyzes incoming and outgoing network packets in accordance with configured rules (see also Section 6, “mGuard firewall properties and possible applications”).

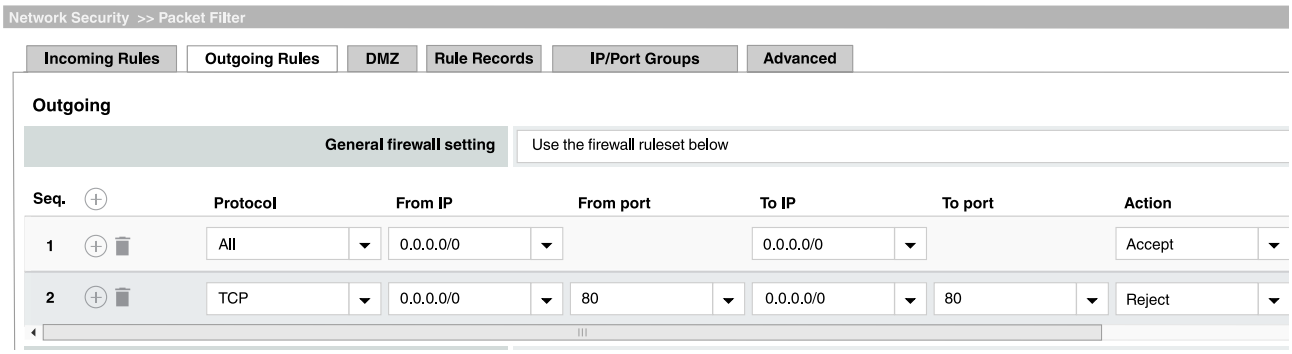
Common errors: when creating firewall rules in a table, their sequence is decisive. The firewall rules created in the table are checked consecutively from top to bottom. If a rule applies, the specified action (*accept, drop*, or *reject*) is performed and the subsequent rules are subsequently **disregarded**.

7.1.1 Example

Access to HTTP web servers from the internal network are to be prevented with the aid of configured firewall rules (mGuard menu: **Network Security >> Packet Filter >> Outgoing Rules**).

Specified ports (*From port* and *To port*) are only taken into consideration if the protocol is set to TCP or UDP.

7.2 Incorrect configuration



Error 1: incorrect sequence

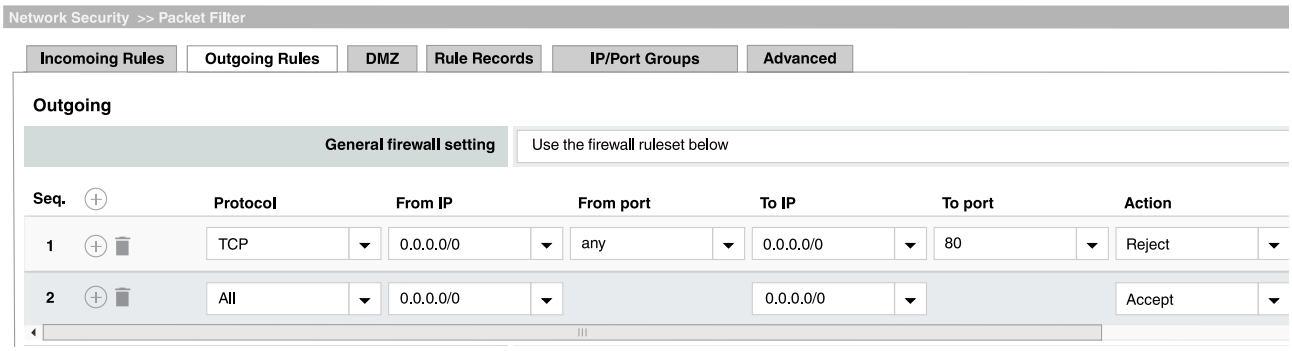
Because the first rule in line 1 already applies for all packets, the following rules are disregarded. An outgoing TCP connection to port 80 will therefore not be rejected.

Error 2: incorrect source port

HTTP requests from web browsers use a varying source port greater than or equal to 1024. The request is sent to port 80. The rule specified in line 2 will not apply because of the entered source port (*From port* = 80), i.e. less than 1024.

7.3 Correct configuration

In the correct configuration, the sequence of the firewall rules must be changed such that the rule that rejects access to a web server is checked first.



"*any*" can be specified, for example, as the source port (*From port*) in order to check requests from a standard web browser. Specifying the destination port (*To port* = *80*) rejects access to a web server.

If the first rule **applies**, the second rule is disregarded. If the first rule **does not apply**, the second rule allows the outgoing data traffic to pass.

8 Using firewall rule records

Document ID: 108402_en_00
Document designation: AH EN MGUARD FIREWALL RULESETS 1

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

The use of firewall rule records is described in this document. This simplifies and accelerates the creation of firewall rules.

8.1 Introduction

Individual firewall rules can be summarized in rule records. These rule records can then be selected in firewall rules as actions and therefore put into use.

8.1.1 Example

External access to three particular servers in the internal network via the network services *ftp*, *telnet* and *https* is to be allowed. Access to all other services and network addresses from the external network (WAN) is to be prohibited.

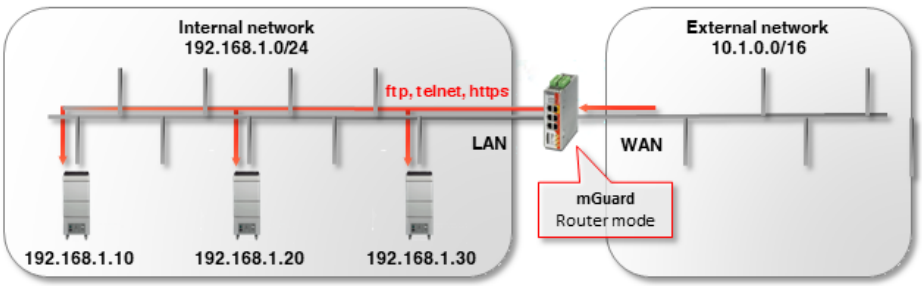


Figure 8-1 Allow access to special services on certain servers

Problem

Without rule records, nine firewall rules must be created in a firewall table: three server IP addresses for each of the three services.

Solution

With the help of rule records, certain sub-rules, i.e. the server IP addresses or the network services, can be summarized in rule records. These can then be selected as actions in firewall tables.

In this example, three incoming rules in the firewall table are sufficient to only allow access to the three servers and the three network services. Here, **either** a "*Server*" rule record or a "*Service*" rule record must be created (see "Example 1 ("Server" rule record)" and "Example 2 ("Service" rule record)").

Please note: if a connection associated with a firewall rule record has been established and this connection is continuously generating data traffic, deactivation of the firewall rule record may not interrupt this connection as would normally be expected (see mGuard firewall user manual).

8.1.2 Procedure

To allow access to defined servers and network services, the following work steps are necessary:

1. Create firewall rule record.
2. Create firewall rule records in firewall tables and refer to the rule record.

8.2 Example 1 ("Server" rule record)

To create the rule record, proceed as follows:

1. Log in to the web interface of the mGuard device.
2. Go to **Network Security >> Packet Filter >> Rule Records**.
3. Create a new rule record with the name *Server* and click on the icon *Edit Row*.
4. Configure the rule record in accordance with Figure 8-2.

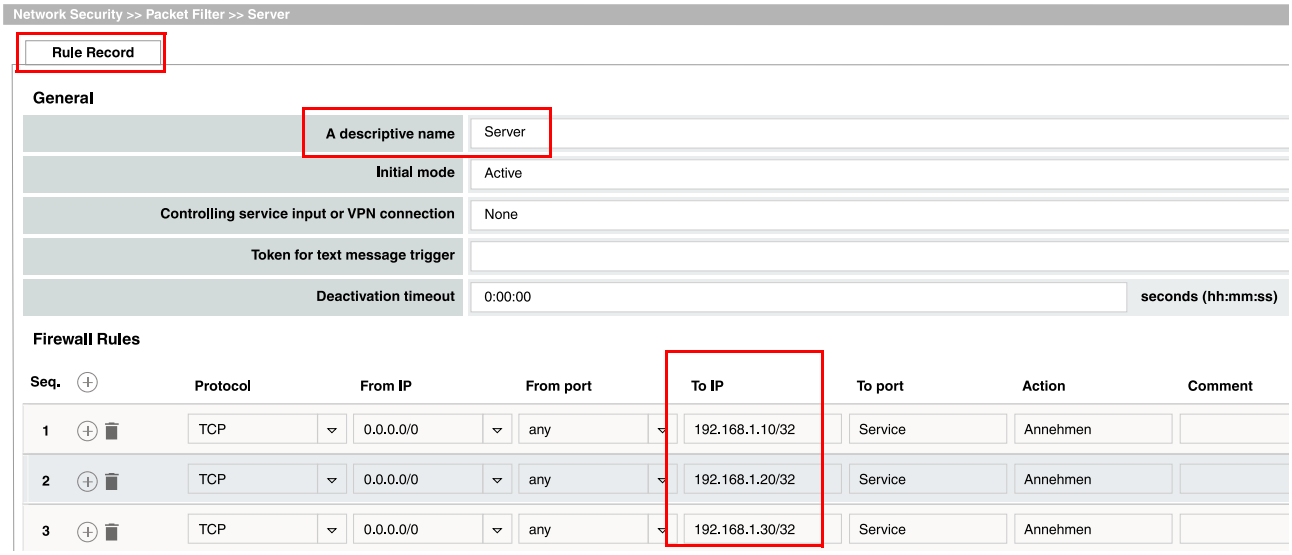


Figure 8-2 The permitted destination IP addresses (destination server) are summarized in the *Server* **rule record**.

To use the rule record in a firewall rule, proceed as follows:

1. Log in to the web interface of the mGuard device.
2. Go to **Network Security >> Packet Filter >> Incoming Rules**.
3. Select **Use the firewall ruleset below**.
4. Create three firewall rules in accordance with Figure 8-3.

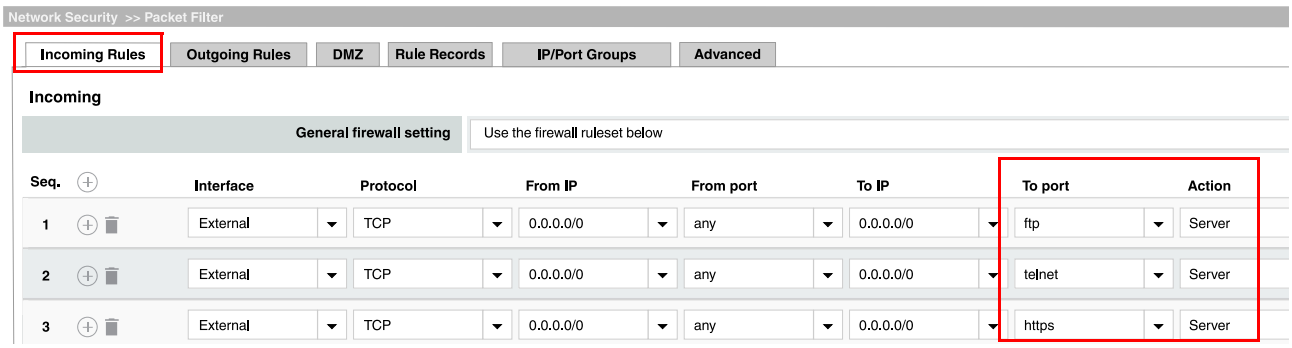


Figure 8-3 The **firewall table** refers to the *Server* rule record as an action when accessing the specified network services.

The firewall rules define access to specific network services (*To port*) and refer to the *Server* rule record. Access to the destinations are defined in this rule record.

8.3 Example 2 ("Service" rule record)

Instead of the server IP addresses, you can also summarize the network services in a rule record and use these in the firewall rules. The settings are as follows (see Figure 8-4 and Figure 8-5).

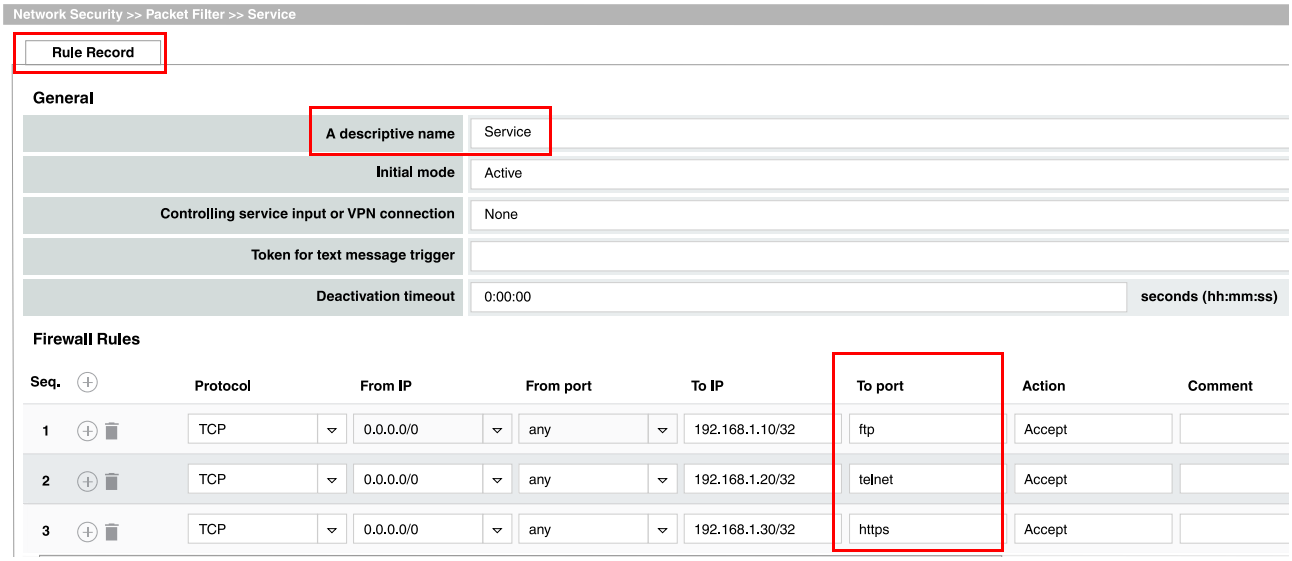


Figure 8-4 The permitted network services are summarized in the *Service* **rule record**.

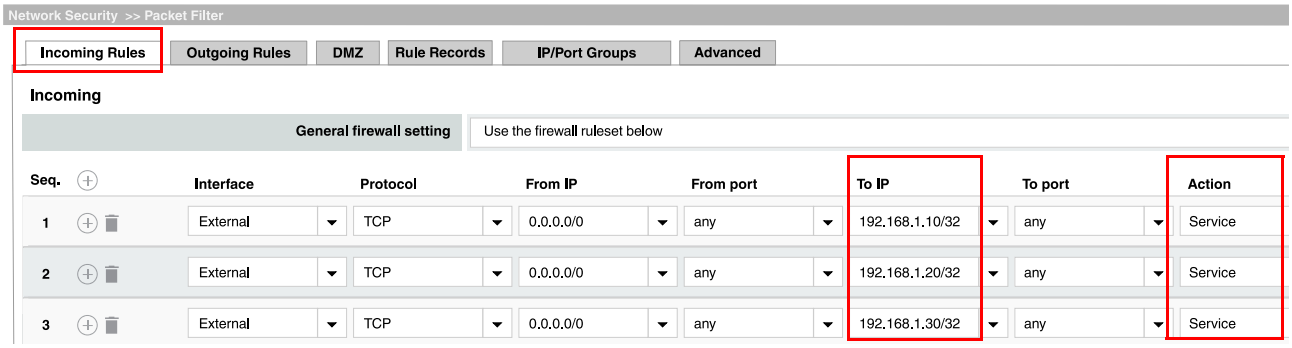


Figure 8-5 The **firewall table** refers to the *Service* **rule record** as an action when accessing the destination IP addresses (destination server).

The firewall rules define access to specific destination IP addresses (*To IP*) and refer to the *Service* rule record. Access to the permitted network services are defined in this rule record.

9 Using the user firewall to enable access to an external network

Document ID: 108401_en_00
Document designation: AH EN MGUARD USERFIREWALL 1

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes how to allow a firewall user access from the internal network to an external network assisted by user firewall rules.

A user firewall is not available on devices of the RS2000 series and the mGuard Blade controller.

9.1 Introduction

The user firewall enables user-specific firewall rules that only apply for defined firewall users or user groups to be defined.

User firewall rules have priority over firewall rules configured elsewhere (e.g. *Incoming/Outgoing Rules*) and override these where applicable.

Access to the destination is not allowed on the basis of statically configured firewall rules, but dynamically after the firewall user logs on using the user firewall rules assigned to the firewall user.

9.1.1 Example

In this example, access from the production network (internal) to the company network (external) is enabled by NAT (IP masquerading) (see also “Option 1: masking / IP masquerading” on page 31).

At the same time, however, **all access instances** from the production network to the company network are prohibited via a general firewall rule (outgoing rule).

Assisted by the user firewall, the firewall users *pwerner* and *hpotter* now have individual access to web servers and can therefore access the web server in the company network.

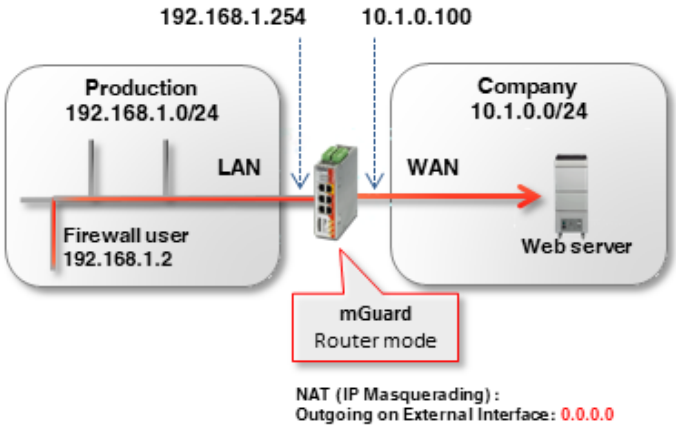


Figure 9-1 Firewall users with access rights to HTTP(S) web servers

9.1.2 Procedure

To allow the firewall users *pwerner* and *hpotter* access to a web server via port 80 (http) and 443 (https), the following work steps are necessary:

1. Create firewall users
2. Create user firewall template with firewall rules
3. Enable user firewall
4. Log in as firewall user

9.2 Creating firewall users

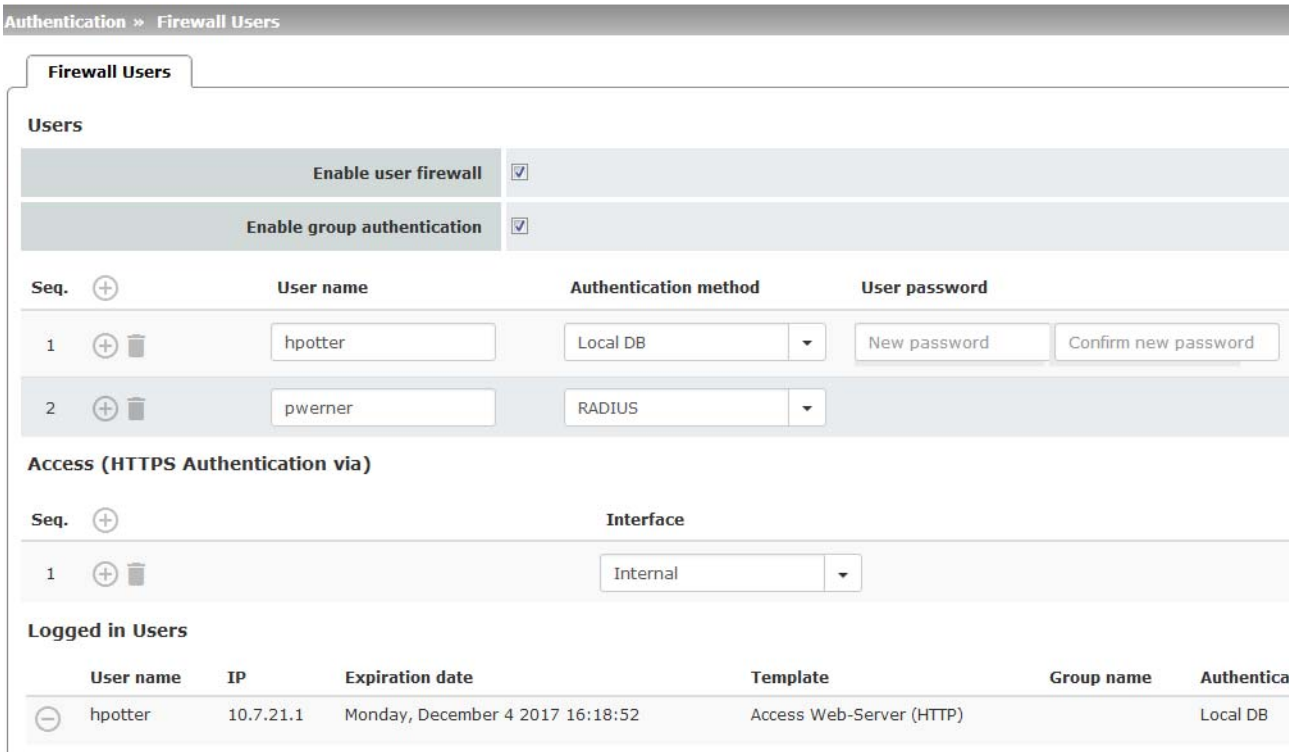


Figure 9-2 Create firewall users

Firewall users are created under **Authentication >> Firewall Users**. Whether the users are authenticated via a RADIUS server or via a user password configured locally on the mGuard device is also specified here.

The general configuration for use of a RADIUS server via the mGuard device is set in the menu **Authentication >> RADIUS**.

A firewall user can be assigned to one or more user firewall templates (see “"Template Users" tab” on page 47).

To create a firewall user, proceed as follows (see also mGuard firmware user manual):

1. Log in to the web interface of the mGuard device.
2. Go to **Authentication >> Firewall Users**.
3. Create the desired firewall users.
4. Specify the authentication procedure for each user (password or RADIUS server).
5. Specify via which interfaces the firewall users may use to log in at the mGuard device.

9.3 Creating a user firewall template

In a user firewall template, firewall rules are created and existing firewall users are assigned.

If a user firewall template or a firewall rule in a template is added, changed, deleted or disabled, all logged-in firewall users are affected immediately.

Existing connections are interrupted. An exception to this is when firewall rules are changed, and the function "Abort existing connections upon firewall reconfiguration" is disabled in the menu **Network security >> Packet Filter >> Advanced**. In this case, a network connection that exists due to a previously permitted rule is not interrupted.

If a firewall rule record (template) is **disabled** and then **enabled**, the affected logged-in user must first log out then log back in for the firewall rules in the template to be enabled again.

To create a user firewall template, proceed as follows:

1. Log in to the web interface of the mGuard device.
2. Go to **Network security >> User firewall**.
3. Create a new template and click on the *Edit Row* icon .

9.3.1 "General" tab

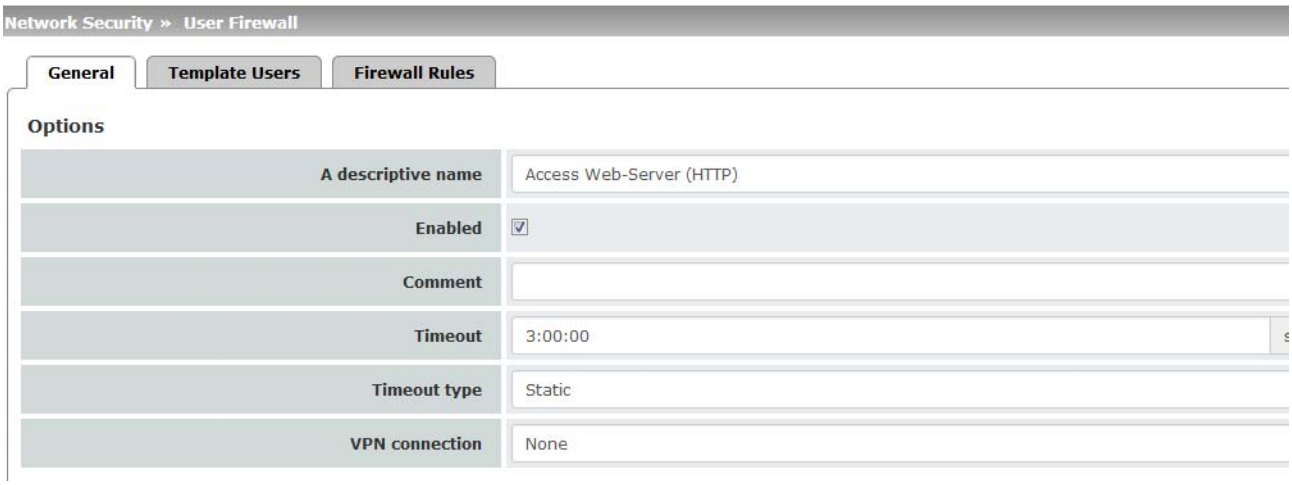


Figure 9-3 Creating a user firewall template: *General* tab

Proceed as follows (see also mGuard firmware user manual):

- Assign a descriptive name to the user firewall template.
- Specify how long a user firewall is to be valid once a firewall user has logged in (note Timeout type).
- If the rules of the user firewall template are only to be valid for a particular VPN connection, specify this.

9.3.2 "Template Users" tab

Figure 9-4 Creating a user firewall template: *Template Users* tab

Proceed as follows (see also mGuard firmware user manual):

- Specify the name of the firewall users for which the rules of this user firewall template are to apply.

The specified users must have been defined and created under **Authentication >> Firewall Users >> Users** (see “Creating firewall users” on page 45).

NOTE: A check is not made as to whether the specified user names actually exist. Ensure that the names are entered correctly.

9.3.3 "Firewall Rules" tab

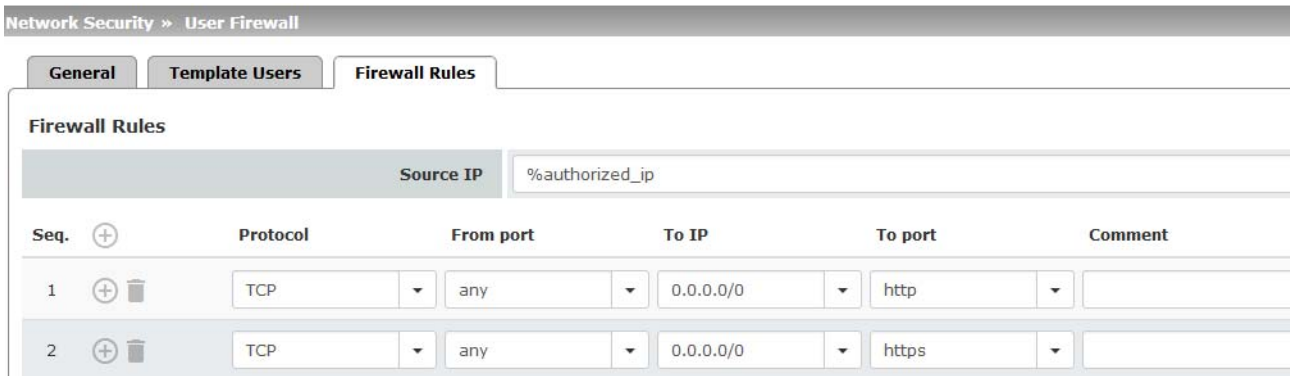


Figure 9-5 Creating a user firewall template: *Firewall Rules* tab

The mGuard device automatically recognizes the interface used to log in and applies the user firewall templates accordingly as *Incoming Rules* (logon from the external network) or *Outgoing Rules* (logon from the internal network).

If the template is configured with dynamic timeout, approved UDPs and other network packets (excluding ICMP) reset the dynamic timeout to the initial value at this point.

To configure the firewall rules of the template, proceed as follows (see also mGuard firmware user manual):

- Specify a source IP address from where the connection is to be permitted.

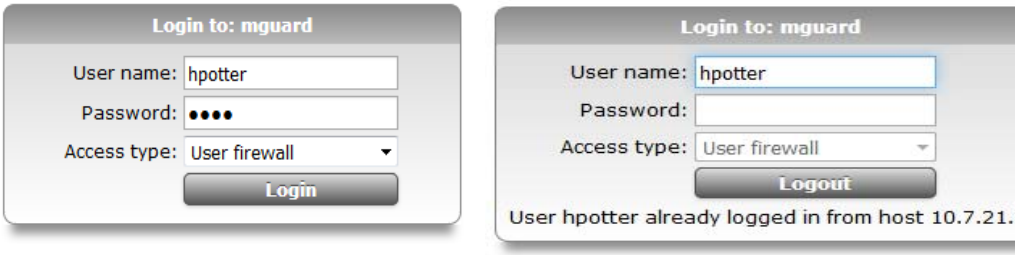
If %authorized_ip is specified, the firewall rules are used on data packets that are sent from the same source IP address via which the user logged in. Data packets from other IP addresses are dropped .

If an IP address is specified, the firewall rules are applied to data packets that are sent from this source IP address. Data packets from other IP addresses are dropped. This option should be used, for example, if an administrator logs into the device to enable the user firewall for a technician who is working on another computer.

- Create firewall rules to enable the assigned firewall users access in accordance with the rules created.

  In this example, access to any web server via the network services *http* and *https*.

9.4 Logging in as a firewall user



A firewall user must log into the mGuard device web interface via HTTPS using the web browser in order to enable the firewall rules. This can be from either the internal network or the external network (or via VPN, DMZ, and dial-up). To log in via the external network on the device, HTTPS remote access must be enabled on the mGuard device (Menu **Management >> Web Settings >> Access**).

The mGuard device automatically recognizes the interface used to log in and applies the user firewall templates accordingly as *Incoming Rules* (logon from the external network) or *Outgoing Rules* (logon from the internal network).

To log in as a firewall user, proceed as follows:

1. Open the login window in the mGuard device web interface.
2. Select the access type "User firewall".
3. Enter the user name and password for the firewall user.
4. If the login is successful, this is displayed in the login window.

Result

All connections to an HTTP(S) web server via the selected protocol are enabled when a firewall user logs in until the timeout period elapses.

10 IPsec VPN – Basic functions

Document ID: 108413_en_00
Document designation: AH EN MGUARD IPSEC VPN OVERVIEW

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes general application possibilities and the basic function of IPsec VPN connections.

10.1 Introduction

Data packets are normally sent via the Internet unprotected and therefore do not meet the basic security requirements:

– Encryption (data confidentiality)
– Authentication (proof of the identity of the sender)
– Integrity (assurance that the data packets have not been modified).

A *Virtual Private Network* (VPN) is a communications channel that uses encryption and authentication to protect data transmitted over a public medium (e.g. the Internet).

The most commonly used VPN protocol today is *Internet Protocol Security* (IPsec). Most VPN devices and clients are IPsec-compliant. IPsec is scalable and can be used in both small applications and in large VPM gateways with over 1,000 VPN connections.

IPsec supports transport connections that connect two individual hosts, as well as tunnel connections that connect two networks.

10.1.1 Setup of ISAKMP SA and IPsec SA

A VPN connection is established in two phases: Phase I (ISAKMP SA key exchange) and Phase II (IPsec-SA data exchange). SA stands for *Security Association*.

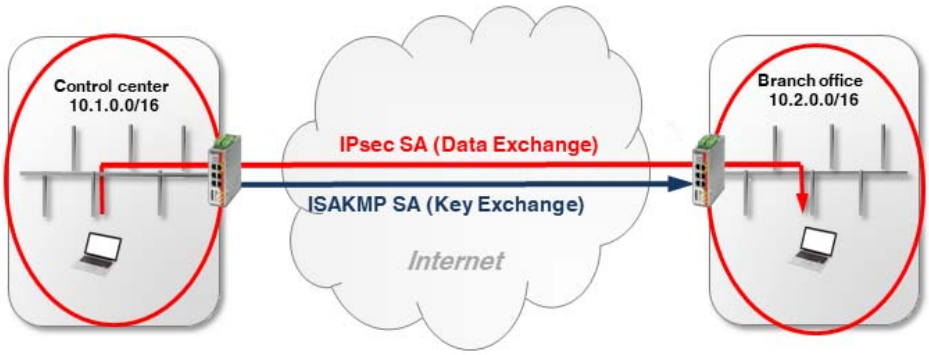


Figure 10-1 Setting up an IPsec connection (ISAKMP SA and IPsec SA)

Phase I (ISAKMP SA):

ISAKMP SA is a secure connection (*Security Association*) between two VPN peers, via which the secure exchange of the *keys* necessary for VPN encryption is agreed as the first step.

For this, both VPN peers negotiate the encryption and hash algorithm for phase I and authenticate one other mutually using *pre-shared keys* (PSK) or X.509 certificates (see Section 10.3).

Both peers then agree upon a *key* for encrypting the phase II data exchange.

Phase II (IPsec SA):

The IPsec SA (*Security Association*) is a secure connection via which the internal networks of the VPN peers are connected and data exchanged.

For this, both peers negotiate the encryption and hash algorithm for phase II and exchange information regarding the networks to be connected.

10.1.2 Configuration of IPsec VPN connections

The IPsec VPN connections between an mGuard device and a VPN peer are configured in the menu **IPsec VPN >> Connections** (see also mGuard firmware user manual). A VPN connection is normally *initiated* by a device, while the peer device *waits* for the connection request from the initiator.

The VPN connection is configured in the following tabs:

- **"General" tab**
- **"Authentication" tab**
- **"Firewall" tab**
- **"IKE Options" tab**

10.2 "General" tab

The settings in the "General" tab depend upon the network environment in which the VPN connection is established (e.g. network mode *Stealth, Router, PPPoE*) and on the VPN properties that are to be used (e.g. *1:1 NAT for local networks* or *hub and spoke*). See also Section 11 and 12.

10.2.1 Example

An encrypted IPsec VPN tunnel is to be established between **company network 1** (192.168.1.0/24) and **company network 2** (192.168.2.0/24). The VPN connection is initiated by *mGuard 1*. Both devices are operated in the *Router* network mode.

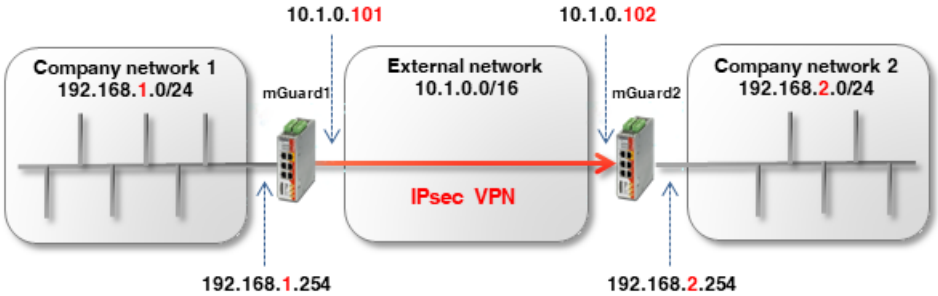


Figure 10-2 Connecting two networks via IPsec VPN

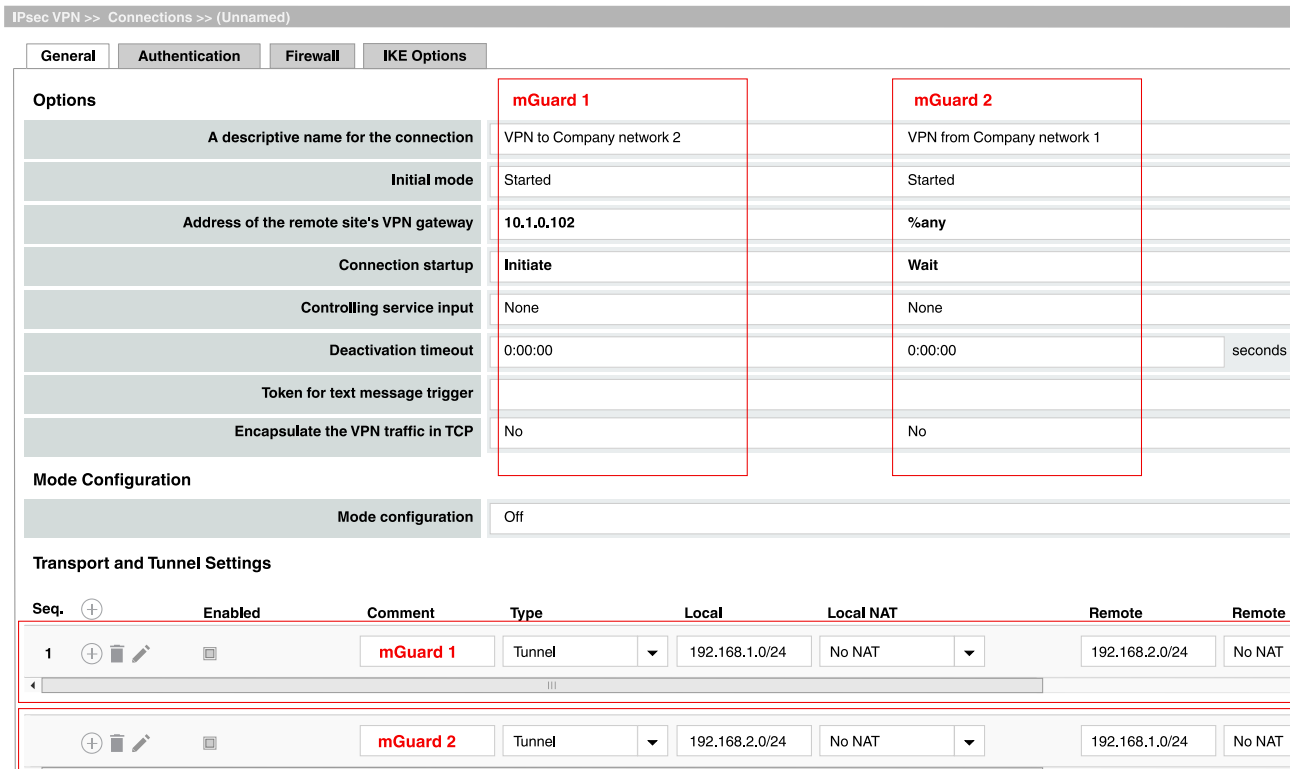


Figure 10-3 Menu: IPsec VPN >> Connections >> (Edit) >> General

10.3 "Authentication" tab

Mutual authentication of the two VPN peers can be performed in two ways:

– X.509 certificates
– Pre-shared key (PSK)

Pre-shared key (PSK)

This procedure is mainly supported by older IPsec implementations. In this case, both sides of the VPN connection authenticate each other using the same password (PSK). The PSK is made up of a string consisting of alphanumeric characters. The PSK procedure can be used in the secure *Main Mode* or in the unsecure *Aggressive Mode* (see also mGuard firmware user manual, section "IPsec VPN >> Connections >> Authentication").

X.509 certificates

This procedure is supported by most IPsec implementations. Here, each VPN device has a (secret) private key and a public key in the form of an X.509 certificate, which contains further information on its owner and a *Certification Authority (CA*) (see also mGuard firmware user manual, section "IPsec VPN >> Connections >> Authentication").

Which procedure is to be used?

Certificates are generally more secure and can be applied in all network scenarios. Creating a certificate, however, requires a certain amount of effort and precise planning.

Using a PSK in *Main Mode* with a sufficiently complex password is also relatively secure. In some network environments, however, PSKs cannot be used or can only be used with difficulty:

– PSKs in the secure *Main Mode* cannot be used if the VPN connection is established via one or more gateways with *Network Address Translation* (NAT) enabled. This means that PSKs can only be used if both devices are connected to the same external network or are connected directly to the Internet. Otherwise, the unsecure *Aggressive mode* would be necessary.
– When using PSKs, the external (or public) IP address of the peer VPN gateway must be entered in each location in the VPN configuration. The generic entry *%any* cannot be used on the responding side. For this, the unsecure *Aggressive mode* would be necessary.

10.3.1 Example: Creating X.509 certificates

A certificate acts as a unique ID and must therefore be unique for each device. The X.509 certificates can either be obtained from a commercial certification authority (e.g. *VeriSign*), or a Microsoft CA server, or can be created with software tools such as *OpenSSL* and *XCA* (see also application note "Creating X.509 certificates with OpenSSL/XCA".

When creating a certificate, the parameters that can be used to clearly determine the ownership of the certificate must first be specified (*Common Name, Organization, Organization Unit,* etc.).

Next, a key pair is generated: a private key and the corresponding public key. The private key *must* be carefully protected, while the public key can be published.

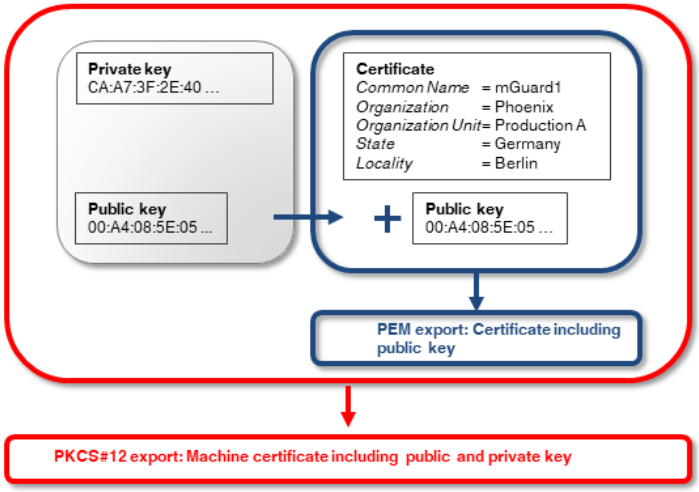


Figure 10-4 PEM and PKCS#12 exports of X.509 certificates with public or public and private keys

10.3.2 Example: Using X.509 certificates

In a VPN connection, the following must be determined

– How the mGuard device is authenticated by the peer and
– How the mGuard device authenticates the remote peer.

If authorization occurs via X.509 certificates, the VPN connection can only be established if the private key on one side “corresponds“ to the public key on the other side (see also Section 11.3, “Importing machine certificates (PKCS)”).

The certificates created must therefore be exported in two different formats, and imported into the respective devices:

1. **PEM format:**

   The certificate in PEM format only contains the public key. It must be imported into every device that attempts to establish a VPN connection with the device to which the certificate (PKCS#12 export = *Machine Certificate*) belongs (see Figure 10-5).

2. **PKCS#12 format:**

   The certificate in PKCS#12 format contains both the public and the associated (corresponding) private key. It will only be imported into a particular device as the unique *Machine Certificate* of this device (see Figure 10-5).

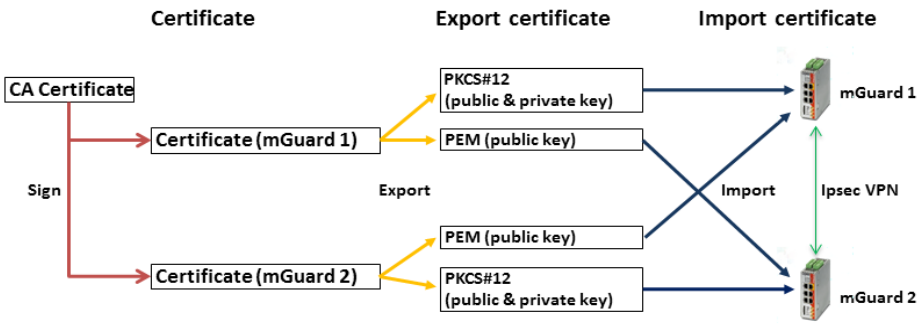


Figure 10-5 Necessary certificates in an IPsec VPN connection

Table 10-1 Example: Certificates in an IPsec VPN connection

Device	Machine certificate (also contains the private key)	Client certificate (only contains the public key)
mGuard 1	*mGuard1.p12*	*mGuard1.pem*
mGuard 2	*mGuard2.p12*	*mGuard2.pem*

mGuard devices also support CA authentication. With this function, the peer is authenticated via the CA certificate used to sign the peer certificate (remote certificate). Authentication via the remote certificate itself is then no longer necessary. This function is mainly used in VPN tunnel groups.

Multiple use of a certificate (as a device-specific ID) on different devices is not recommended and will normally lead to problems.

Uploading X.509 certificates to devices and using in VPN connections

The use of X.509 certificates in mGuard devices is described in Section 11, “VPN Kickstart – Connecting two networks together via IPsec VPN”.

10.4 "Firewall" tab

VPN-specific firewall rules can be specified when configuring the VPN connection. The VPN firewall allows access via the VPN tunnel to be restricted. It can be configured as necessary. In the default configuration, all incoming and outgoing connections are accepted.

(See also mGuard firmware user manual, section "IPsec VPN >> Connections >> Firewall").

10.4.1 Example

An encrypted IPsec VPN tunnel is to be established between **company network 1** (192.168.1.0/24) and **company network 2** (192.168.2.0/24).

Two clients in company network 1 are to have access to two controllers in company network 2. All other clients are to be denied access to company network 2. All connections from company network 2 to company network 1 are not permitted.

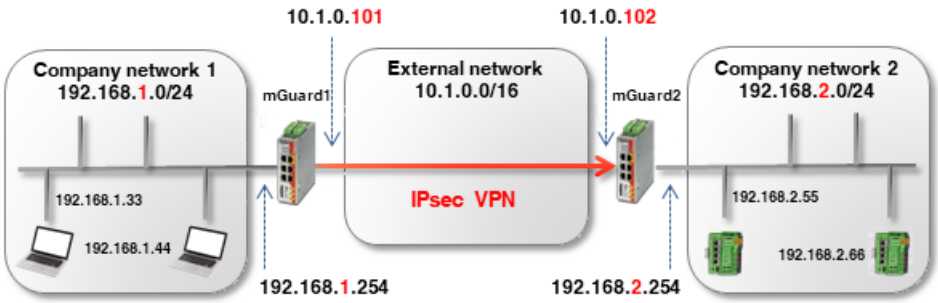


Figure 10-6 VPN connection between two networks with firewall

The firewall settings can, in principle, be configured on *mGuard 1* or *2* or on both devices. In this example, the firewall is configured on *mGuard 1*. The use of firewall rule records is also possible (see also Section 8).

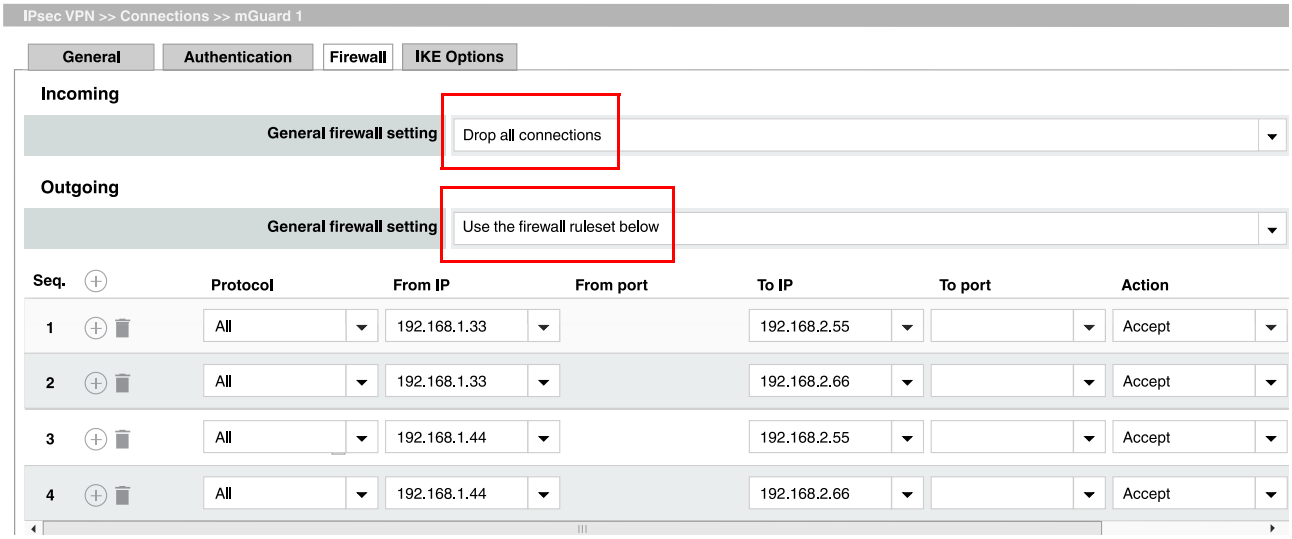


Figure 10-7 *mGuard 1*: IPsec VPN >> Connections >> (Edit) >> Firewall

10.5 "IKE Options" tab

Internet Key Exchange (IKE) is a protocol used for management and exchange of the keys involved within the IPsec protocol.

The IKE options specify

– The encryption and hash algorithms that are to be used for ISAKMP SA and IPsec SA
– The service life of the SAs and
– the parameters for Dead Peer Detection (DPD).

The strongest or most secure encryption method and/or hash algorithms are to be used wherever possible. Otherwise, the standard settings can in principle be used. (See also mGuard firmware user manual, section "IPsec VPN >> Connections >> IKE Options").

For information on secure encryption, see mGuard firmware user manual (Section "Secure encryption").

10.6 mGuard behind a NAT router

If the VPN connection is established via one or more gateways on which *Network Address Translation* (NAT) is enabled

1. X.509 certificates must be used for secure authentication. *Pre-Shared Keys* (PSK) can only be used in the unsecure *Aggressive Mode*.
2. only one of the mGuard devices can *initiate* the VPN connection. The other mGuard device must *wait* for the connection.
3. *%any* must be specified as *Address of the remote site's VPN gateway* on the responding mGuard, even if the NAT router of the peer has a static public IP address.
4. it must be ensured that the VPN connection is established via the UDP ports 500 and 4500.

The network and NAT settings shown in the following example are to be observed.

10.6.1 VPN initiator behind NAT router

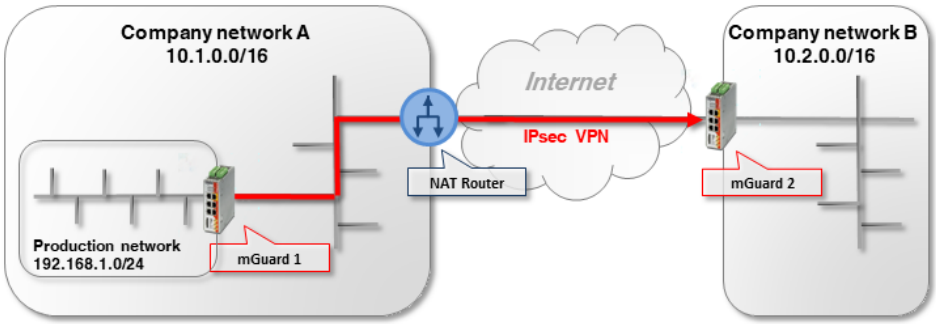


Figure 10-8 VPN initiator behind NAT router

mGuard 1 (*initiator*) initiates the VPN connection to *mGuard 2* (*responder*).

The NAT router firewall must allow outgoing UDP packets to the ports 500 and 4500. If these ports cannot be opened for any particular reason, *TCP encapsulation* or the *Path Finder* function can be used to establish the VPN connection (see Section 10.7).

10.6.2 VPN responder behind NAT router

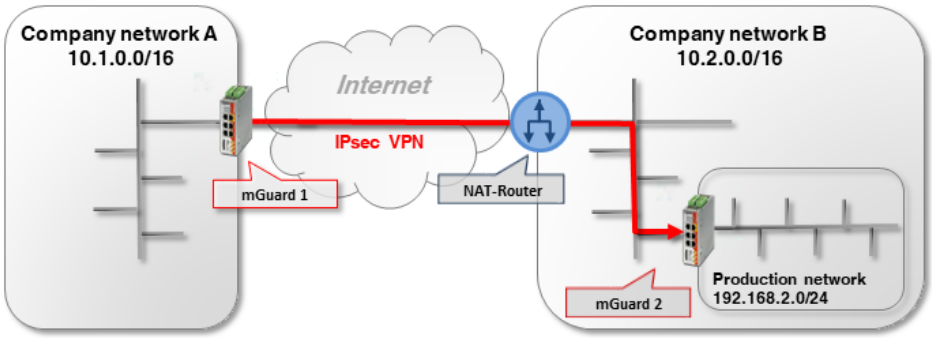


Figure 10-9 VPN responder behind NAT router

mGuard 1 (*initiator*) initiates the VPN connection to *mGuard 2* (*responder*).

Port forwarding to the external IP address (WAN port) of *mGuard 2* must be configured on the NAT router for the UDP ports 500 and 4500. (If it is an mGuard device, this can be set via **Network >> NAT >> IP and port forwarding**).

As port forwarding is necessary on the NAT router for the UDP ports 500 and 4500, no further VPN connections to the NAT router can be established (termination). (This would be possible using TCP encapsulation/Path Finder function.) Likewise, it will not be possible to establish VPN connections to further mGuard devices in the company network B.

If this is to be the case, *mGuard 2* would have to initiate the VPN connection to *mGuard 1*. It would then not be necessary to configure port forwarding on the NAT router.

10.6.3 VPN initiator and responder behind NAT routers

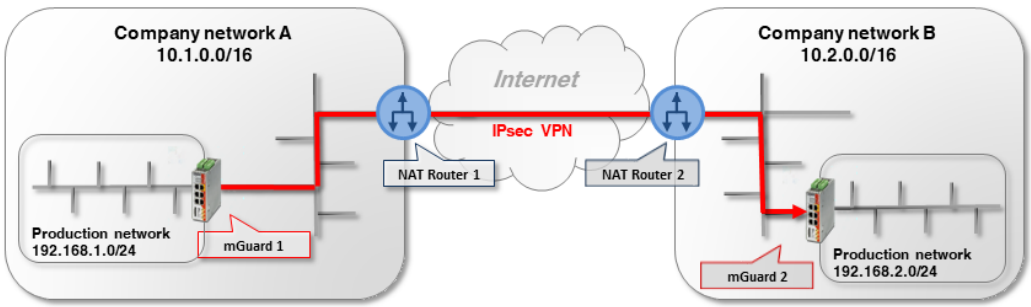


Figure 10-10 VPN initiator and VPN responder behind NAT routers

mGuard 1 (*initiator*) initiates the VPN connection to *mGuard 2* (*responder*).

The NAT router 1 firewall must allow outgoing UDP packets to the ports 500 and 4500.

Port forwarding for the UPD ports 500 and 4500 to the external IP address (WAN port) of *mGuard 2* must be configured on NAT router 2.

10.7 TCP encapsulation

In order to be able to establish an IPsec VPN connection, the UDP ports 500 and 4500 must be open in an outgoing firewall. If these ports are blocked, the VPN connection can be established using *TCP encapsulation* or the *Path Finder* function via a permitted TCP port.

Here, the UDP packets are packed in TCP packets (encapsulated) and sent to a TCP port which allows outgoing TCP packets (e.g. port 80 or 8080) according to the firewall settings of the NAT router.

TCP encapsulation can also be used for establishing the VPN connection if access to the Internet is only possible via a customer proxy server. In this case, the access parameters must be configured in the proxy server (menu **Network >> Proxy Settings**).

10.7.1 Example

A customer would like to access the server of a manufacturing company via a VPN connection. However, the customer firewall blocks the UDP ports 500 and 4500 for outgoing connections.

TCP connections via TCP port 80 are, on the other hand, permitted. The VPN connection is therefore to be established using TCP encapsulation via the TCP port 80. (The configuration of VPN connections is described in detail in Section 11 and 12).

Certificates must be used for secure authentication because the VPN connection is established via a NAT router. If authentication is to be via *pre-shared key*, the unsecure *Aggressive Mode* must be used (see Section 10.3).

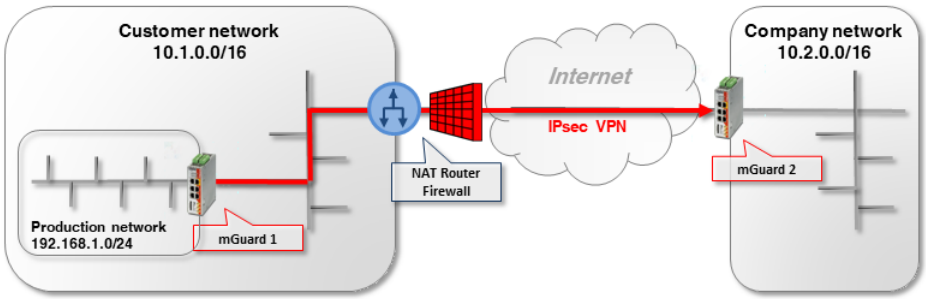
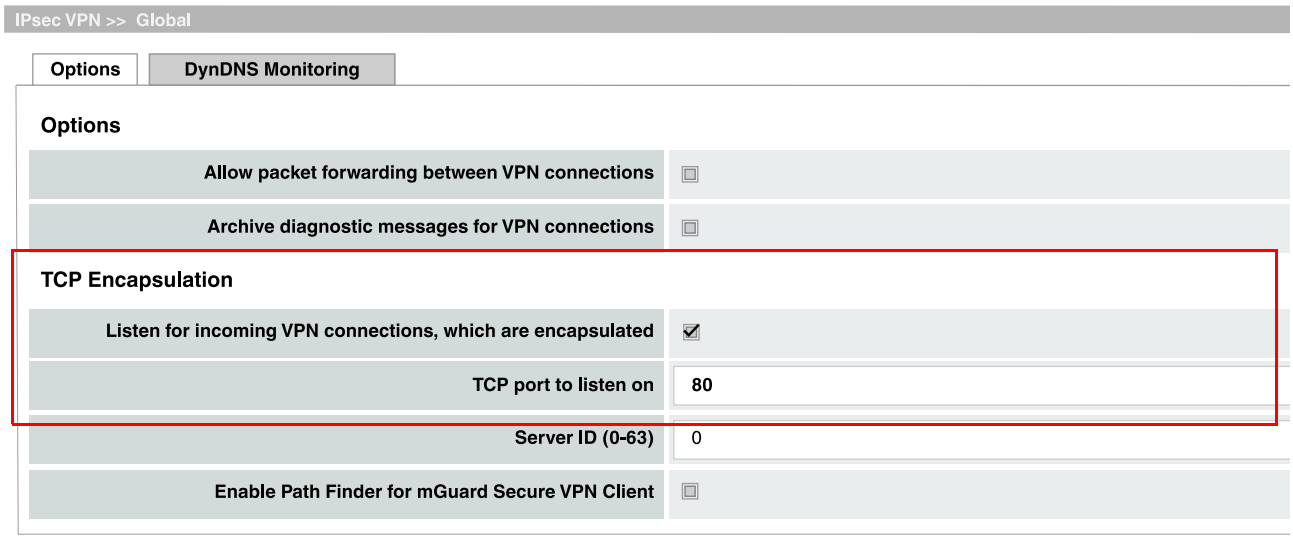


Figure 10-11 VPN initiator behind NAT router and firewall

mGuard 1 (*initiator*) initiates the VPN connection to *mGuard 2* (*responder*). Normally, a VPN connection is established using NAT via the UDP ports 500 and 4500. These, however, are blocked by the customer firewall of the NAT router.

The encrypted ESP packets are also encapsulated in UDP packets by NAT-T. They would also be affected if UDP ports 500 and 4500 were blocked.

10.7.2 mGuard 2 settings (responder)



To set which port the *VPN responder* should be listening on for encapsulated VPN connections, proceed as follows:

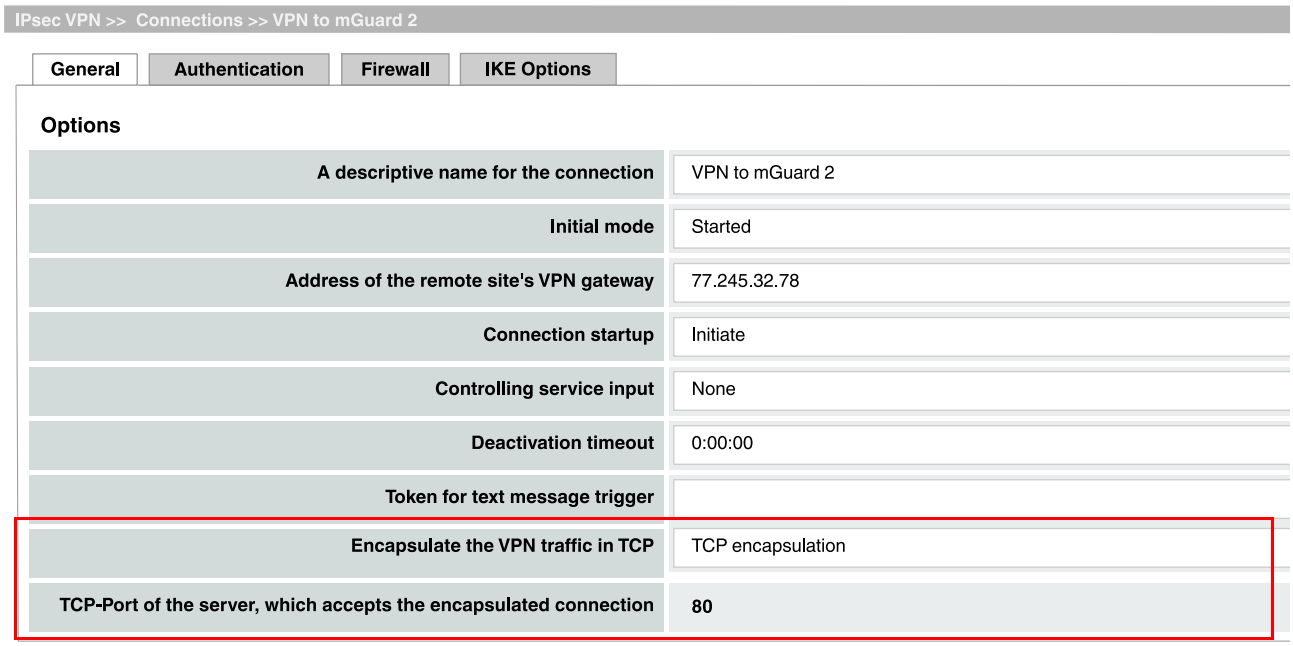
1. Log in to the *mGuard 2* web interface.
2. Go to **IPsec VPN >> Global** (*Options* tab).
3. In the section **TCP Encapsulation**: activate the option **Listen for incoming encapsulated VPN connections**. This will start the IPsec TCP server on the device.
4. In this example, enter port *80* in the field **TCP port to listen on**. This port must also be entered for TCP encapsulation in the *VPN initiator* (*mGuard 1*) (see Section 10.7.3).

Do not select TCP port 443, as it is already used by default to access the device's Web-based management via HTTPS remote access.

If TCP encapsulation also uses port 443, HTTPS remote access to the web interface is no longer possible.

Either specify a different TCP port for remote access (menu **Management >> Web Settings**, *Access* tab), e.g. Port 4443, or select another TCP port for TCP encapsulation.

10.7.3 mGuard 1 settings (initiator)



To inform the *VPN initiator* which port the peer device (*VPN responder*) listens for on encapsulated VPN connections, proceed as follows:

1. Log in to the *mGuard 1* web interface.
2. Go to **IPsec VPN >> Connections**.
3. Click on the ⊕ icon to add a new VPN connection.
4. Specify a unique name for the connection and click on the ✎ icon to edit the connection.
5. Enter either the DynDNS name or the public IP address of the peer (*mGuard 2*) (e.g. *mGuard2.dyndns.org* or 77.245.32.78) as the **Address of the remote site's VPN gateway**.
6. Select *Initiate* in the **Connection startup** field.
7. Select TCP encapsulation in the field **Encapsulate the VPN traffic in TCP**.
8. In this example, enter port *80* in the field **TCP port of the server, which accepts the encapsulated connection**. This port must also be entered for TCP encapsulation under *VPN responder* (*mGuard 2*) (see Section 10.7.2).

10.8 Starting/stopping or analyzing VPN connections using URLs

It is possible to start or stop or query the status of a VPN connection configured on the mGuard using the command line command *curl*:

https://<user>:<password>@<mGuard IP>/nph-vpn.cgi?name=<name>&cmd=[up|down|status]

<user>: the following users can be used: *admin*, *root* and *user*.

<name>: name of the VPN connection, as displayed in the menu **IPsec VPN >> Connections**.

Using the command line command **wget** only works in conjunction with **mGuard firmware versions <8.4.0**. The *curl* command line tool can be used with mGuard firmware version 8.4.0, or higher.

The user password and the name that an action relates to may only contain the following characters:

– Letters: A - Z, a - z
– Numbers: 0 - 9
– Characters: - . _ ~

Other characters, such as a space or question mark, must be encoded accordingly (see also mGuard firmware user manual).

10.8.1 Examples

The mGuard device on which, for example, the "*Athen*" VPN connection is configured can be reached via the IP address 192.168.1.1.

1. Starting the "*Athen*" VPN connection:

wget --no-check-certificate "https://admin:mGuard@192.168.1.1/nph-vpn.cgi?name=Athen&cmd=up"

curl --insecure "https://admin:mGuard@192.168.1.1/nph-vpn.cgi?name=Athen&cmd=up“

2. Stopping the "*Athen*" VPN connection:

wget --no-check-certificate "https://admin:mGuard@192.168.1.1/nph-vpn.cgi?name=Athen&cmd=down"

curl --insecure "https://admin:mGuard@192.168.1.1/nph-vpn.cgi?name=Athen&cmd=down“

3. Requesting the status of the "*Athen*" VPN connection:

wget --no-check-certificate "https://admin:mGuard@192.168.1.1/nph-vpn.cgi?name=Athen&cmd=status"

curl --insecure "https://admin:mGuard@192.168.1.1/nph-vpn.cgi?name=Athen&cmd=status“

The option ***--no-check-certificate*** (*wget*) or **--insecure** (*curl*) ensures that the HTTPS certificate of the mGuard device will not subsequently be checked.

10.9 Starting or stopping a VPN connection via button or switch

Service contacts (I/Os) can be connected to some mGuard devices:

TC MGUARD RS4000/RS2000 3G, TC MGUARD RS4000/RS2000 4G,
FL MGUARD RS4004/RS2005, FL MGUARD RS4000/RS2000, FL MGUARD RS,
FL MGUARD GT/GT

The user manual describes how the service contacts are connected to the devices (see mGuard Hardware Manual – UM EN MGUARD DEVICES).

Input (CMD I1, I2, and I3)

Buttons or on/off switches (e.g. a key switch) can be connected to the inputs. One or more freely selectable VPN connections or firewall rule records can be switched via the corresponding switch. A combination of VPN connections and firewall rule records is also possible.

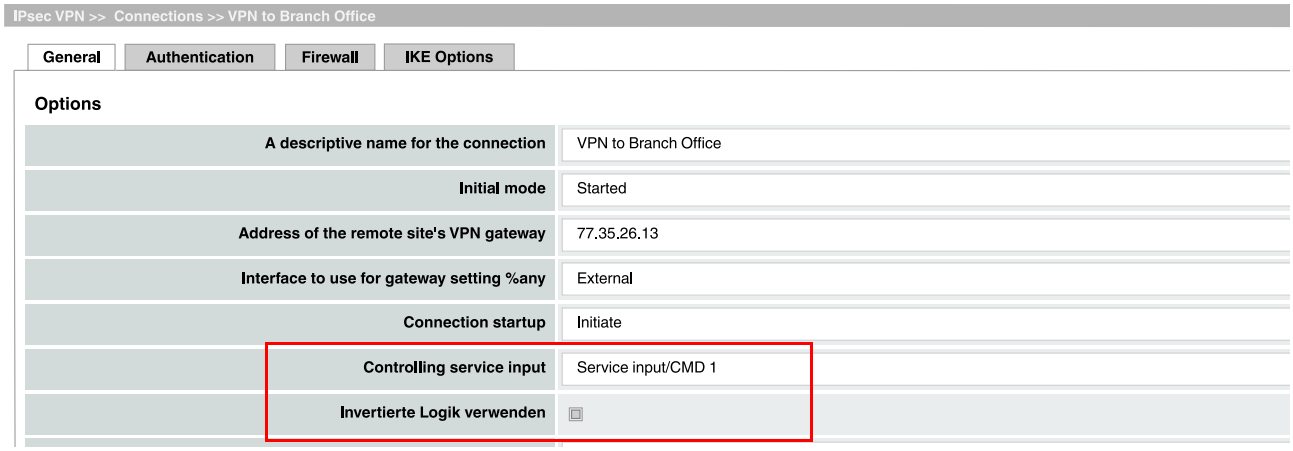


Figure 10-12 A service input is assigned to the VPN connection via which it can be started or stopped via button or on/off switch.

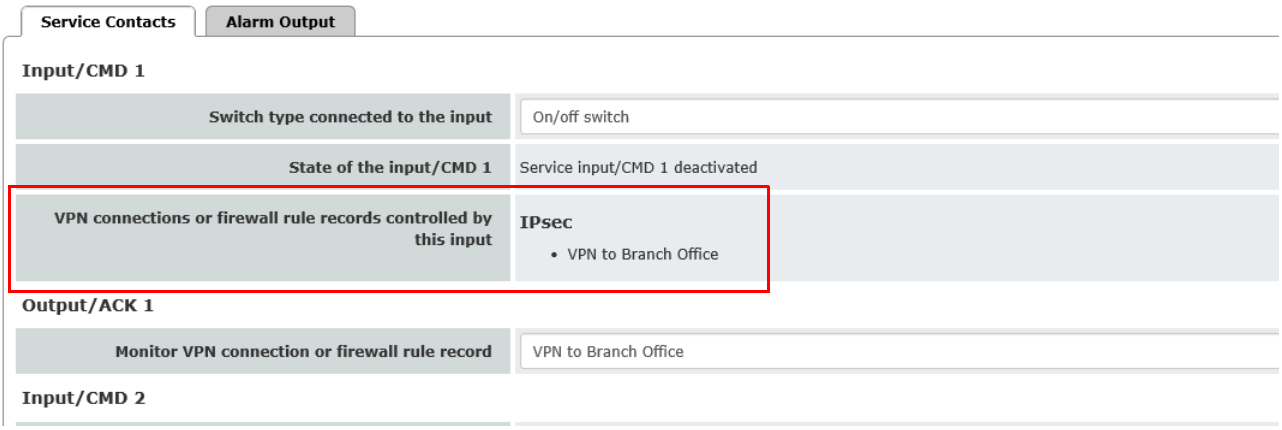


Figure 10-13 The web interface displays which VPN connections and which firewall rule records are controlled via a service input.

Signal contact (signal output) ACK 1/2 (O1, O2)

You can set whether specific VPN connections or firewall rule records are monitored and displayed via the signal output ACK 1, ACK 2, or LEDs.

11 VPN Kickstart – Connecting two networks together via IPsec VPN

Document ID: 108404_en_00
Document designation: AH EN MGUARD VPN KICKSTART

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes the configuration of an IPsec VPN connection between two networks.

11.1 Introduction

Using IPsec VPN, networks can be connected together via an encrypted VPN tunnel.

11.1.1 Example

Using two mGuard devices, an encrypted IPsec VPN tunnel is to be established between **company network 1** (192.168.1.0/24) and **company network 2** (192.168.2.0/24).

If two locations have the same internal network, the VPN 1:1 NAT function has to be used for the local network (see “Using NAT in VPN connections” on page 89).

In this case the VPN connection is initiated by *mGuard 1*. The VPN tunnel is established once the *waiting* mGuard device of the peer (*mGuard 2*) is available. Both mGuard devices are operated in the *Router* network mode.

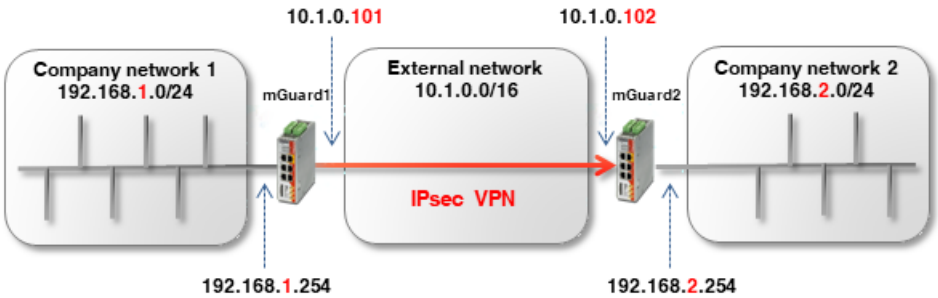


Figure 11-1 Connecting two networks via IPsec VPN

Optional: router mode PPPoE

Establishing a VPN tunnel between two mGuard devices in the *PPPoE* router mode via the Internet is similar in principle (see Figure 11-2). In this case, the Internet is the external network. The devices receive their external IP settings from the Internet Service Provider (ISP). Static name resolution with dynamically assigned IP addresses is then possible with the aid of a DynDNS service.

If the responding (waiting) mGuard device (*mGuard 2*) has a dynamic public IP address, this mGuard must register its external IP address with a DynDNS service (e.g. *mGuard2.dyndns.org*) using a freely selectable name. The initiating mGuard device (*mGuard 1*) must provide a reference to this name in order to establish the VPN connection.

In this case, activate **DynDNS Monitoring** (**IPsec VPN >> Global >> DynDNS Monitoring**) in the VPN connection of the initiating device (mGuard 1). Otherwise, the device will not know when the IP address of the peer has changed and the VPN connection will not be established.

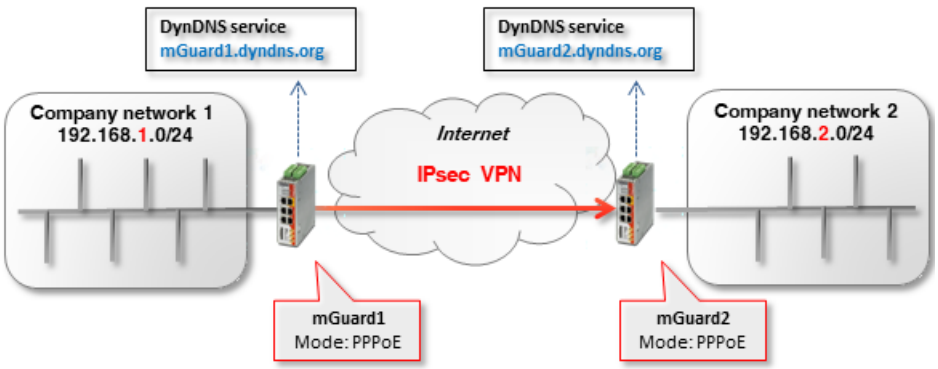


Figure 11-2 Determination of the host names for the mGuard devices using DynDNS. Because the VPN connection is initiated via *mGuard 1*, this does not need a DynDNS address.

11.1.2 Prerequisite

1. Two mGuard devices with the latest firmware (e.g. version 8.6.1 or higher).
2. An existing network connection (IP connection) between the mGuard devices (e.g. via Internet, WAN or LAN).
3. An internal and an external IP address for each mGuard device.
4. UDP ports 500 and 4500 open in the firewall on both sides of the IPsec VPN connection.
5. (Optional) a host name for each mGuard device, e.g. via DynDNS (e.g. *mGuard1.dyndns.org* and *mGuard2.dyndns.org*).

11.1.3 Procedure

1. Generate X.509 certificates and keys
2. Import X.509 certificates and keys
3. Configure IPsec VPN connection tunnel settings
4. Test IPsec VPN connection setup

11.2 Generating machine certificates (X.509 certificates)

Certificates that are necessary for secure authentication of mGuard devices can be obtained from a commercial certification authority. Programs such as *XCA*, *OpenSSL* and *Microsoft Certification Authority (CA) Server* can be used for creating self-signed certificates.

Self-signed certificates are not accredited by an official certification authority, and can therefore only be used under certain circumstances.

The application notes "Creating X.509 certificates with OpenSSL/XCA" describe how to generate self-signed certificates using OpenSSL and XCA.

The following certificates are necessary for the authentication of an IPsec VPN connection between two mGuard devices. (In our example, the unique names *mGuard 1* and *mGuard 2* are used as the *common names* in the certificates.)

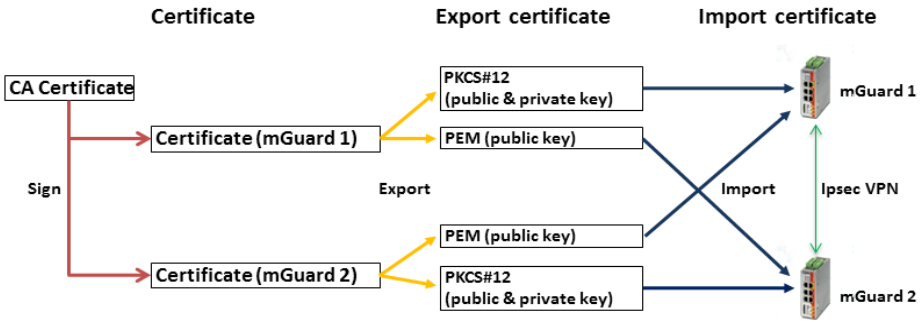
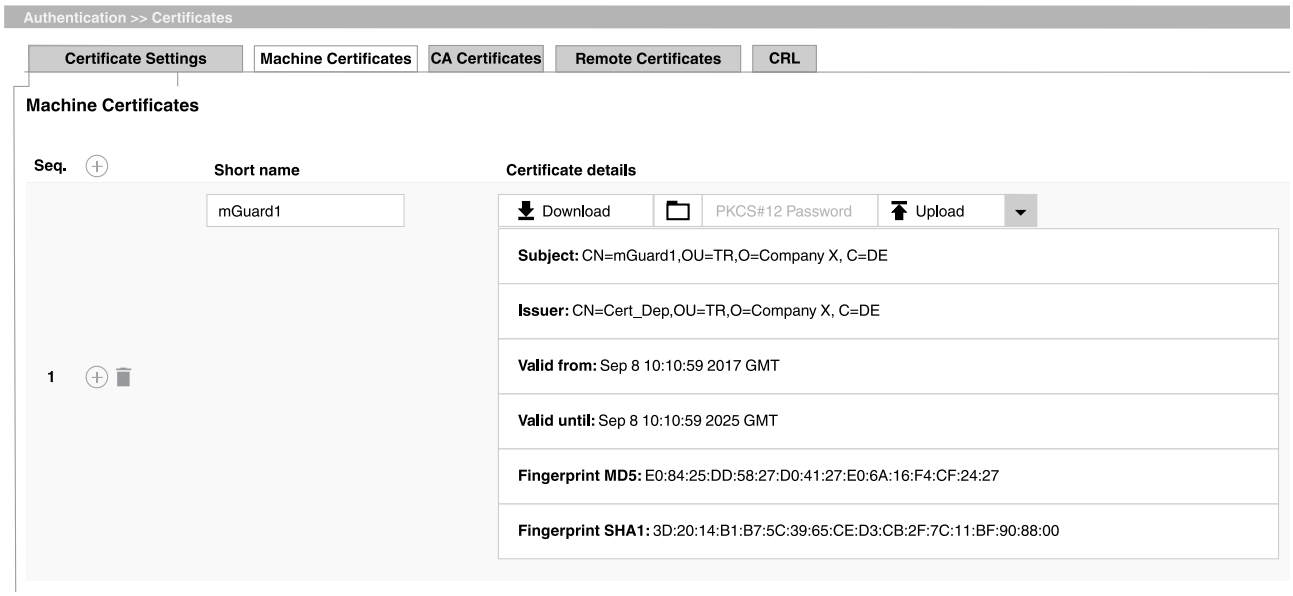


Figure 11-3 Participating certificates in an IPsec VPN connection

Table 11-1 Necessary certificates

Device	Machine certificate (also contains the private key)	Client certificate (only contains the public key)
mGuard 1	*mGuard1.p12*	*mGuard1.pem*
mGuard 2	*mGuard2.p12*	*mGuard2.pem*

11.3 Importing machine certificates (PKCS)



To import X.509 machine certificates (incl. private key) into your mGuard devices, proceed as follows:

1. Log into the *mGuard 1* web interface (e.g. https://192.168.1.254).
2. Go to **Authentication >> Certificates** (*Machine certificates* tab).
3. Click on the ⊕ icon to add a new machine certificate.
4. Click on the 🗀 icon to select the certificate file on the installation computer.
5. Select the previously created file *mGuard1.p12.*
6. Enter the PCKS#12 password issued when generating the certificate.
7. Give the certificate a unique short name. If you leave this field empty, the *common name (CN)* of the certificate is used automatically.
8. Click on the **Upload** button to import the certificate.
9. Click on the "*Save*" icon 💾 to complete the import.

Repeat the procedure for the device *mGuard2*, and import the machine certificate with the file name *mGuard2.p12.*

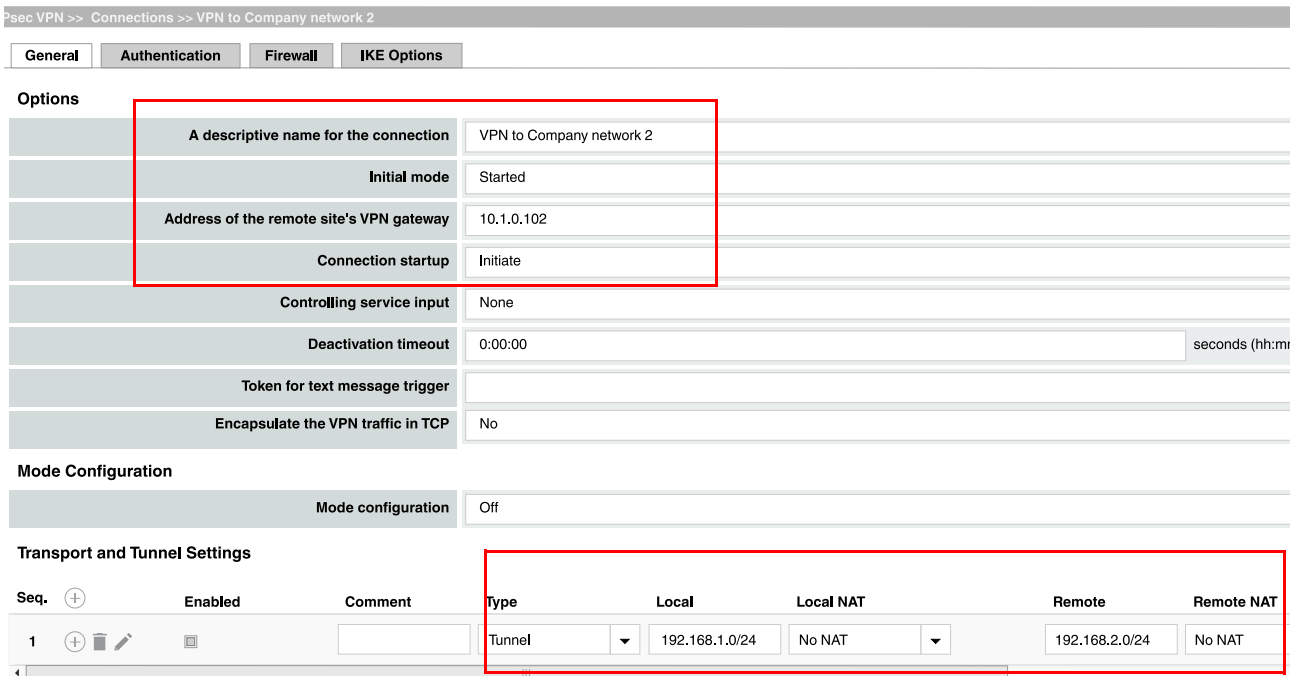
11.4 Creating the mGuard1 VPN connection

11.4.1 Prerequisite

To configure the IPsec VPN connection, the following basic settings must be made:

1. Log into the *mGuard 1* web interface (e.g. https://192.168.1.254).
2. Go to **IPsec VPN >> Global** (*Options* tab).
3. In the section **IP Fragmentation**: activate the option *IKE fragmentation* and as a precaution set a value of 1414 or less *for IPsec* under *MTU* for compatibility reasons.

11.4.2 Configuring the VPN connection



To configure the VPN connection, proceed as follows:

1. Go to **IPsec VPN >> Connections**.
2. Click on the ⊕ icon to add a new VPN connection.
3. Specify a unique name for the connection and click on the ✎ icon to edit the connection.

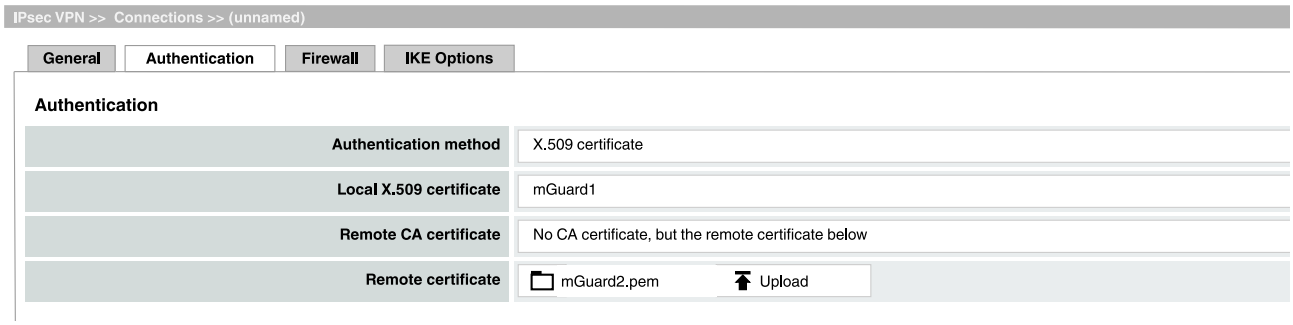
Section "Options"

1. Enter either the DynDNS name or the external IP address of the peer (*mGuard 2*) (*mGuard2.dyndns.org* or 10.1.0.102) as the **Address of the remote site's VPN gateway**.
2. Select *Initiate* in the **Connection startup** field.

Section "Transport and Tunnel Settings"

1. Enter the address of the network that is to be accessible via the *mGuard1* internal interface in the field **Local** (192.168.**1**.0/24).
2. Enter the address of the network that is to be accessible via the *mGuard2* internal interface in the field **Remote** (192.168.**2**.0/24).
3. Click on the "*Save*" icon to complete the procedure.

11.4.3 Configuring authentication of the VPN connection



To configure mutual authentication of the two peers when setting up the VPN connection, proceed as follows:

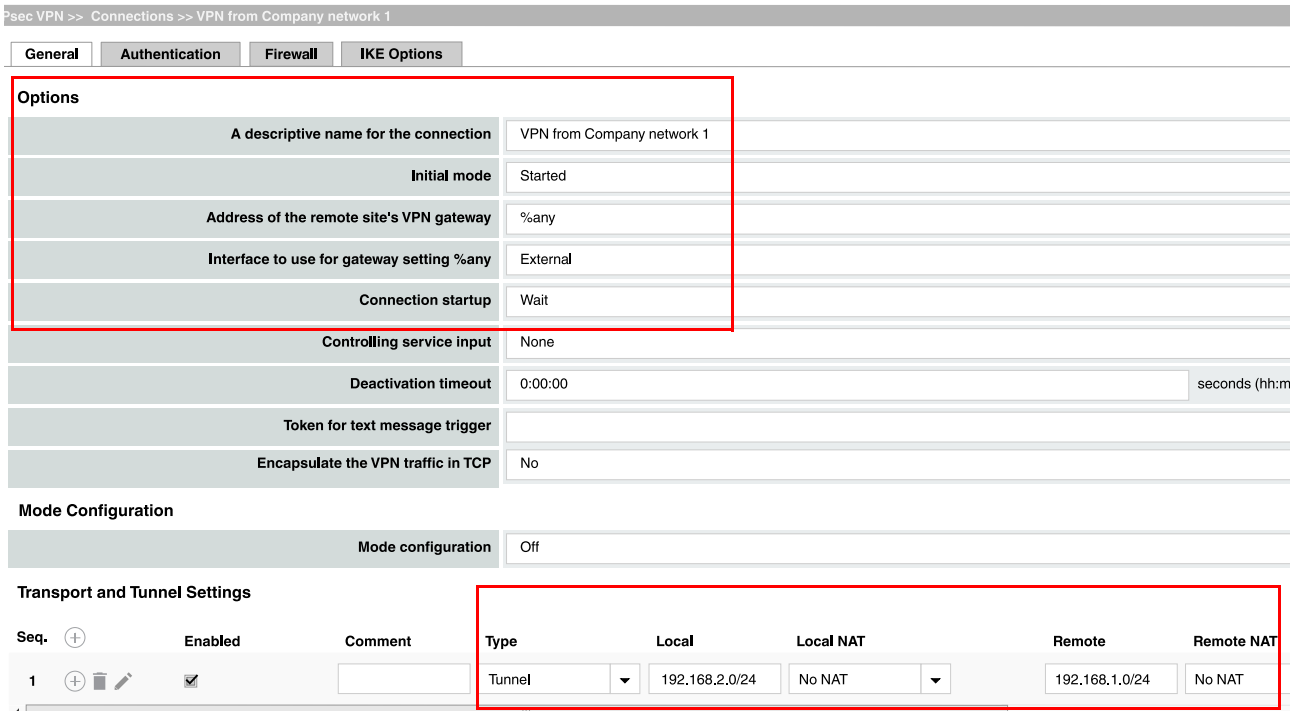
1. Go to **IPsec VPN >> Connections** (*Authentication* tab)
2. In the **Local X.509 certificate** field, select the certificate (*mGuard1*) that you previously imported into the device as the machine certificate for *mGuard1*.
3. In the **Remote CA certificate** field, select the option *No CA certificate and select the remote certificate below instead*.
4. In the **Remote certificate** field, import the *mGuard2* client certificate.
   To do so, click on the icon and select the certificate (*mGuard2.pem*) saved onto the configuration computer. Then click on the **Upload** button.
5. Click on the "*Save*" icon to complete the procedure.

11.5 Creating the mGuard2 VPN connection

11.5.1 Prerequisite

The same prerequisites apply here as to mGuard 1 (see “Prerequisite” on page 71).

11.5.2 Configuring the VPN connection



Repeat the configuration steps described above (*mGuard1*) for the VPN peer (*mGuard2*). Note the following differences:

IPsec VPN >> Connections (*General* tab)

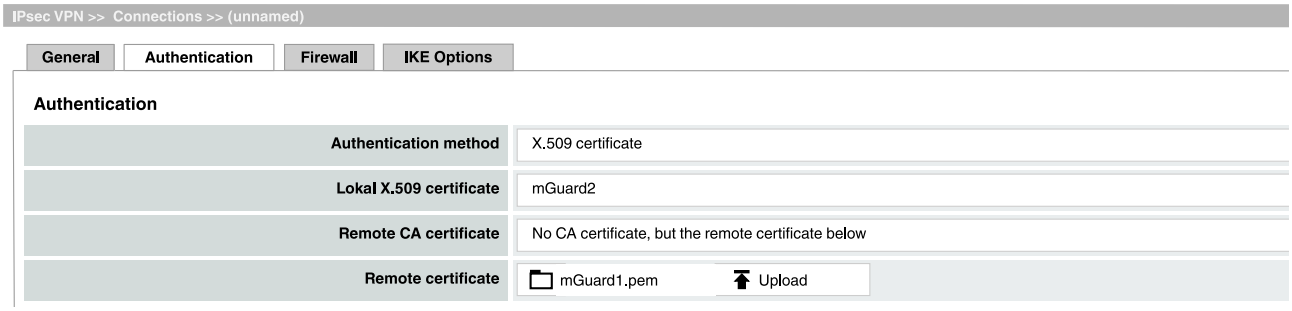
Section "Options"

1. Enter *%any as* **Address of the remote site’s VPN gateway**.
2. Enter *External u*nder **Interface to use for gateway setting %any**.
3. Select *Wait* in the **Connection startup** field.

Section "Transport and Tunnel Settings"

1. Enter the address of the network that is to be accessible via the *mGuard2* internal interface in the field **Local** (192.168.**2**.0/24).
2. Enter the address of the network that is to be accessible via the *mGuard1* internal interface in the field **Remote** (192.168.**1**.0/24).
3. Click on the "*Save*" icon to complete the procedure.

11.5.3 Configuring authentication of the VPN connection



Repeat the configuration steps described above (*mGuard1*) for the VPN peer (*mGuard2*). Note the following differences:

IPsec VPN >> Connections (*Authentication* tab)

1. In the **Local X.509 certificate** field, select the certificate (*mGuard2*) that you previously imported into the device as the machine certificate for *mGuard2*.
2. In the **Remote CA certificate** field, select the option *No CA certificate and select the remote certificate below instead*.
3. In the **Remote certificate** field, import the *mGuard1* client certificate.
   To do so, click on the icon and select the certificate (*mGuard1.pem*) saved onto the configuration computer. Then click on the **Upload** button.
4. Click on the "*Save*" icon to complete the procedure.

11.6 Testing the VPN connection

11.6.1 Prerequisite

– Connect the two configured mGuard devices into the corresponding network environments.
– Optional: ensure that a connection to the Internet can be established (UDP ports 500 and 4500 must be open).

11.6.2 Procedure

1. Log into the *mGuard 1* or *mGuard 2* web interface (e.g. https://192.168.1.254).
2. Go to **IPsec VPN >> IPsec Status**.
3. On the status page, check whether a VPN connection between the two devices (*mGuard1* and *mGuard2*) exists.

   Both an ISAKMP and an IPsec SA connection must have been established.
4. Check the secure VPN connection by either pinging the respective VPN peer or by testing access to a peer (e.g. web server, controller, computer) in the remote network.

12 Configuring VPN connections with various network modes

Document ID: 108410_en_00
Document designation: AH EN MGUARD IPSEC VPN NW MODE

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes the configuration of IPsec VPN connections between two mGuard devices with different network modes (*Router*, *Stealth*).

The examples show the configuration under **IPsec VPN >> Connections >> (*Edit*) >> General**.

12.1 Introduction

VPN connections are configured via the menu **IPsec VPN >> Connections** in four tabs.

Configuration in the *Authentication, Firewall* and *IKE Options* tabs is carried out independenty of the general network properties of the mGuard device, such as **network mode** (e.g. *Stealth*, *Router*, *Router/PPPoE*) or **VPN function** (e.g. *1: 1 NAT* for the local network, *hub and spoke*).

In the *General* tab, however, these properties have an effect on the tunnel settings; various properties in the *General* tab will therefore be considered in the following examples.

12.2 VPN transport connection (Stealth <-> Stealth)

12.2.1 Introduction

In contrast to a VPN tunnel connection that connects two networks, a VPN transport connection is used to link two individual clients (hosts).

If the VPN transport connection is used between two mGuard devices in the *Router* network mode, it is not possible to access all clients in the internal network of the devices via the VPN connection.

Using a transport connection is therefore only meaningful if the mGuard devices are operated in the *Single Stealth mode* (e.g. to secure data transfer between two clients or to access a client via a secure connection for maintenance purposes). The devices must be in the same network.

A transport connection cannot be used if the connection is established via one or more gateways in which Network Address Translation (NAT) is enabled.

12.2.2 Example

Two clients (hosts) in the same network are to be connected via an IPsec VPN connection in order to ensure permanent encrypted data exchange. Figure 12-1 shows the network configuration of the participating clients.

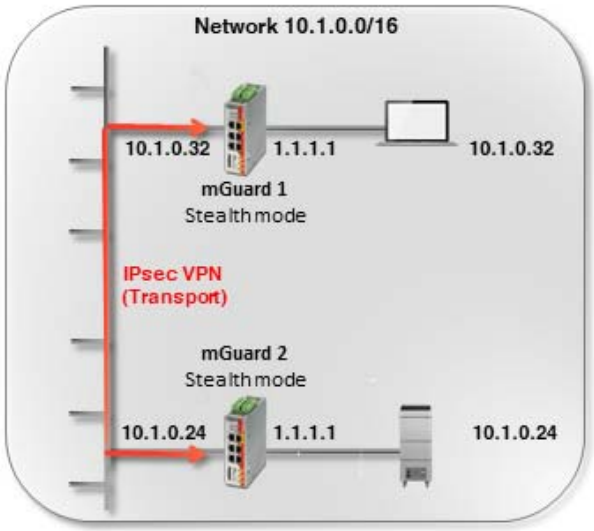


Figure 12-1 VPN transport connection in the *Stealth* network mode

The VPN connection (type: *transport*) is established and made available via two mGuard devices in the *Stealth* (*Automatic*) network mode connected upstream of the respective clients.

In *Stealth mode* (*Automatic*), the two mGuard devices adopt the IP and MAC address of their respective internal clients (*mGuard 1* adopts 10.1.0.32 and *mGuard 2*: 10.1.0.24).

12.2.3 Configuring the VPN connection

Figure 12-2 shows the configuration of the mGuard devices (in illustrated form for the sake of clarity). The transport and tunnel settings are the same on both devices.

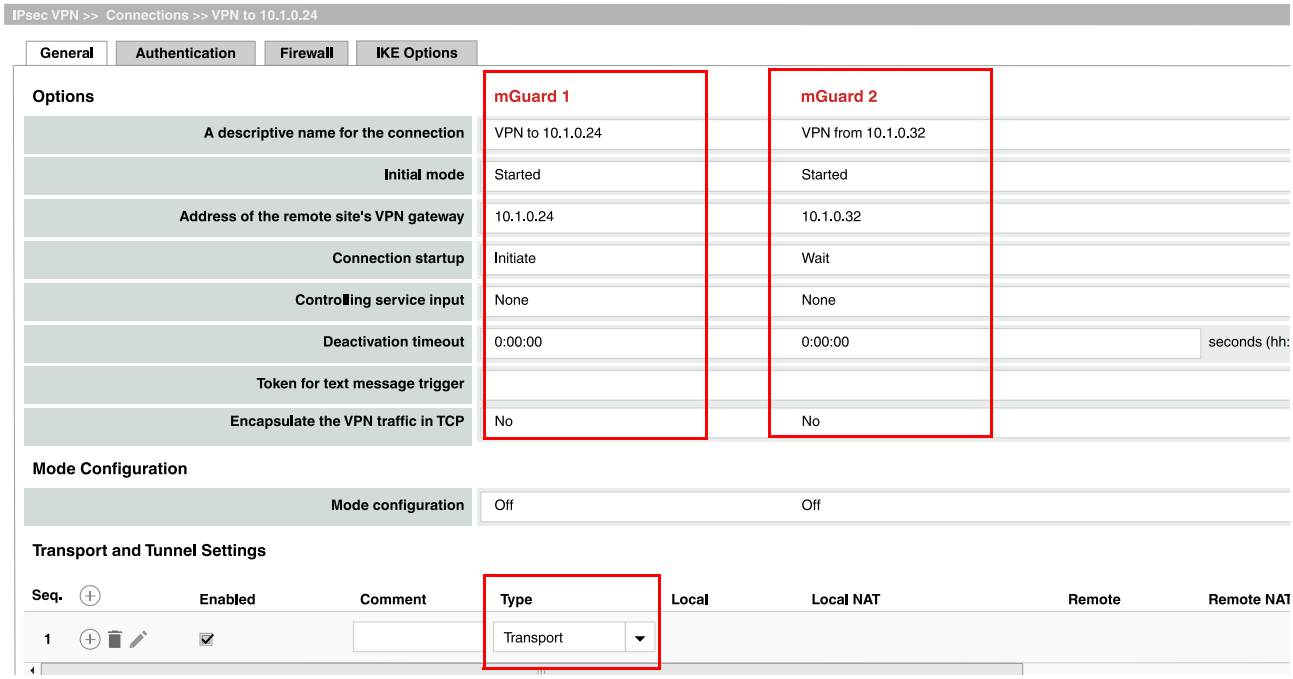


Figure 12-2 VPN connection (type: *transport*): Stealth mode <-> Stealth mode

To configure the VPN connection of the mGuard devices, proceed as follows:

1. Go to **IPsec VPN >> Connections**.
2. Click on the ⊕ icon to add a new VPN connection.
3. Specify a unique name for the connection and click on the ✎ icon to edit the connection.
4. Configure the VPN connection in accordance with Figure 12-2 or Table 12-1.

Table 12-1 Configuring VPN connection (*IPsec VPN* >> *Connections* >> *(Edit)* >> *General*)

Section	Parameter	mGuard 1	mGuard 2
Options	**A descriptive name for the connection**	VPN to 10.1.0.24	VPN from 10.1.0.32
	Address of the remote site's VPN gateway	10.1.0.24	10.1.0.32
	Connection startup	Initiate	Wait
Transport and tunnel settings	**Type**	Transport	Transport

Result

The two clients, each of which are connected to the network via an mGuard device in the *Stealth* network mode, communicate via the encrypted IPsec VPN connection established between the mGuard devices (type: *transport*).

A *transport connection* only ever connects two individual clients (hosts), not networks – as is the case with a *tunnel connection*.

12.3 VPN tunnel connection (Router <-> Router)

12.3.1 Introduction

In contrast to a VPN transport connection that connects two individual hosts, a VPN tunnel connection is used to connect two networks.

12.3.2 Example

An IPsec VPN tunnel is to be established between **company network 1** (192.168.1.0/24) and **company network 2** (192.168.2.0/24) using two mGuard devices.

A VPN tunnel can only be established between different networks. If two locations have the same internal network, the VPN 1:1 NAT function has to be used for the local network (see “Using NAT in VPN connections” on page 89).

In this case *mGuard 1* initiates the VPN connection. *mGuard 2* waits for the connection. Both mGuard devices are operated in the *Router* (*static*) network mode.

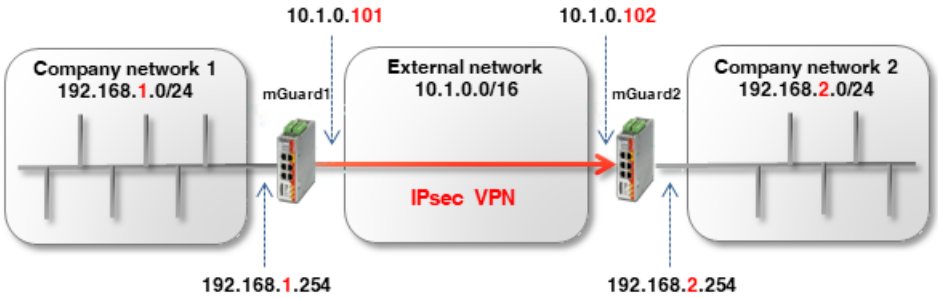


Figure 12-3 Connecting two networks via IPsec VPN

The network settings of the interfaces of the two mGuard devices are configured in the menu **Network >> Interfaces** (tabs: *General*, *External*, *Internal*). Both devices are operated in the *Router* (*static*) network mode.

Table 12-2 Network configuration of the interfaces

Parameter	mGuard 1	mGuard 2
External IP address	10.1.0.101	10.1.0.102
Netmask	255.255.0.0	255.255.0.0
Default gateway	10.1.0.254	10.1.0.254
Internal IP address	192.168.1.254	192.168.2.254
Netmask	255.255.255.0	255.255.255.0

The clients in the internal networks are to use the internal IP address of the respective mGuard device as the default gateway.

Optional setup in the PPPoE router mode

Establishing a VPN tunnel between two mGuard devices in the *PPPoE* router mode via the Internet is similar in principle (see Figure 12-4). In this case, the Internet is the external network. The devices receive their dynamically assigned public (external) IP addresses from the Internet Service Provider (ISP).

In order to enable static name resolution under these circumstances, the devices must register their current IP addresses under a fixed name with a DynDNS provider.

The initiating mGuard device (*mGuard 1*) must then provide a reference to the DynDNS name of the responding mGuard device (e.g. *mGuard2.dyndns.org*) in order to establish a VPN connection.

In this case, activate **DynDNS Monitoring** (**IPsec VPN >> Global >> DynDNS Monitoring**) in the VPN connection of the initiating device (*mGuard 1*). Otherwise, the device will not know when the IP address of the remote peer has changed and the VPN connection will not be established.

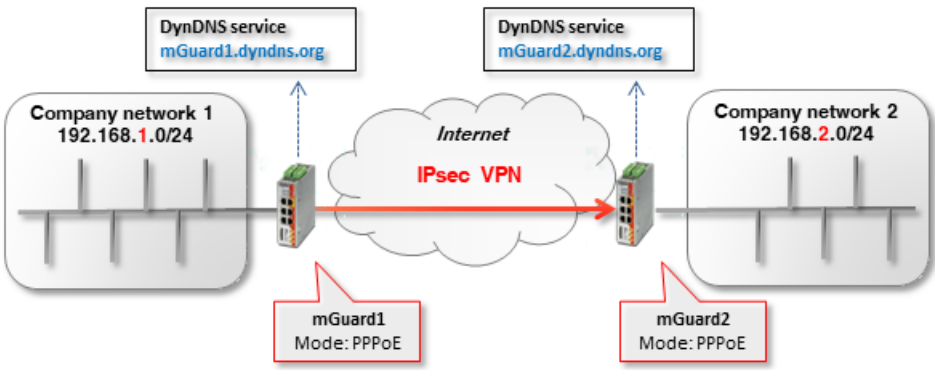


Figure 12-4 Connecting two networks via IPsec VPN (*Router/PPPoE* <-> *Router/PPPoE*). The host names for the mGuard devices are determined using DynDNS. (Because in this example the VPN connection is initiated by *mGuard 1*, in principle it does not need a DynDNS address.)

12.3.3 Configuring the VPN connection

Configure the VPN connection in accordance with Figure 12-5 and 12-6 or Table 12-3.

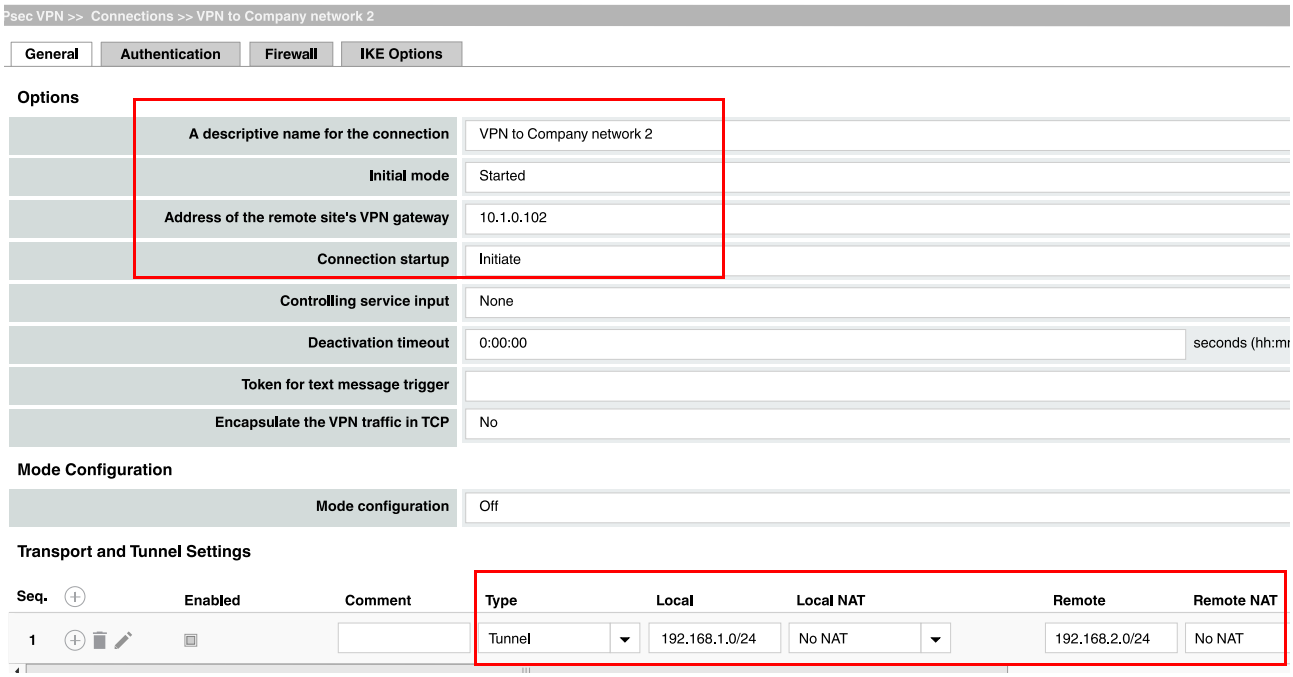


Figure 12-5 *mGuard 1* (*initiator*): VPN connection configuration

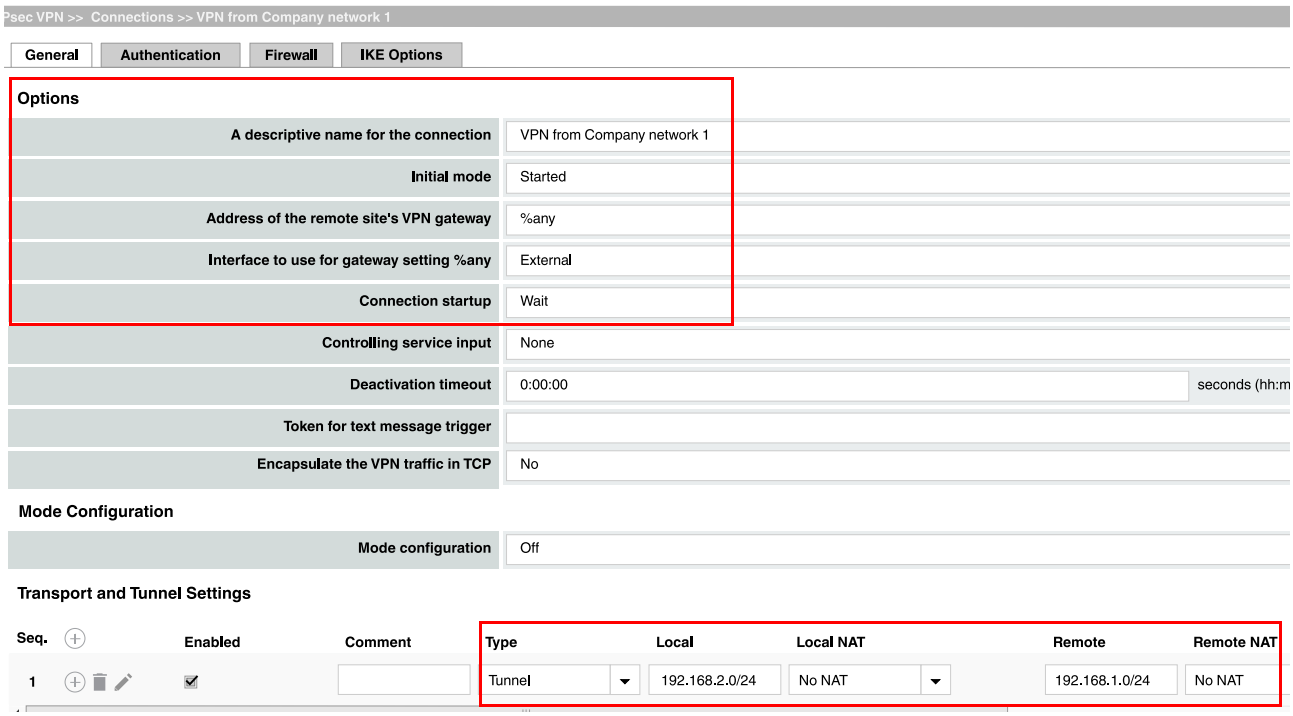


Figure 12-6 *mGuard 2* (*responder*): VPN connection configuration

To configure the VPN connection of the mGuard devices, proceed as follows:

1. Go to **IPsec VPN >> Connections**.
2. Click on the ⊕ icon to add a new VPN connection.
3. Specify a unique name for the connection and click on the ✎ icon to edit the connection.
4. Configure the VPN connection in accordance with Figure 12-5 and 12-6 or Table 12-3.

Table 12-3 Configuring VPN connection (*IPsec VPN* >> *Connections* >> *(Edit)* >> *General*)

Section	Parameter	mGuard 1	mGuard 2
Options	**A descriptive name for the connection**	VPN to company network 2	VPN from company network 1
	Address of the remote site's VPN gateway	10.1.0.102	%any
	Interface to use for gateway setting %any	(field not visible)	External
	Connection startup	Initiate	Wait
Transport and tunnel settings	**Type**	Tunnel	Tunnel
	Local	192.168.1.0/24	192.168.2.0/24
	Remote	192.168.2.0/24	192.168.1.0/24

Result

The two networks are connected via an IPsec VPN tunnel. Communication between each client and clients of the other network can be encrypted.

A *tunnel connection* always connects networks (incl. networks with the subnet mask /32), and not just two individual clients (hosts) – as is the case with *transport connections*.

12.4 VPN tunnel connection (Single Stealth <-> Router)

12.4.1 Introduction

If a VPN connection is established between two mGuard devices one of which is operated in *Single Stealth mode* (= *static* or *automatic*), it is possible that the IP address of the assigned client is controlled dynamically via a DHCP server. If this IP address is changed, the IP address of the mGuard device also changes in *Stealth mode*.

A *virtual IP address* is used in this case so the VPN configuration of the mGuard devices does not have to be changed. The device then automatically forwards the packets that are sent to this *virtual IP address* via the VPN tunnel to the real IP address of the client.

12.4.2 Example

An IPsec VPN tunnel is to be established between **company network 1** (10.1.0.0/16) and **company network 2** (192.168.2.0/24) using two mGuard devices.

Here, an mGuard device in *Single Stealth mode* (*static* or *automatic*) is to establish a VPN tunnel to an mGuard device in the *Router* network mode (*static* or *PPPoE*).

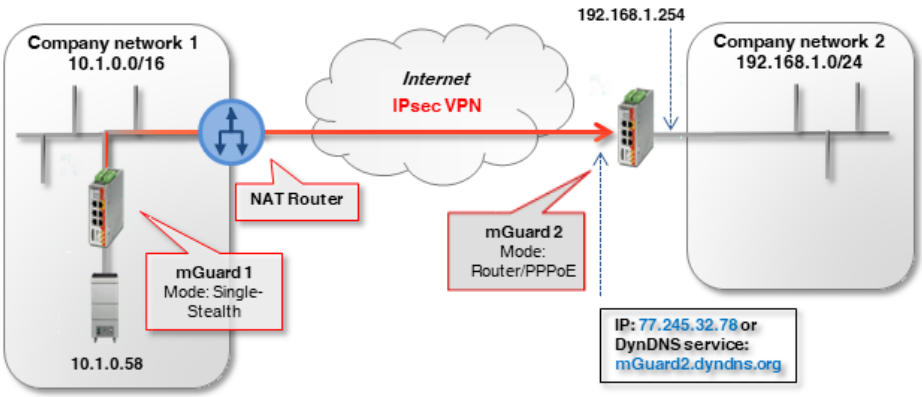


Figure 12-7 Connecting two networks via IPsec VPN (*Single Stealth <-> Router*)

In this example, the responding mGuard device (*mGuard 2*) can be reached from the Internet via a static public IP address.

If the mGuard device is connected to the Internet via changing (dynamic) IP addresses, its current IP address must be registered with a DynDNS provider under a fixed name.

The initiating mGuard device (*mGuard 1*) in *Stealth mode* must then provide a reference to the DynDNS name of the responding mGuard device (e.g. *mGuard2.dyndns.org*) in order to establish a VPN connection.

In this case, activate **DynDNS Monitoring** (**IPsec VPN >> Global >> DynDNS Monitoring**) in the VPN connection of the initiating device (*mGuard 1*). Otherwise, the device will not know when the IP address of the remote peer has changed and the VPN connection will not be established.

12.4.3 Configuring the VPN connection

The *mGuard 1* device initiates the VPN tunnel. In the *Stealth mode* (*automatic*), *mGuard 1* adopts the IP and MAC address of its respective client (10.1.0.58). In *Stealth mode* (*static*), the IP address are entered as fixed addresses.

The responding *mGuard 2* in *Router mode* (*PPPoE*) can be reached via the static public (external) IP address (77.245.32.78) via the Internet. With its internal IP address (192.168.1.254), the device acts as the default gateway for the connected clients in the network 192.168.1.0/24.

If the client receives its IP settings from a DHCP server, it can, in principle, change its IP address. In order for a configured VPN tunnel to continue to be established even with a dynamic change of IP address, a *virtual IP address must* be specified in the settings which is then used by a peer as the endpoint of the VPN tunnel.

If in our example the client in company network 1 (10.1.0.58) is to be accessed from company network 2 via a VPN tunnel, it *must* be accessed via the virtual address (e.g. 172.16.1.1/32).

mGuard 1 would then automatically perform a 1:1 NAT from the *virtual IP address* (172.16.1.1/32) to the real IP address of the client (10.1.0.58/32).

To configure the VPN connection of the mGuard devices, proceed as follows:

1. Go to **IPsec VPN >> Connections**.
2. Click on the ⊕ icon to add a new VPN connection.
3. Specify a unique name for the connection and click on the ✎ icon to edit the connection.
4. Configure the VPN connection in accordance with Table 12-4.

Table 12-4 Configuring VPN connection (*IPsec VPN >> Connections >> (Edit) >> General*)

Section	Parameter	mGuard 1 (Stealth)	mGuard 2
Options	**A descriptive name for the connection**	VPN to company network 2	VPN from company network 1
	Address of the remote site's VPN gateway	77.245.32.78	%any
	Interface to use for gateway setting %any	-----	External
	Connection startup	Initiate	Wait
Transport and tunnel settings	**Type**	Tunnel	Tunnel
	Local	172.16.1.1/32	192.168.1.0/24
	Remote	192.168.1.0/24	172.16.1.1/32
	Virtual IP	172.16.1.1	-----

12.5 VPN tunnel connection (Multi Stealth <-> Router)

12.5.1 Introduction

In *Multi Stealth mode*, in contrast to *Single Stealth mode* (*automatic* or *static*), more than one computer can be connected to the LAN interface of the mGuard device, and therefore several IP addresses can be used at the LAN interface.

12.5.2 Example

An IPsec VPN tunnel is to be established between **company network 1** (10.1.0.0/16) and **company network 2** (192.168.2.0/24) using two mGuard devices.

Here, an mGuard device in the *Stealth* (*multiple clients*) network mode is to establish a VPN tunnel to an mGuard device in the *Router* network mode (*static* or *PPPoE*). The clients behind the mGuard device in company network 1 (*mGuard 1*) should be accessible via a VPN tunnel.

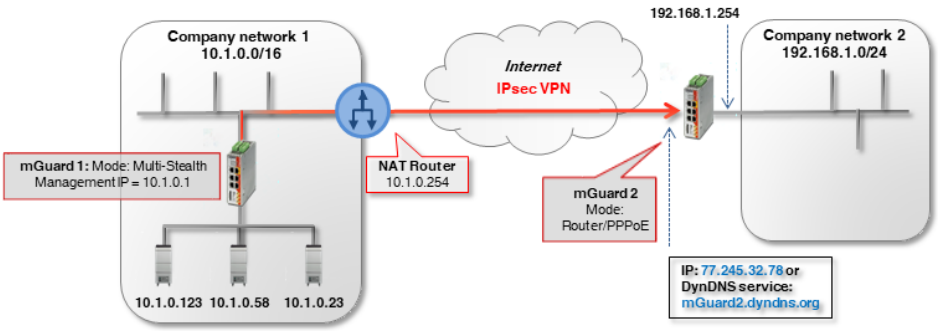


Figure 12-8 Connecting two networks via IPsec VPN (Multi Stealth <-> Router)

In this example, the responding mGuard device (*mGuard 2*) can be reached from the Internet via a static public IP address.

If the mGuard device is connected to the Internet via changing (dynamic) IP addresses, its current IP address must be registered with a DynDNS provider under a fixed name (see Section 12.4.1).

The network settings of the interfaces of the two mGuard devices are configured in the menu **Network >> Interfaces** (tabs: *General*, *Stealth*, *Internal*).

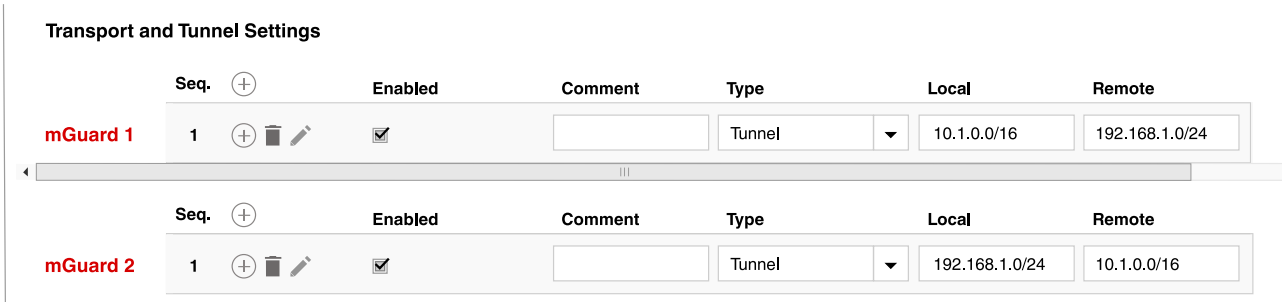
Table 12-5 Network configuration of the interfaces

Parameter	mGuard 1 (Multi Stealth)	mGuard 2 (Router)
Stealth Management IP Address	10.1.0.1	-----
Netmask	255.255.0.0	-----
Default gateway	10.1.0.254	-----
Internal IP address	-----	192.168.1.254
Netmask	-----	255.255.255.0

12.5.3 Configuring the VPN connection

The VPN connection is initiated by *mGuard 1*. To be able to use the VPN function in Stealth mode (*multiple clients*), a *Management IP address* must be assigned to the device. This IP address must belong to the same network as the mGuard device. It may not be used by any other device in the network.

The waiting device *mGuard 2* has the static public IP address 77.245.32.78.



To configure the VPN connection of the mGuard devices, proceed as follows:

1. Go to **IPsec VPN >> Connections**.
2. Click on the ⊕ icon to add a new VPN connection.
3. Specify a unique name for the connection and click on the ✎ icon to edit the connection.
4. Configure the VPN connection in accordance with Table 12-6.

Table 12-6 Configuring VPN connection (*IPsec VPN >> Connections >> (Edit) >> General*)

Section	Parameter	mGuard 1 (Stealth)	mGuard 2
Options	**A descriptive name for the connection**	VPN to company network 2	VPN from company network 1
	Address of the remote site's VPN gateway	77.245.32.78	%any
	Interface to use for gateway setting %any	-----	External
	Connection startup	Initiate	Wait
Transport and tunnel settings	**Type**	Tunnel	Tunnel
	Local	10.1.0.0/16	192.168.1.0/24
	Remote	192.168.1.0/24	10.1.0.0/16

13 Using NAT in VPN connections

Document ID: 108411_en_00
Document designation: AH EN MGUARD IPSEC VPN NAT

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes the configuration of IPsec VPN connections using 1:1 NAT and IP masquerading.

13.1 Introduction

A VPN connection can normally only be established between different networks (e.g. network A: 192.168.**1**.0/24 <-> network B: 192.168.**2**.0/24).

If the same internal networks (e.g. 192.168.**1**.0/24) are used at two locations, the following problems can arise:

1. If the locations are connected via a VPN tunnel, this would lead to routing problems. It would not be clear which network should receive the packets that are sent to IP addresses of the internal network which is the same on both sides.
   This problem can be avoided by using of **1:1 NAT** (see Section 13.2).
2. If several locations with partially identical internal networks are connected to a central location via a VPN tunnel, this would also lead to routing problems. This problem can be avoided by using **1:1 NAT** or partially avoided by using **IP masquerading** (see Section 13.3 and 13.5).

13.1.1 1:1 NAT

1:1 NAT means that the **network part** of an IP address is assigned to another network and the **host part** remains unchanged (e.g. **192.168.1.102**/24 <-> **192.168.2.102**/24). The network part is defined via the subnet mask.

Here, a *Real network* (e.g. the internal network) is assigned to a *Virtual network* in order to circumvent existing network overlapping. The VPN tunnels are then established via *Virtual* instead of *Real networks*.

13.1.2 IP masquerading

IP Masquerading is a special type of NAT. It must be enabled on gateways that connect private networks to the Internet in order to be able to access the Internet.

When accessing a website from an internal network, the gateway (NAT router) replaces the private IP address of the sender (e.g. 192.168.1.100) with its own public IP address (e.g. 77.245.32.78). The destination web server therefore knows which public address it should reply to.

This then replaces the reply of the web server to the NAT router (77.245.32.78) with the IP address of the original sender (192.168.1.100) and forwards it to the client in the internal network.

IP masquerading is only used in one direction, e.g. from the internal to an external network or the Internet. A client in the internal network (e.g. 192.168.1.100) could then access destinations in the external network or on websites in the Internet, but would not be accessible via its private IP address from the external network or the Internet.

IP masquerading in VPN connections

IP masquerading in VPN networks provides the same functionality, however within a VPN connection.

If data packets are sent to a remote network via the VPN tunnel, the mGuard device replaces the IP address of the sender with a specific, unique IP address and reverses the masquerading upon receipt of the answer from the remote network.

The great advantage here is that the entire real (local) network is *masked* by a single IP address.

If several VPN connections end at a central VPN gateway, this function reduces the necessary address space for the VPN connections and makes VPN configuration clearer.

13.2 Connecting locations with the same internal networks (1:1 NAT)

13.2.1 Example

Two locations with the same internal network (192.168.1.0/24) are to be connected via a VPN tunnel. For this, **local NAT for IPsec tunnel connections (1:1 NAT)** must be used on both mGuard devices.

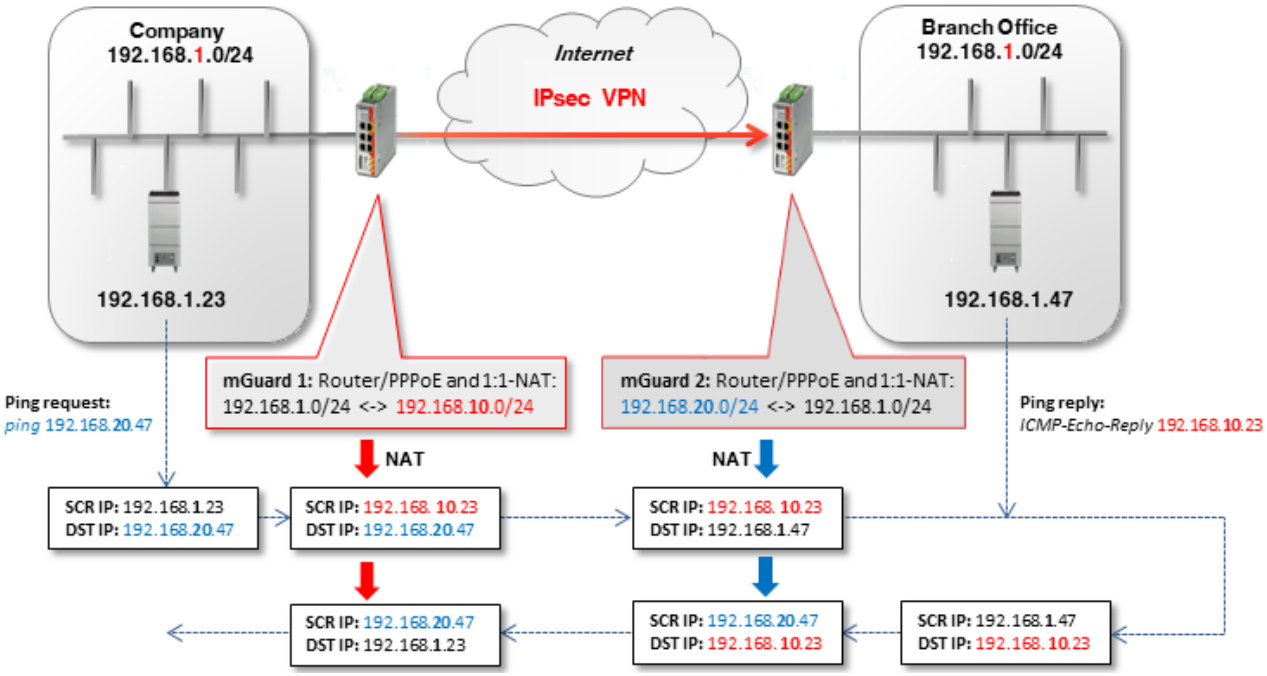


Figure 13-1 Same internal networks: ping request via a VPN tunnel using local 1:1 NAT

- **mGuard 1** performs 1:1-NAT: 192.168.**1**.0/24 <-> 192.168.**10**.0/24.
  The network part is rewritten and the host part is retained. The clients can therefore be reached in the company network via the VPN tunnel in the *Virtual network* 192.168.**10**.0/24.
- **mGuard 2** also performs 1:1-NAT: 192.168.**1**.0/24 <-> 192.168.**20**.0/24.
  The clients in the branch network can be reached via the VPN tunnel in the *Virtual network* 192.168.**20**.0/24.

13.2.2 Configuring the VPN connection

The VPN tunnel must be established between *Virtual networks*. For this, a local 1:1 NAT is performed on both devices.

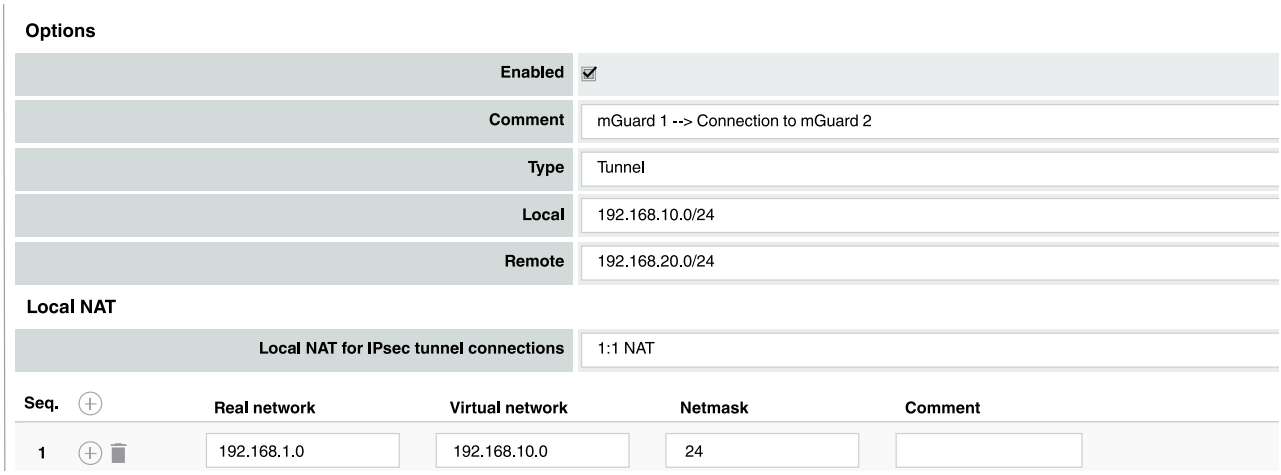


Figure 13-2 **mGuard 1**: IPsec VPN >> General (tunnel setting with 1:1 NAT)

To configure the VPN connection of the mGuard devices, proceed as follows:

1. Go to **IPsec VPN >> Connections**.
2. Click on the ⊕ icon to add a new VPN connection.
3. Give the connection a unique name and click on the ✎ icon.
4. Under **Transport and tunnel settings**, click on the icon ✎.
5. Configure the VPN connection in accordance with Table 13-1 and Figure 13-2.

Table 13-1 Configuring the VPN connection

Section	Parameter	Company / mGuard 1	Branch / mGuard 2
IPsec VPN >> Connections >> (Edit) >> General			
Options	**A descriptive name for the connection**	VPN to the branch	VPN from the company
	Address of the remote site's VPN gateway	77.245.32.78	%any
	Interface to use for gateway setting %any	-----	External
	Connection startup	Initiate	Wait
Transport and tunnel settings >> (Edit) >> General			
Transport and tunnel settings	**Type**	Tunnel	Tunnel
	Local	192.168.**10**.0/24	192.168.**20**.0/24
	Remote	192.168.**20**.0/24	192.168.**10**.0/24
Local NAT	**Local NAT for IPsec tunnel connections**	1:1 NAT	1:1 NAT
	Real network	192.168.**1**.0	192.168.**1**.0
	Virtual network	192.168.**10**.0	192.168.**20**.0
	Netmask	24	24

Result

– Packets to the company network in the internal network of *mGuard 1* must be sent to the *Virtual network* 192.168.**10**.0/24.
– Packets to the branch network in the internal network of *mGuard 2* must be sent to the *Virtual network* 192.168.**20**.0/24.

13.3 Connecting locations with the same internal networks to a control center (1:1 NAT)

13.3.1 Example

Two locations that use the same internal network (192.168.1.0/24) are to be connected simultaneously to the company control center via a VPN tunnel. For this, **local NAT for IPsec tunnel connections (1:1 NAT)** must be used on both mGuard devices.

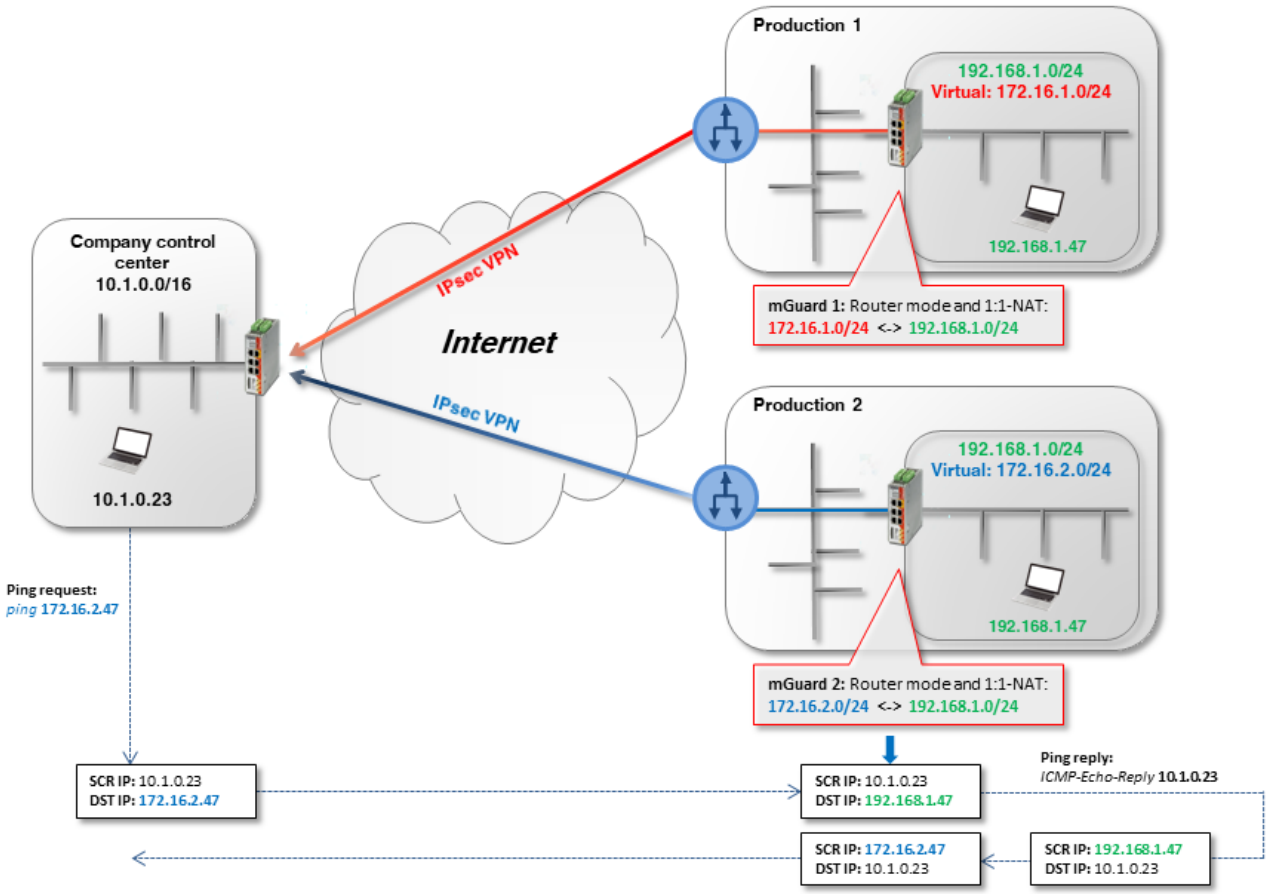


Figure 13-3 (Example *mGuard 2*) The same internal networks: ping request (to production 2) from the company control center via the VPN tunnel using local 1:1 NAT

- **mGuard 1** performs 1:1-NAT: 192.168.1.0/24 <-> 172.16.**1**.0/24).
  The clients in its internal network (**production 1**) can be reached via the VPN tunnel in the *Virtual network* 172.16.**1**.0/24.
- **mGuard 2** performs 1:1-NAT (192.168.1.0/24 <-> 172.16.**2**.0/24).
  The clients in its internal network (**production 2**) can be reached via the VPN tunnel in the *Virtual network* 172.16.**2**.0/24.

Configuring VPN connection

Two VPN connections must be configured on the mGuard device of the control center and a local 1:1 NAT must be performed on each. Here, the *Virtual network* of mGuard **1** or **2** must be specified in the tunnel settings as the peer (172.16.**1**.0/24 or 172.16.**2**.0/24).

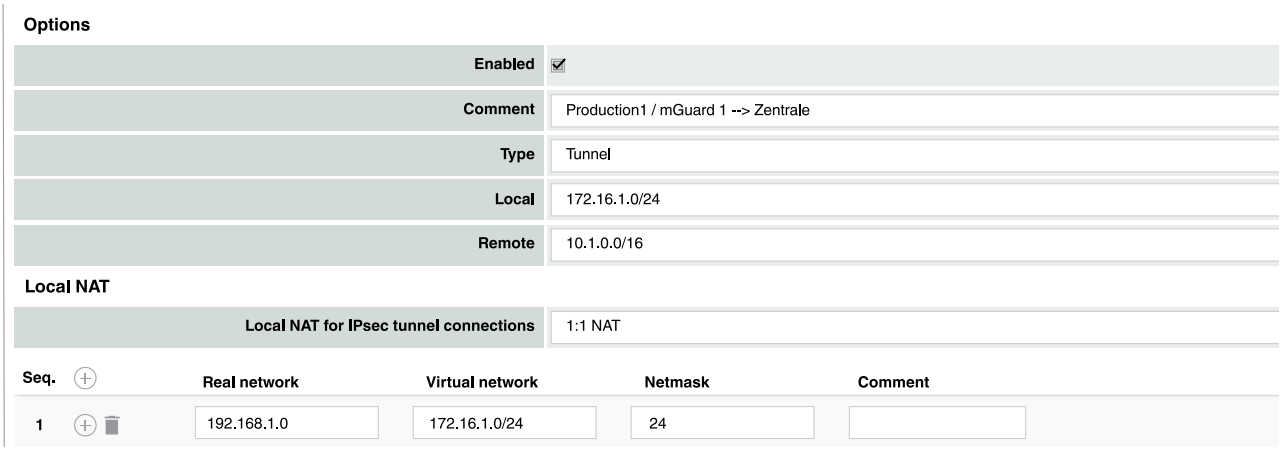


Figure 13-4 **mGuard 1**: IPsec VPN >> General (tunnel settings with 1:1-NAT)

To configure the VPN connection of the mGuard devices, proceed as follows:

1. Go to **IPsec VPN >> Connections**.
2. Click on the ⊕ icon to add a new VPN connection.
3. Give the connection a unique name and click on the ✎ icon.
4. Under **Transport and tunnel settings**, click on the icon ✎.
5. Configure the VPN connection in accordance with Table 13-2 and Figure 13-3.

Table 13-2 Configuring the VPN connection

Section	Parameter	Production mGuard 1	Production mGuard 2	Control center
IPsec VPN >> Connections >> (Edit) >> General				
Options	**A descriptive name for the connection**	VPN to control center	VPN to control center	To production (**1** or **2**)
	Address of the remote site's VPN gateway	77.245.32.78	77.245.32.78	%any
	Interface to use for gateway setting %any	-----	-----	External
	Connection startup	Initiate	Initiate	Wait
Transport and tunnel settings >> (Edit) >> General				
Transport and tunnel settings	**Type**	Tunnel	Tunnel	Tunnel
	Local	172.16.**1**.0/24	172.16.**2**.0/24	10.1.0.0/16
	Remote	10.1.0.0/16	10.1.0.0/16	172.16.**1**.0/24 or 172.16.**2**.0/24
Local NAT (Only mGuard 1 or 2)	**Local NAT for IPsec tunnel connections**	1:1 NAT	1:1 NAT	
	Real network	192.168.**1**.0	192.168.**1**.0	
	Virtual network	172.16.**1**.0/24	172.16.**2**.0/24	
	Netmask	24	24	

Result

Packets to the network **production 1** (in the internal network of *mGuard 1*) or **production 2** (in the internal network of *mGuard 2*) must be sent to the *Virtual network* 172.10.**1**.0/24 or 172.16.**2**.0/24.

13.4 Connecting locations with the same internal networks to a control center (masquerading)

The control center is to be connected to several external locations (production) using a central VPN gateway via VPN tunnel. Some of the external locations use the same internal networks or the same internal network as the control center.

13.4.1 Example 1: Transmission in one direction (IP masquerading)

IP masquerading can be used if the data is only to be transmitted in one direction – from the machine controllers to the control center.

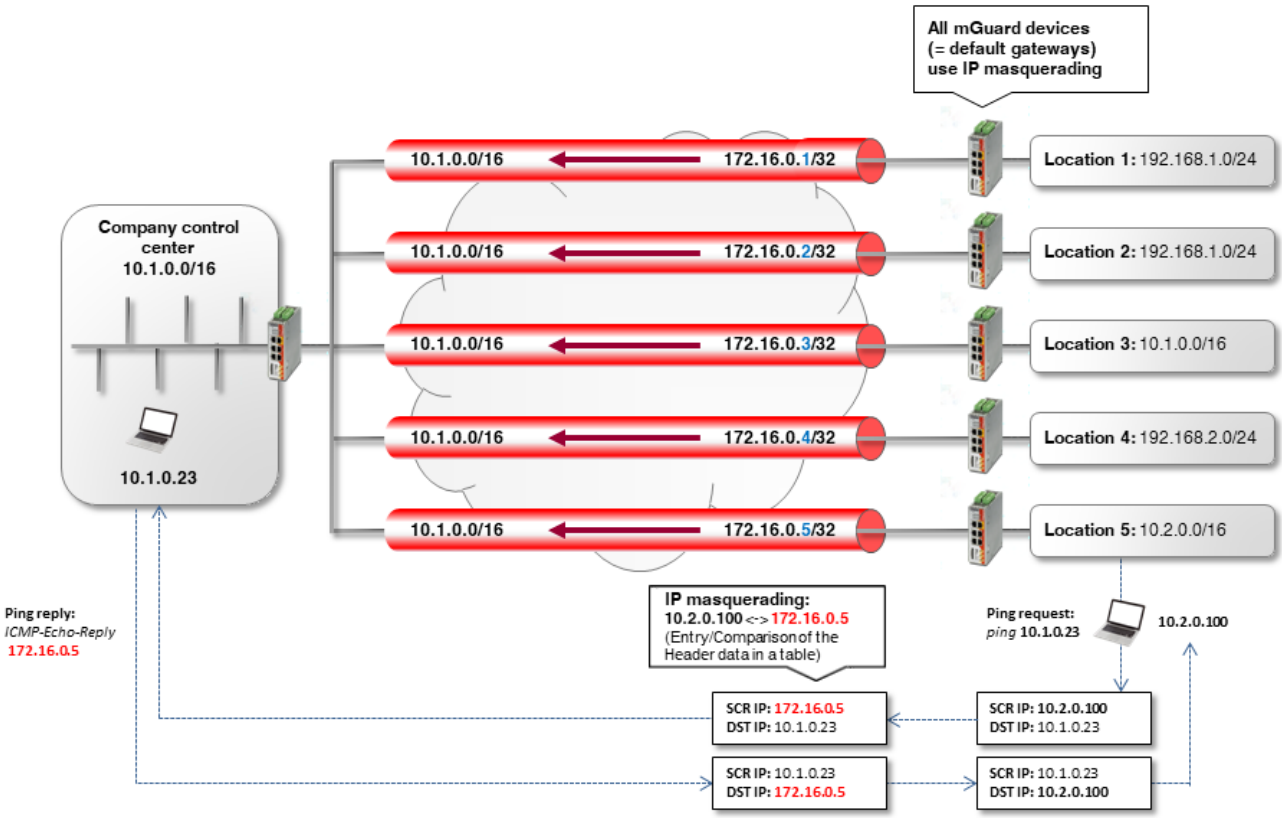


Figure 13-5 **Transmission in just one direction (IP masquerading)**: clients (e.g. PLCs) in the external networks can send data to the control center via VPN. However, the control center **cannot** access the clients. The respective mGuard device is the default gateway of the internal client.

Configuring VPN connection

To be able to establish connections with the control center from all locations, IP masquerading must be used at every location. In this case, the IP address used for masquerading can simply be increased at each location.

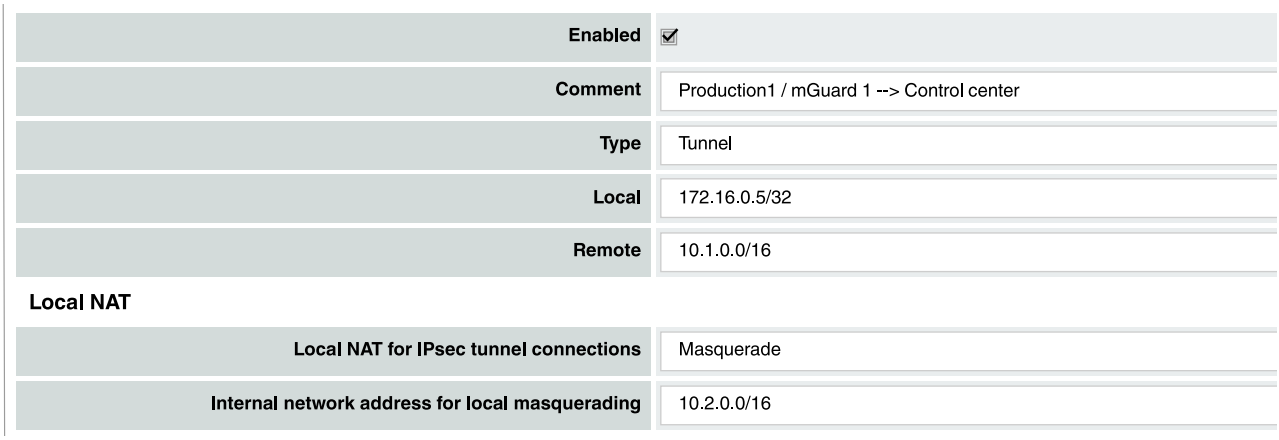


Figure 13-6 Configuration example *Location 5* (tunnel settings with IP masquerading)

Table 13-3 Configuring VPN connection

Section	Parameter	Control center	Location 5
IPsec VPN >> Connections >> (Edit) >> General			
Options	**A descriptive name for the connection**	VPN from Location 5	VPN to control center
	Address of the remote site's VPN gateway	%any	77.245.32.78
	Interface to use for gateway setting %any	External	-----
	Connection startup	Wait	Initiate
Transport and tunnel settings >> (Edit) >> General			
Transport and tunnel settings	**Type**	Tunnel	Tunnel
	Local	10.1.0.0/16	172.16.0.5/32
	Remote	172.16.0.5/32	10.1.0.0/16
Local NAT	**Local NAT for IPsec tunnel connections**	No NAT	Masquerade
	Internal network address for local masquerading	-----	10.2.0.0/16

Result

The clients in the network of the control center can be reached via their real IP addresses.

Advantages

Configuring VPNs is uncomplicated and easy to understand. The address space for the peers is reduced.

Disadvantages

The VPN connections can only be used in one direction. In the above example, only the locations can access the control center.

13.4.2 Example 2: Transmission in both directions (1:1 NAT)

If data is to be transmitted in both directions, local 1:1 NAT must be used (see also Section 13.3).

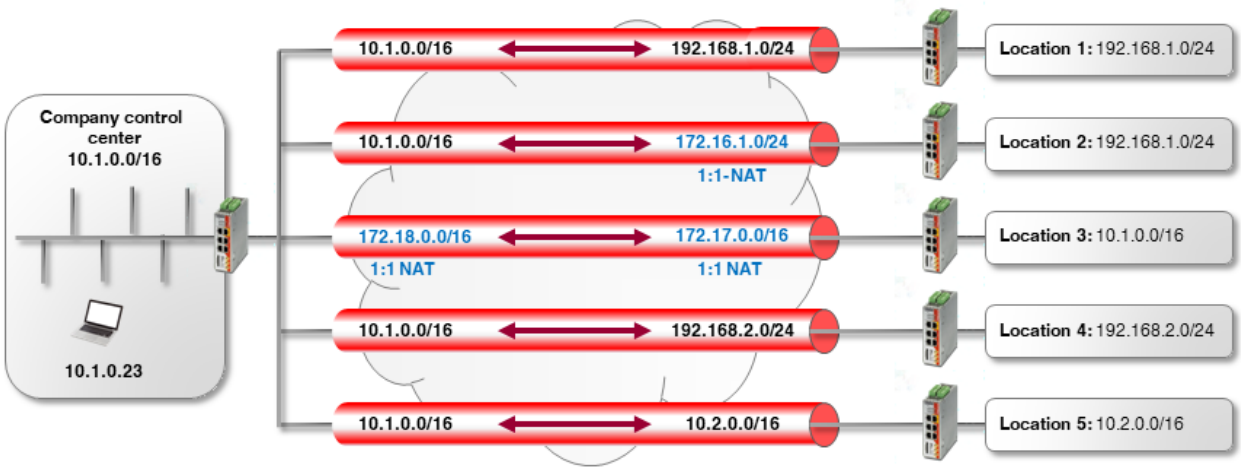


Figure 13-7 **Transmission in both directions (local 1:1 NAT):** the clients (e.g. PLCs) in the external networks can access the control center network via the VPN network, and vice versa.

1. **Location 1**: both locations have different internal networks, which means that the VPN tunnel can be established between networks 10.1.0.0/16 and 192.168.1.0/24.
2. **Location 2**: the internal network of *Location 2* (192.168.1.0/24) is already used for the VPN connection to *Location 1*.

   In order to be able to access the internal network of Location 2 via VPN, 1:1 NAT must be used at the VPN gateway. The VPN tunnel will be established between the Real *network* 10.1.0.0/16 and the *Virtual network* 172.16.1.0/24 (see also Section 13.3).
3. **Location 3**: both networks have the same internal network 10.1.0.0/16.

   In order to establish a VPN connection between the two networks, 1:1 NAT must be used at both VPN gateways. The VPN tunnel will be established between the *Virtual networks* 172.18.0.0/16 and 172.17.0.0/16 (see also Section 13.2).
4. **Locations 4 and 5**: both locations have internal networks that are not used by other VPN connections. Therefore, neither 1:1 NAT nor IP masquerading needs to be used to be able to access the other network in each case.

NOTE: Do not use *Virtual networks* that are already used for other VPN connections.

Configuring VPN connection

The connections are configured along the same lines as Section 13.3.

Advantages

The VPN connections can be used in both directions. The locations can be reached by the control center via the VPN connections, and vice versa.

Disadvantages

Each VPN connection has to be configured separately, depending on which internal network configuration the participating peers use.

Configuration becomes increasingly complex as the number of remote locations increases – which can easily lead to incorrect configurations.

13.5 Using 1:1 NAT for the remote network

The company network is connected to a branch via a VPN connection. The clients (destination systems) in the branch network can be reached via the VPN tunnel.

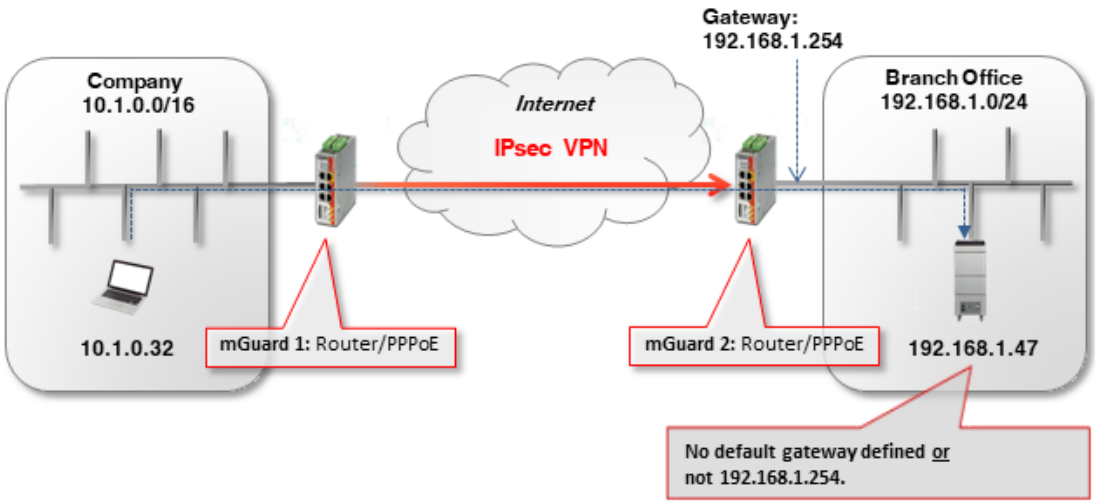
However, there is no default gateway defined on a destination system (e.g. a machine controller which normally only has to be accessed internally), or the defined default gateway is not the mGuard device that makes the VPN tunnel available as the VPN gateway.

The destination system therefore cannot respond to VPN access instances from the company network. If the IP setting of the destination system cannot be changed, the **Remote NAT for IPsec tunnel connections** function can be used to avoid this problem.

13.5.1 Example

The company network (10.1.0.0/16) is not recognized by the destination system (192.168.1.47/24). If the destination (machine controller) receives a packet via VPN tunnel from the company network,

– it will not respond to this at all (if a default gateway has not been defined) or
– it will send the response to its default gateway (and not to the *mGuard 2* VPN gateway).

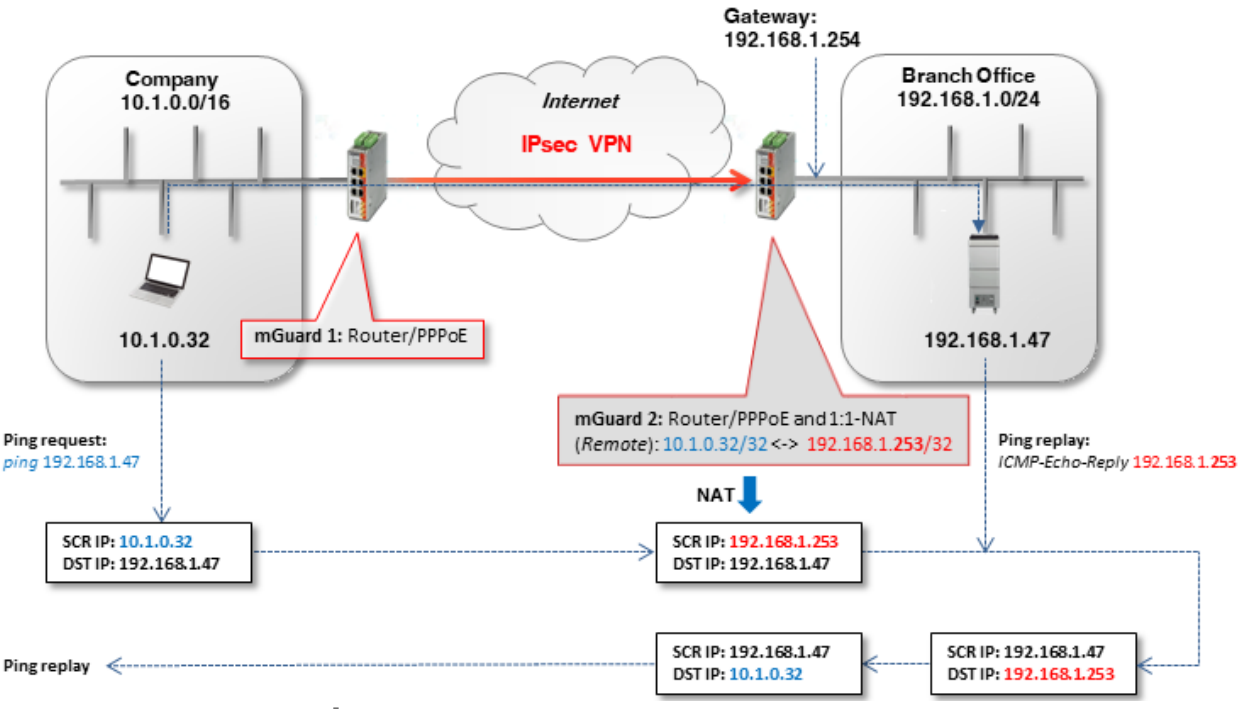


Solution

The **Remote NAT for IPsec tunnel connections** function (*1:1 NAT*) is used in the VPN tunnel settings of *mGuard 2* (branch VPN gateway).

13.5.2 Configuring VPN connection

Remote 1:1 NAT must be used to enable the destination system (e.g. machine controller with IP address 192.168.1.47) to send a response to the "unknown" sender.



Options	
Enabled	☑
Comment	From Company to Branch Office
Type	Tunnel
Local	192.168.1.0/24
Remote	10.1.0.32/32
Local NAT	
Local NAT for IPsec tunnel connections	No NAT
Remote NAT	
Remote NAT for IPsec tunnel connections	1:1 NAT
Network address for remote 1:1 NAT	192.168.1.253

Figure 13-8 **mGuard 2**: IPsec VPN >> General (Tunnel settings with 1:1 NAT)

To configure the VPN connection of the mGuard devices, proceed as follows:

1. Go to **IPsec VPN >> Connections**.
2. Click on the ⊕ icon to add a new VPN connection.
3. Give the connection a unique name and click on the ✎ icon.
4. Under **Transport and tunnel settings**, click on the icon ✎.
5. Configure the VPN connection in accordance with Table 13-4 and Figure 13-8.

Table 13-4 Configuring the VPN connection

Section	Parameter	Company / mGuard 1	Branch / mGuard 2
IPsec VPN >> Connections >> (Edit) >> General			
Options	**A descriptive name for the connection**	VPN to the branch	VPN from the company
	Address of the remote site's VPN gateway	77.245.32.78	%any
	Interface to use for gateway setting %any	-----	External
	Connection startup	Initiate	Wait
Transport and tunnel settings >> (Edit) >> General			
Transport and tunnel settings	**Type**	Tunnel	Tunnel
	Local	**10.1.0.32/32**	192.168.1.0/24
	Remote	192.168.1.0/24	**10.1.0.32/32**
Remote NAT	**Remote NAT for IPsec tunnel connections**	No NAT	1:1 NAT
	Network address for 1:1 NAT in the remote network	-----	192.168.1.**253**

The remote network or the remote IP address is rewritten *(mapped)* to a **free (virtual) IP address** in the internal network of the branch: **10.1.0.32/32 <-> 192.168.1.253**.

A netmask does not need to be specified for the remote network (192.168.1.253). This is automatically adopted by the specified peer network.

The *Virtual network* / IP address must not be used by network clients in the internal network of the branch.

According to the configuration used in the example, only the client 10.1.0.32 in the company network has access to the destination in the branch.

Be careful when selecting the subnet mask for the remote network and specify the network to which the remote network is to be assigned (see "Problem with 1:1 NAT for remote networks").

The ARP proxy of *Guard 2* provides the ARP resolution for the *Virtual network* / IP address. The destination system sends its responses to *mGuard 2*:

– Packets from the company network (10.1.0.0/16) are sent via the VPN gateway (*mGuard 1*) to the real IP address of the destination client in the branch (**192.168.1.47)**.
– *mGuard 2* receives the request, performs a 1:1-NAT for the remote network / IP address (**10.1.0.32/32** <-> **192.168.1.253**) and forwards the request to the destination client (**192.168.1.47**).
– The destination client receives the request and sends its response packet to the virtual sender's IP address (**192.168.1.253**).
– *mGuard 2* receives the response, reverses the 1:1-NAT (**192.168.1.253** <-> **10.1.0.32/32**) and forwards the response to *mGuard 1* or the sender in the company network (**10.1.0.32**).

Problem with 1:1 NAT for remote networks

The subnet mask */24* for the remote network (e.g. 10.1.0.0/24) and a remote 1:1-NAT address (e.g. 192.168.1.0) would not work, because in this case, the ARP proxy of *mGuard 2* would respond to all ARP requests from the internal network of the branch (192.168.1.0 – 192.168.1.255).

Increasing the subnet mask of the remote network would also increase the number of clients in the company network from where the client in the branch can be accessed. However, the number of unused IP addresses in the branch required to assign the source IP address would also increase.

The following table summarizes the relationship between

– the remote subnet mask,
– the clients that can access the destination system,
– the number of necessary unused IP addresses in the internal network.

	Example 1	Example 2	Example 3	Example 4
Specified remote network	10.1.0.0**/26**	10.1.0.64**/26**	10.1.0.128**/28**	10.1.0.32**/32**
Remote IP addresses that can access the destination system	10.1.0.0 – 10.1.0.63	10.1.0.64 – 10.1.0.127	10.1.0.128 – 10.1.0.143	10.1.0.32
Internal network	192.168.1.0**/24**			
Network address for remote 1:1 NAT	192.168.1.128**/26**	192.168.1.192**/26**	192.168.1.240**/28**	192.168.1.253**/32**
Hosts to which the mGuard would respond to ARP requests **(Must not be used in the internal network!)**	192.168.1.128 – 192.168.1.191 **64 hosts**	192.168.1.192 – 192.168.1.255 **64 hosts**	192.168.1.240 – 192.168.1.255 **16 hosts**	192.168.1.253 **1 host**

Additional NAT router

If several clients in the company network are to access the destination system in the branch, a NAT router can be used before the packets are transferred to the VPN tunnel.

For this, the IP address of the NAT router must be specified with subnet mask */32* as the remote network. Only one unused IP address would be necessary.

IP masquerading

If the VPN connection is only to be used in one direction, e.g. from the company network to the branch (remote maintenance), *IP masquerading* can be used on *mGuard 1* in the tunnel instead of an additional NAT router (see also Section 13.4).

In this way, the incoming data packets at *mGuard 2* always have the same source IP address (/32).

14 Connecting networks via hub and spoke (IPsec VPN)

Document ID: 108412_en_00
Document designation: AH EN MGUARD IPSEC VPN HUB SPOKE

Make sure you always use the latest documentation.
This is available to download at phoenixcontact.net/products.

Contents of this document

This document describes the *hub and spoke* function which can be used to connected two or more IPsec VPN tunnels via a central mGuard.

14.1 Introduction

The *hub and spoke* function enables network packets that have been received via a VPN tunnel to be forwarded directly in another VPN tunnel.

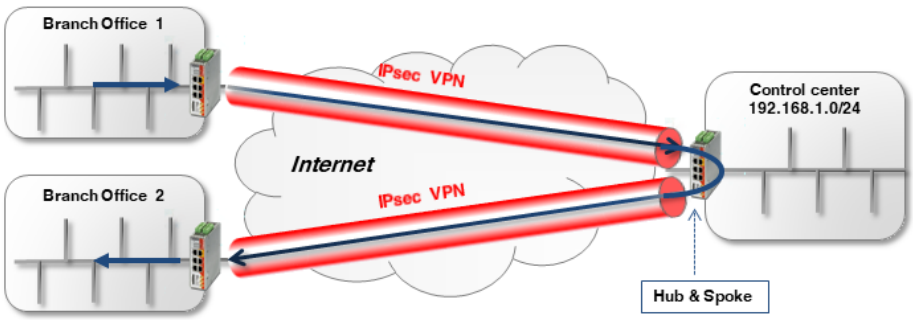
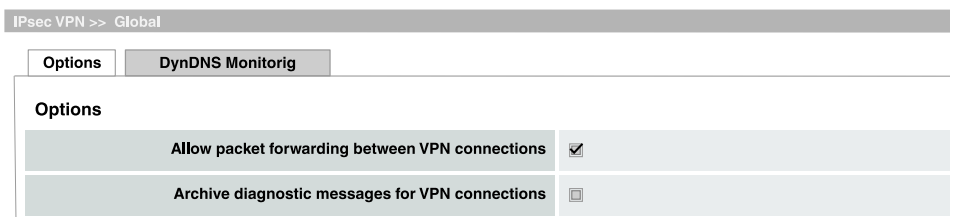


Figure 14-1 *Hub and spoke* via company control center (IPsec VPN)

If several remote locations are connected to the control center and large volumes of data are transmitted, the Internet connection in the control center can become a bottleneck. In such a case, a fully *meshed* network should be used instead of a *hub and spoke* setup.

Along with the activation of *hub and spoke*, the respective networks in the VPN connections must be specified appropriately in order to enable direct routing between the VPN tunnels.



14.2 Connecting branches together via the control center using hub and spoke

Two branches are to communicate with each other via an IPsec VPN connection. The connection is made via the control center, to which both branches have each established a VPN tunnel. The two VPN tunnels are "connected" using the *hub and spoke* function on the mGuard device in the control center (*mGuard 3*).

To enable *routing* from one tunnel to the other, the local network configured in *mGuard 3* must contain all peer networks (e.g. 192.168.0.0/16).

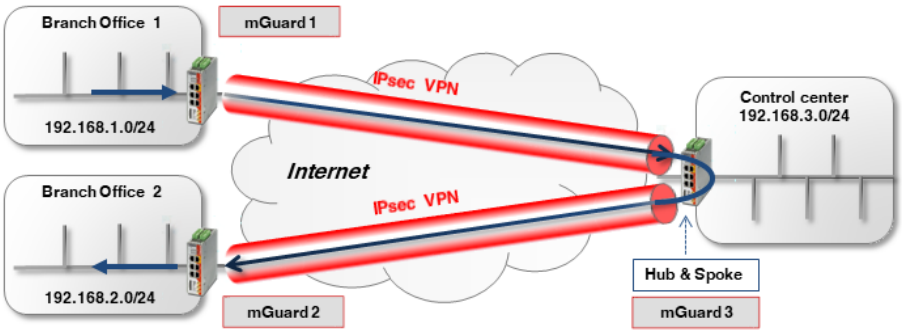


Figure 14-2 *Hub and spoke* via company control center (IPsec VPN)

14.2.1 Configuration

To activate *hub and spoke* on *mGuard 3*, proceed as follows:

1. Log into the web interface of the mGuard device to be configured.
2. Go to **IPsec VPN >> Global** (*Options* tab).
3. Activate the option *Allow packet forwarding between VPN connections*.

The general VPN connection settings are configured under **IPsec VPN >> Connections >> (Edit) >> General** and are described in Section 11 and 12.

The configuration of the respective **transport and tunnel settings** is as follows**:**

mGuard 1 <-> mGuard 3

Seq.	Enabled	Comment	Type	Local	Local NAT	Remote
1	✓	mGuard 1	Tunnel	192.168.1.0/24	No NAT	192.168.0.0/16
2	✓	mGuard 1	Tunnel	192.168.1.0/24	No NAT	10.1.0.0/16

mGuard 2 <-> mGuard 3

Seq.	Enabled	Comment	Type	Local	Local NAT	Remote
1	✓	mGuard 2	Tunnel	192.168.2.0/24	No NAT	192.168.0.0/16
1	✓	mGuard 3	Tunnel	192.168.0.0/16	No NAT	192.168.2.0/24

Hub and spoke, if the local network does not contain all peer networks

What happens if the control center network is not a part of the network **192.168.0.0/16**, but is a part of, e.g. **10.1.0.0/16**?

In this case, the two branches can communicate with each other via the VPN tunnel. However, neither **branch 1** nor **branch 2** have access to the **control center** network, and vice versa.

This problem can be resolved by specifying a second VPN tunnel in each of the configured VPN tunnels which addresses the control center network (see following example for connecting *mGuard 1* to *mGuard 3*).

mGuard 1 <-> mGuard 3

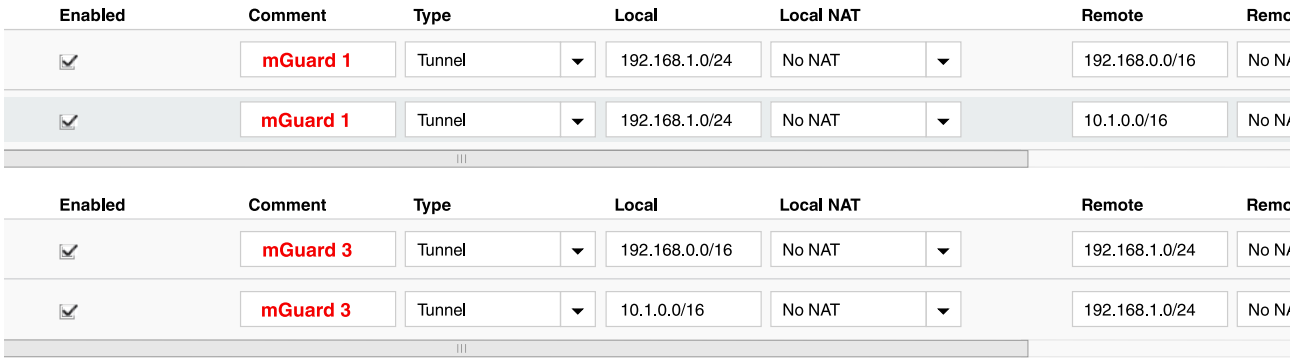


Table 14-1 shows the transport and tunnel settings for all devices (*mGuard 1, 2,* and *3*) in this case:

mGuard 1 <-> mGuard 3 | mGuard 2 <-> mGuard 3

Table 14-1 Transport and tunnel settings with *hub and spoke* (different networks)

VPN connection	Tunnel Settings	Local	Remote
mGuard 1 <---> mGuard 3	mGuard 1	192.168.**1**.0/24	192.168.0.0/16
		192.168.**1**.0/24	10.1.0.0/16
	mGuard 3	192.168.0.0/16	192.168.**1**.0/24
		10.1.0.0/16	192.168.**1**.0/24
mGuard 2 <---> mGuard 3	mGuard 2	192.168.**2**.0/24	192.168.0.0/16
		192.168.**2**.0/24	10.1.0.0/16
	mGuard 3	192.168.0.0/24	192.168.**2**.0/24
		10.1.0.0/16	192.168.**2**.0/24

14.3 Connecting external technicians to production locations via hub and spoke

Two remote maintenance technicians are to be able to access the machines in all production locations (branches) from their laptops via a VPN connection (via VPN Client software or mGuard device). Initially, the VPN connection is via a central mGuard (*mGuard 4*), which establishes a VPN connection with the machine network of the respective production location via *hub and spoke*.

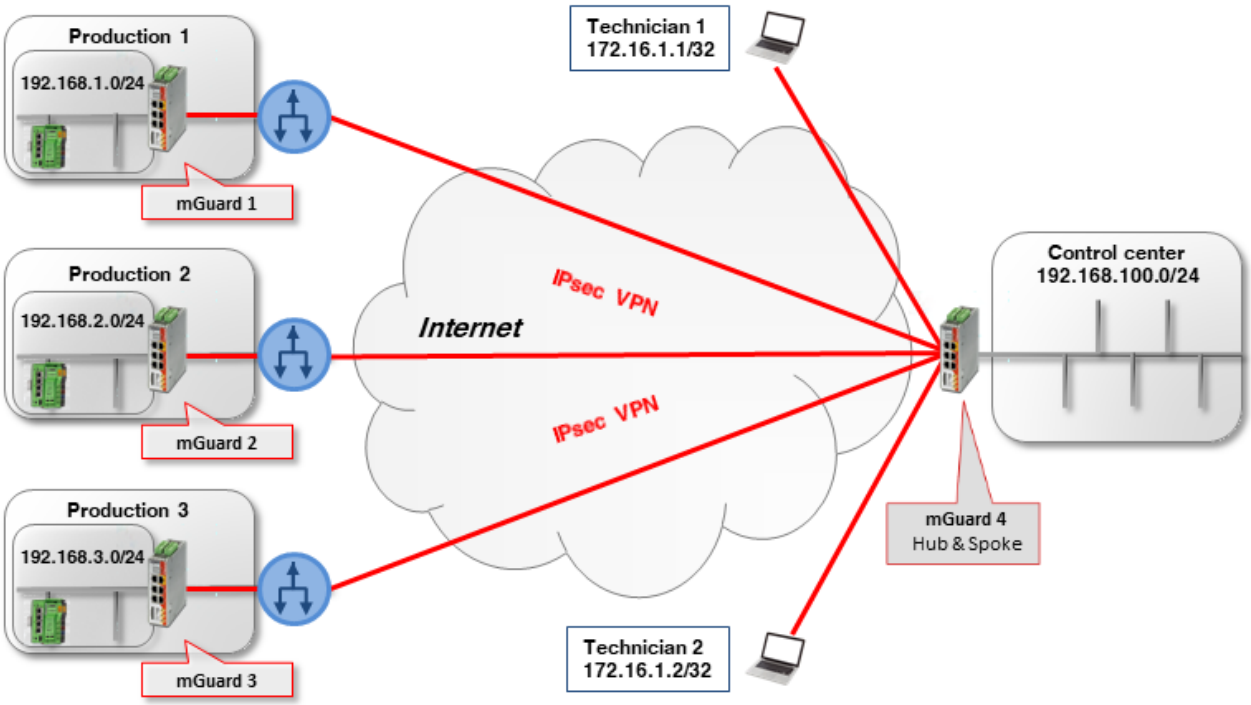


Figure 14-3 Remote maintenance via *hub and spoke* via the company control center (IPsec VPN)

An mGuard device is installed as a router in each of the production locations in order to connect the machine network with the branch network, and to establish a VPN connection to the mGuard device in the company control center.

The technicians use *virtual IP addresses* on their laptops so they are not dependent on the real IP addresses assigned to the laptops:

– Technician 1: 172.16.1.**1**/32
– Technician 2: 172.16.1.**2**/32.

In order access all production locations, the VPN network of the peer specified in each case must contain the machine networks of all three locations (192.168.**1**.0/24, 192.168.**2**.0/24 und 192.168.**3**.0/24): i.e. **192.168.0.0/16**.

The mGuard devices of the branches use the internal networks 192.168.**1**.0/24, 192.168.**2**.0/24 and 192.168.**3**.0/24. Data packets that are sent via the VPN connection from the technicians' laptops to the mGuard devices have one of the two sender IP addresses: 172.16.1.**1**/32 or 172.16.1.**2**/32.

If remote maintenance is not just to be limited to two technicians, a peer VPN network must be specified at the production location mGuard devices via which in principle several technicians can be connected: in this example 172.16.1.0/24.

Example: Access via hub and spoke by two technicians

If the *hub and spoke* function is enabled on the mGuard device in the control center (*mGuard 4*), the tunnel settings for the VPN connections must be configured as follows – taking into consideration the above listed points – (see also the example configuration in Section 14.2.1):

Table 14-2 *Hub and spoke*: Transport and tunnel settings with **different** local networks

VPN connection	Client	Local	<-->	Remote
Technician 1 <–> mGuard 4	Technician 1	172.16.1.1/32	<-->	192.168.0.0/16
	mGuard 4	192.168.0.0/16	<-->	172.16.1.1/32
Technician 2 <–> mGuard 4	Technician 2	172.16.1.2/32	<-->	192.168.0.0/16
	mGuard 4	192.168.0.0/16	<-->	172.16.1.2/32
mGuard 1 <–> mGuard 4	mGuard 1	192.168.1.0/24	<-->	172.16.1.0/24
	mGuard 4	172.16.1.0/24	<-->	192.168.1.0/24
mGuard 2 <–> mGuard 4	mGuard 2	192.168.2.0/24	<-->	172.16.1.0/24
	mGuard 4	172.16.1.0/24	<-->	192.168.2.0/24
mGuard 3 <–> mGuard 4	mGuard 3	192.168.3.0/24	<-->	172.16.1.0/24
	mGuard 4	172.16.1.0/24	<-->	192.168.3.0/24

Example: Access with the same networks in the production locations

What happens when the mGuard devices at the production locations all use the same internal network (e.g. 192.168.**1**.0/24)?

In this case, the mGuard devices in the branches must use *Local 1:1 NAT for IPsec tunnel connections* for the local network (see also Section 13.3, “Connecting locations with the same internal networks to a control center (1:1 NAT)”).

The individual production locations are then accessed via a *Virtual network* and the mGuard device performs a local 1:1 NAT from the *Virtual network* to the local *Real network* (192.168.1.0/24).

In this example, the following *Virtual networks* are used for the production locations:

– Branch 1:172.17.**1**.0/24
– Branch 2: 172.17.**2**.0/24
– Branch 3: 172.17.**3**.0/24.

The technicians must use these virtual networks to access the respective machine. Therefore, the technicians must specify 172.17.0.0/16 as the peer VPN network.

The tunnel settings for this setup are as follows (see Table 14-3 and Figure 14-4).

Table 14-3 *Hub and spoke*: Tunnel settings with the **same** local networks (with local 1:1 NAT)

VPN connection	Client	Local	<-->	Remote
Technician 1 <–> mGuard 4	Technician 1	172.16.1.1/32	<–>	172.17.0.0/16
	mGuard 4	172.17.0.0/16	<–>	172.16.1.1/32
Technician 2 <–> mGuard 4	Technician 2	172.16.1.2/32	<–>	172.17.0.0/16
	mGuard 4	172.17.0.0/16	<–>	172.16.1.2/32
mGuard 1 <–> mGuard 4	mGuard 1	172.17.**1**.0/24	<–>	172.16.1.0/24
		Local 1:1 NAT to 192.168.1.0/24		
	mGuard 4	172.16.1.0/24	<–>	172.17.**1**.0/24
mGuard 2 <–> mGuard 4	mGuard 2	172.17.**2**.0/24	<–>	172.16.1.0/24
		Local 1:1 NAT to 192.168.1.0/24		
	mGuard 4	172.16.1.0/24	<–>	172.17.**2**.0/24
mGuard 3 <–> mGuard 4	mGuard 3	172.17.**3**.0/24	<–>	172.16.1.0/24
		Local 1:1 NAT to 192.168.1.0/24		
	mGuard 4	172.16.1.0/24	<–>	172.17.**3**.0/24

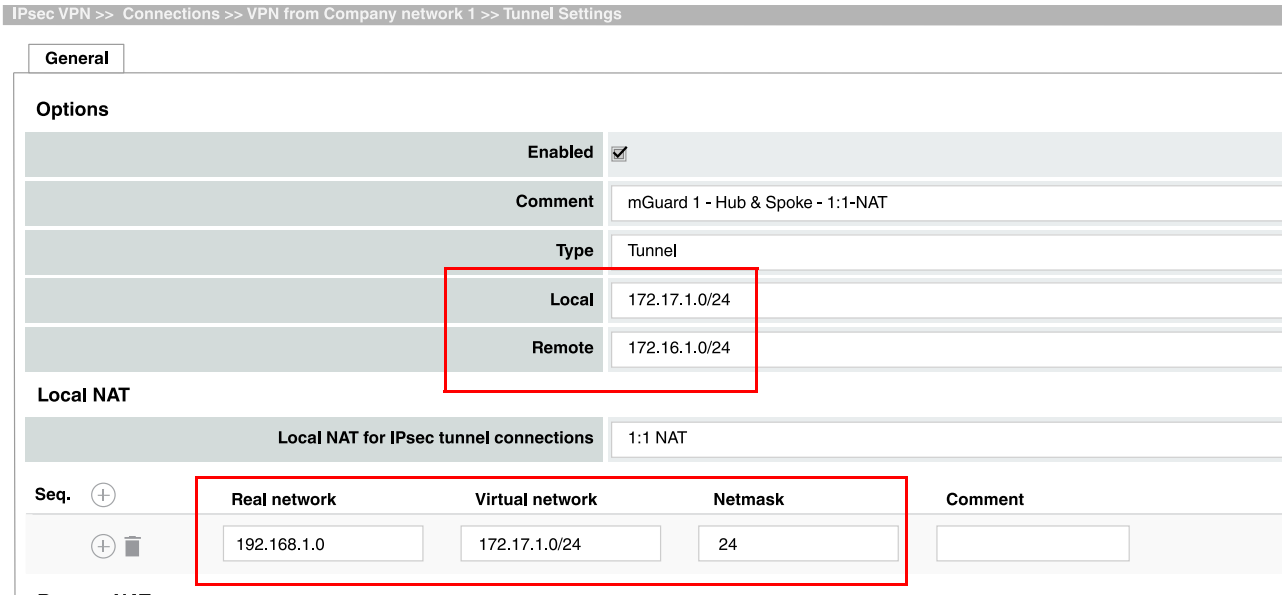


Figure 14-4 *Hub and spoke*: Example *mGuard 1* tunnel settings + local 1:1 NAT

15 VPN Troubleshooting

Document-ID: 108417_en_00
Document-Description: AH EN MGUARD VPN TROUBLESHOOTING

Make sure you always use the latest documentation.
It can be downloaded using the following link phoenixcontact.net/products.

Contents of this document

This document should help to narrow down problems related to VPN connections. The log examples were taken from mGuard devices running firmware version 7.6.

15.1 Introduction

A VPN connection is established in two phases:

1. **Phase I**: In *phase I* (**ISAKMP SA**, SA = *Security Association*) the VPN peers authenticate each other and an encryption key to protect *phase II* is securely negotiated. This SA is a connection between the two VPN peers only and is used to exchange new keys and DPD messages (DPD = *Dead Peer Detection*).
2. **Phase II**: VPN peers only proceed with *phase II* (**IPsec SA**) if *phase I* was established successfully. In *phase II* IPsec connection parameters are negotiated. This SA connects the two networks and is used for the data exchange between the clients of those networks.

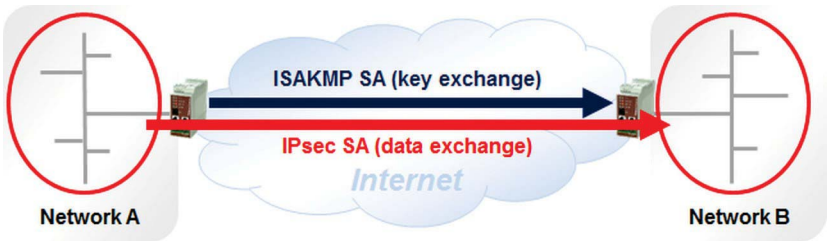


Figure 15-1 Establishment of the two phases of the VPN connection (*ISAKMP SA* and *IPSEC SA*)

Most frequently establishing a VPN connection fails during *phase I* (*ISAKMP SA*), caused either by a wrong configuration of the VPN connection or by routers in-between the two VPN peers. If the establishment fails during *phase II* (*IPsec SA*), it is caused by a configuration mismatch.

If the establishment of a VPN connection fails, inspect at first the *IPsec Status* (menu **IPsec VPN >> IPsec Status**) to get the information, at which stage the failure happens. In the screenshot below, the VPN connection was established successfully.

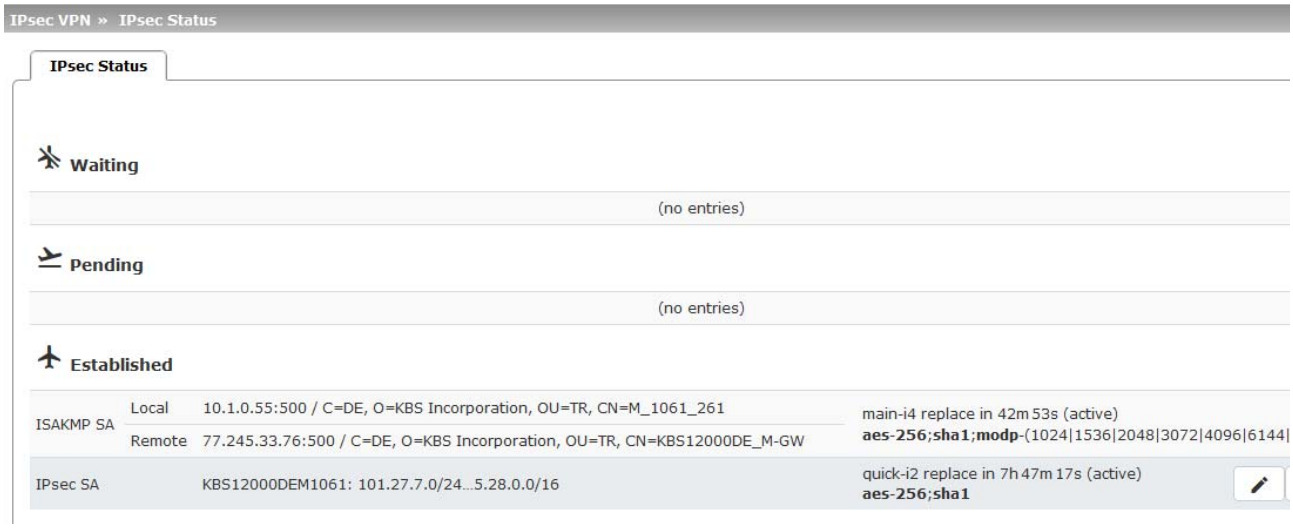


Figure 15-2 IPsec status – VPN connection successfully established

15.1.1 The following situations may occur

Table 15-1 The following situations may occur

Situation that may occur	Refer to chapter
VPN connection not displayed in the "IPsec Status" at all	Section 15.2, “VPN connection not displayed in the IPsec Status”
ISAKMP SA not established ("ISAKMP State" empty)	Section 15.3, “ISAKMP SA (Phase I) can not be established”
IPsec SA not established ("IPsec State" empty)	Section 15.4, “IPSec SA (phase II) can not be established”
Problem transferring data through an established VPN connection ("ISAKMP SA" and "IPsec SA" established)	Section 15.5, “Remote network clients can not be reached through established VPN tunnel”

In the following chapters **Initiator** stands for the mGuard device which initiates the VPN connection, **Responder** for the mGuard device which waits for the VPN connection.

If the establishment of the *ISAKMP SA* or *IPsec SA* fails (2 and 3), in most cases the VPN logs of both VPN peers need to be inspected for being able to locate the reason for the failure.

Request a support snapshot (menu **Support >> Advanced >> Snapshot**) of **both** VPN peers form the customer.

15.2 VPN connection not displayed in the IPsec Status

If the VPN connection does not appear in the *IPsec Status*, it may be caused by the following reasons:

15.2.1 VPN connection not enabled

Disabled VPN connections do not appear in the *IPsec Status*.

- Ensure the VPN connection is enabled (menu **IPsec VPN >> Connections**).
- If the VPN connection is triggered by CMD contact, ensure the button or On/Off switch was pressed to activate the VPN connection.
- If the VPN connection is triggered by calling the script *nph-vpn.cgi*, ensure the according command was called to activate the VPN connection.

15.2.2 Option "Disable VPN until the user is authenticated via HTTP" is enabled

- Ensure the option *Disable VPN until the user is authenticated via HTTP* is not enabled in the menu **Authentication >> Administrative Users**.

If this option is enabled, the user will be prompted to enter the user's password when trying to access any web side after a reboot of the mGuard device. The configured VPN connection will only be added to the VPN service if the entered password is correct. This option was implemented to protect mGuard devices with a configured VPN connection to the headquarters used by road warriors.

15.2.3 Wrong configuration

The problem may also be caused by a wrong configuration.

- Apply a minor change to the VPN configuration, click the icon <*Save*> and inspect the System Message.
- If the System Message does not report any problem, inspect the VPN logs (menu **Logging >> Browse Local Logs***)* for any error messages, as for example:

```
firestarter: vpnd: whack error: "MAI1825301978_1" ikelifetime [3600] must be greater than
rekeymargin*(100+rekeyfuzz)/100 [5400*(100+100)/100 = 10800]

firestarter: tunnel ignored: local address '10.1.80.100' within remote network '10.0.0.0/8'
```

15.2.4 General network problems

The problem may also be caused by some general network problems.

- The mGuard device (*Router mode*) is configured to receive its external IP settings from a DHCP server but did not receive them yet.
- A DNS name is specified as "*Address of the remote site's VPN gateway*" in the VPN connection but the mGuard device could not resolve the DNS name due to problems with the DNS resolution.

15.3 ISAKMP SA (Phase I) can not be established

The *ISAKMP SA* is established using the *Main Mode* provided by the *Internet Key Exchange* (IKE) protocol. IKE also provides the *Aggressive Mode* but this mode less unsecure and only supported by newer mGuard firmware.

In *Main Mode*, three pairs of messages are exchanged between both VPN peers. Keep the following diagram in mind when narrowing down a problem. It helps a lot understanding what could cause the problem.

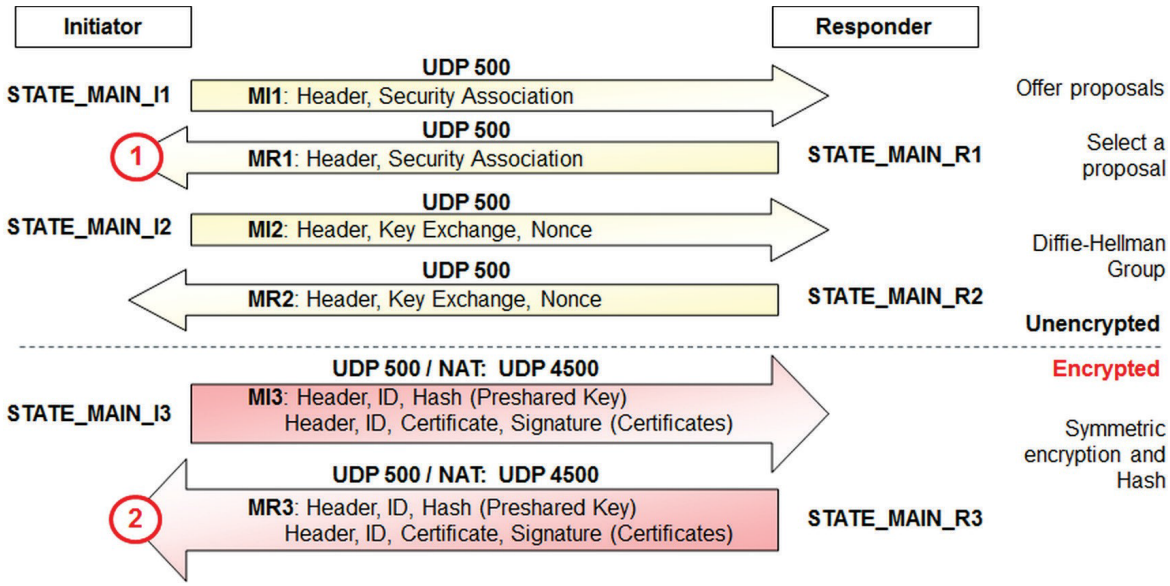


Figure 15-3 *ISAKMP SA - Phase* I

Every time when the **initiator** has sent out a message, its state changes from STATE_MAIN_I1 to STATE_MAIN_I2 and STATE_MAIN_I3, the **responder's** state from STATE_MAIN_R1 to STATE_MAIN_R2 and STATE_MAIN_R3 respectively. The state changes are reflected in the logs. The VPN connection is established through UDP port 500. If the connection is established across one or more gateways that have NAT activated, starting with the third *Main Mode* message MI3 the exchange happens through UDP port 4500.

Problems usually occur at the above marked points ① and ②:

① The **initiator** does not receive a response from the **responder**.

② The **initiator** receives an unexpected packet or an error message from the **responder**.

15.3.1 Log example of a successfully established ISAKMP SA

Initiator | **Responder**

Initiator Log:

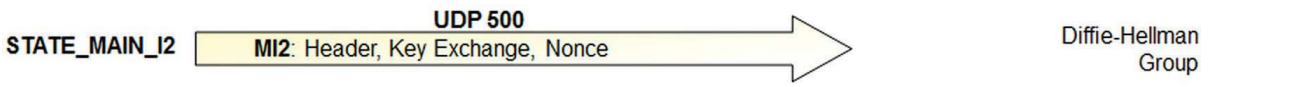
```
08:53:47.90161 "MAI1950251842_1" #2: initiating Main Mode
```

Responder Log:

```
08:53:47.90165 packet from 77.245.32.68:500: received Vendor ID payload [Openswan (this version) 2.6.24 ]
08:53:47.90186 packet from 77.245.32.68:500: received Vendor ID payload [Dead Peer Detection]
08:53:47.90194 packet from 77.245.32.68:500: received Vendor ID payload [RFC 3947] method set to=109
08:53:47.90202 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-03] meth=108, but already using method 109
08:53:47.90210 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02_n] meth=106, but already using method 109
08:53:47.90218 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02] meth=107, but already using method 109
08:53:47.90226 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-00]
08:53:47.90279 packet from 77.245.32.68:500: received Vendor ID payload [Innominate IKE Fragmentation]
08:53:47.90297 packet from 77.245.32.68:500: received Vendor ID payload [Innominate always send NAT-OA]
08:53:47.90305 "MAI0874627901_1"[1] 77.245.32.68 #2: responding to Main Mode from unknown peer 77.245.32.68
08:53:47.90333 "MAI0874627901_1"[1] 77.245.32.68 #2: enabling Innominate IKE Fragmentation (main_inI1_outR1)
08:53:47.90344 "MAI0874627901_1"[1] 77.245.32.68 #2: enabling Innominate Always Send NAT-OA (main_inI1_outR1)
08:53:47.90369 "MAI0874627901_1"[1] 77.245.32.68 #2: transition from state STATE_MAIN_R0 to state STATE_MAIN_R1
08:53:47.90384 "MAI0874627901_1"[1] 77.245.32.68 #2: STATE_MAIN_R1: sent MR1, expecting MI2
```

Initiator Log:

```
08:53:48.15255 "MAI1950251842_1" #2: received Vendor ID payload [Openswan (this version) 2.6.24 ]
08:53:48.15259 "MAI1950251842_1" #2: received Vendor ID payload [Dead Peer Detection]
08:53:48.15263 "MAI1950251842_1" #2: received Vendor ID payload [RFC 3947] method set to=109
08:53:48.15267 "MAI1950251842_1" #2: received Vendor ID payload [Innominate IKE Fragmentation]
08:53:48.15271 "MAI1950251842_1" #2: received Vendor ID payload [Innominate always send NAT-OA]
08:53:48.15275 "MAI1950251842_1" #2: enabling possible NAT-traversal with method 4
08:53:48.15279 "MAI1950251842_1" #2: enabling Innominate IKE Fragmentation (main_inR1_outI2)
08:53:48.15296 "MAI1950251842_1" #2: enabling Innominate Always Send NAT-OA (main_inR1_outI2)
08:53:48.37178 "MAI1950251842_1" #2: transition from state STATE_MAIN_I1 to state STATE_MAIN_I2
08:53:48.37186 "MAI1950251842_1" #2: STATE_MAIN_I2: sent MI2, expecting MR2
```



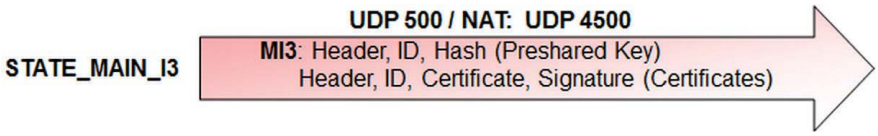
Responder Log:

```
08:53:48.52717 "MAI0874627901_1"[1] 77.245.32.68 #2: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): both are NATed
08:53:50.24004 "MAI0874627901_1"[1] 77.245.32.68 #2: transition from state STATE_MAIN_R1 to state STATE_MAIN_R2
08:53:50.24027 "MAI0874627901_1"[1] 77.245.32.68 #2: STATE_MAIN_R2: sent MR2, expecting MI3
```

Initiator **Responder**

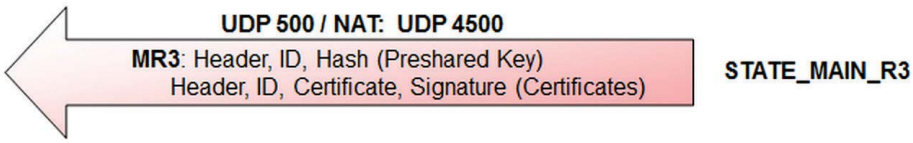
Initiator Log:

```
08:53:50.72881 "MAI1950251842_1" #2: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): both are NATed
08:53:50.72892 "MAI1950251842_1" #2: I am sending my cert
08:53:50.72896 "MAI1950251842_1" #2: I am sending a certificate request
08:53:50.72942 "MAI1950251842_1" #2: transition from state STATE_MAIN_I2 to state STATE_MAIN_I3
08:53:50.72961 "MAI1950251842_1" #2: STATE_MAIN_I3: sent MI3, expecting MR3
```



Responder Log:

```
08:53:50.76811 "MAI0874627901_1"[1] 77.245.32.68 #2: Main mode peer ID is ID_DER_ASN1_DN: 'O=Innominate, OU=Support, CN=mGuard 1'
08:53:50.76831 "MAI0874627901_1"[1] 77.245.32.68 #2: issuer cacert not found
08:53:50.76839 "MAI0874627901_1"[1] 77.245.32.68 #2: X.509 certificate rejected
08:53:50.76846 "MAI0874627901_1"[1] 77.245.32.68 #2: I am sending my cert
08:53:50.76887 "MAI0874627901_1"[1] 77.245.32.68 #2: transition from state STATE_MAIN_R2 to state STATE_MAIN_R3
08:53:50.76905 "MAI0874627901_1"[1] 77.245.32.68 #2: new NAT mapping for #2, was 77.245.32.68:500, now 77.245.32.68:4500
08:53:50.76914 "MAI0874627901_1"[1] 77.245.32.68 #2: new NAT mapping for #1, was 77.245.32.68:500, now 77.245.32.68:4500
08:53:50.76922 "MAI0874627901_1"[1] 77.245.32.68 #2: STATE_MAIN_R3: sent MR3, ISAKMP SA established {auth=OAKLEY_RSA_SIG
                                                     cipher=oakley_3des_cbc_192 prf=oakley_md5 group=modp8192}
08:53:50.76932 "MAI0874627901_1"[1] 77.245.32.68 #2: Dead Peer Detection (RFC 3706): enabled
```



Initiator Log:

```
08:53:50.97225 "MAI1950251842_1" #2: Main mode peer ID is ID_DER_ASN1_DN: 'O=Innominate, OU=Support, CN=mGuard 2'
08:53:50.97229 "MAI1950251842_1" #2: issuer cacert not found
08:53:50.97233 "MAI1950251842_1" #2: X.509 certificate rejected
08:53:50.97236 "MAI1950251842_1" #2: transition from state STATE_MAIN_I3 to state STATE_MAIN_I4
08:53:50.97244 "MAI1950251842_1" #2: STATE_MAIN_I4: ISAKMP SA established {auth=OAKLEY_RSA_SIG cipher=oakley_3des_cbc_192
                                     prf=oakley_md5 group=modp8192}
```

The log entries **issuer cacert not found** and **X.509 certificate rejected** do not indicate that there is a problem.

the mGuard device tries CA authentication first before identifying the remote side by its certificate stored in the VPN connection. If there is no CA certificate present or if there is no matching CA certificate, the above mentioned log entries appear and the mGuard device continues identifying the remote side by its certificate.

15.3.2 Initiator: “pending Quick Mode with w.x.y.z took too long – replacing phase 1”

Initiator Log:

```
08:56:40.12570 "MAI1950251842_1" #6: initiating Main Mode
09:02:50.03792 pending Quick Mode with 77.245.33.66 "MAI1950251842_1" took too long -- replacing phase 1
09:02:50.03804 "MAI1950251842_1" #7: initiating Main Mode to replace #6
09:04:50.04538 pending Quick Mode with 77.245.33.66 "MAI1950251842_1" took too long -- replacing phase 1
09:04:50.04550 "MAI1950251842_1" #8: initiating Main Mode to replace #7
```

The mGuard device initiates the VPN connection by sending the first *Main Mode* message (MI1) but there is no response from the **responder**. The mGuard device keeps on initiating the VPN connection.

Now it is important to inspect the VPN logs of the **responder** to determine whether this message has reached the **responder** or not.

15.3.2.1 Resp.: No received Packet registered in the VPN Logs of the Responder

Responder Log:

No entries for a new VPN connect request appear in the logs. At least **packet from w.x.y.z: received Vendor ID payload** should appear in the logs if the responder has received the first *Main Mode* message. If such a log entry does not appear, the first *Main Mode* message of the initiator did not reach the responder.

Possible reasons:

- The specified IP address or DNS name of the **responder** is incorrect (menu **IPsec VPN >> Connections >> (Edit) >> General**, parameter *Address of the remote site's VPN gateway*).
- If the **initiator** is located behind a firewall, this firewall may block outgoing traffic to UDP port 500.
- If the **responder** is located behind a NAT router, either port forwarding for incoming traffic on UDP port 500 to the IP address of the **responder** is not configured on the NAT router or it is not configured properly.
- The **responder** does not listen for incoming VPN connections (e.g. no VPN connections configured or all VPN connections disabled).

Check on the **initiator** with the Tool *IKE Ping* (menu **Support >> Tools >> IKE Ping**) if the IP address or DNS name of the **responder** is reachable.

15.3.2.2 Responder: “initial Main Mode message received on w.x.y.z:500 but no connection has been authorized

Responder Log:

```
09:07:35.94714 packet from 77.245.32.68:500: received Vendor ID payload [Openswan (this version) 2.6.24 ]
09:07:35.94748 packet from 77.245.32.68:500: received Vendor ID payload [Dead Peer Detection]
09:07:35.94757 packet from 77.245.32.68:500: received Vendor ID payload [RFC 3947] method set to=109
09:07:35.94764 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-03] meth=108, but already using method 109
09:07:35.94772 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02_n] meth=106, but already using method 109
09:07:35.94780 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02] meth=107, but already using method 109
09:07:35.94789 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-00]
09:07:35.94796 packet from 77.245.32.68:500: received Vendor ID payload [Innominate IKE Fragmentation]
09:07:35.94803 packet from 77.245.32.68:500: received Vendor ID payload [Innominate always send NAT-OA]
09:07:35.94811 packet from 77.245.32.68:500: initial Main Mode message received on 192.168.3.1:500 but no connection has been authorized
                                             with policy=RSASIG
```

The **responder** has received the first *Main Mode* message from the **initiator**. The **initiator** informs the **responder**, among other things, about the encryption and hash algorithm (e.g. AES-256/SHA-1) that shall be used for the establishment of the *ISAKMP SA*. The **responder** checks if there is any VPN connection configured which also supports these algorithms. If there is no accordance, the above mentioned message appears in the logs. In this case the **responder** does not send a reply to the **initiator**.

Reason:

Mismatch of the specified encryption and/or hash algorithms for the *ISAKMP SA*. Check the specified encryption and hash algorithms for the *ISAKMP SA* on the **initiator** and on the **responder** (menu **IPsec VPN >> Connections >> (Edit) >> IKE Options**, section *ISAKMP SA* (*Key Exchange*). Both VPN connections need to support the same encryption and hash algorithm.

15.3.3 Initiator: “Possible authentication failure: no acceptable response to our first encrypted message”

Initiator Log:

```
09:54:06.14104 "MAI1950251842_1" #55: initiating Main Mode
09:54:08.02489 "MAI1950251842_1" #55: received Vendor ID payload [Openswan (this version) 2.6.24 ]
09:54:08.02493 "MAI1950251842_1" #55: received Vendor ID payload [Dead Peer Detection]
09:54:08.02497 "MAI1950251842_1" #55: received Vendor ID payload [RFC 3947] method set to=109
09:54:08.02501 "MAI1950251842_1" #55: received Vendor ID payload [Innominate IKE Fragmentation]
09:54:08.02505 "MAI1950251842_1" #55: received Vendor ID payload [Innominate always send NAT-OA]
09:54:08.02509 "MAI1950251842_1" #55: enabling possible NAT-traversal with method 4
09:54:08.02513 "MAI1950251842_1" #55: enabling Innominate IKE Fragmentation (main_inR1_outI2)
09:54:08.02528 "MAI1950251842_1" #55: enabling Innominate Always Send NAT-OA (main_inR1_outI2)
09:54:08.35894 "MAI1950251842_1" #55: transition from state STATE_MAIN_I1 to state STATE_MAIN_I2
09:54:08.35902 "MAI1950251842_1" #55: STATE_MAIN_I2: sent MI2, expecting MR2
09:54:10.71933 "MAI1950251842_1" #55: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): both are NATed
09:54:10.71945 "MAI1950251842_1" #55: I am sending my cert
09:54:10.71948 "MAI1950251842_1" #55: I am sending a certificate request
09:54:10.72057 "MAI1950251842_1" #55: transition from state STATE_MAIN_I2 to state STATE_MAIN_I3
09:54:10.72076 "MAI1950251842_1" #55: STATE_MAIN_I3: sent MI3, expecting MR3
09:54:20.23466 "MAI1950251842_1" #55: discarding duplicate packet; already STATE_MAIN_I3
09:54:40.23282 "MAI1950251842_1" #55: discarding duplicate packet; already STATE_MAIN_I3
09:55:21.23123 "MAI1950251842_1" #55: max number of retransmissions (2) reached STATE_MAIN_I3. Possible authentication failure:
                                      no acceptable response to our first encrypted message
```

The **initiator** has sent his third *Main Mode* message (MI3) and expects now the response from the **responder** (MR3). But he has received MR2 again from the **responder**. Thus he exclaims “*discarding duplicate packet; already STATE_MAIN_I3*”.

If the VPN connection is established across one or more gateways that have NAT activated, starting with the third *Main Mode* message (MI3) the exchange happens through UDP port 4500 instead of UDP port 500 due to NAT-Traversal.

The log of the **responder** will tell us more about the reason.

Responder Log:

```
09:54:07.89904 packet from 77.245.32.68:500: received Vendor ID payload [Openswan (this version) 2.6.24 ]
09:54:07.89913 packet from 77.245.32.68:500: received Vendor ID payload [Dead Peer Detection]
09:54:07.89921 packet from 77.245.32.68:500: received Vendor ID payload [RFC 3947] method set to=109
09:54:07.89928 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-03] meth=108, but already using method 109
09:54:07.89936 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02_n] meth=106, but already using method 109
09:54:07.89989 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02] meth=107, but already using method 109
09:54:07.90049 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-00]
09:54:07.90061 packet from 77.245.32.68:500: received Vendor ID payload [Innominate IKE Fragmentation]
09:54:07.90089 packet from 77.245.32.68:500: received Vendor ID payload [Innominate always send NAT-OA]
09:54:07.90100 "MAI0874627901_1"[1] 77.245.32.68 #67: responding to Main Mode from unknown peer 77.245.32.68
09:54:07.90108 "MAI0874627901_1"[1] 77.245.32.68 #67: enabling Innominate IKE Fragmentation (main_inI1_outR1)
09:54:07.90117 "MAI0874627901_1"[1] 77.245.32.68 #67: enabling Innominate Always Send NAT-OA (main_inI1_outR1)
09:54:07.90142 "MAI0874627901_1"[1] 77.245.32.68 #67: transition from state STATE_MAIN_R0 to state STATE_MAIN_R1
09:54:07.90171 "MAI0874627901_1"[1] 77.245.32.68 #67: STATE_MAIN_R1: sent MR1, expecting MI2
09:54:08.55076 "MAI0874627901_1"[1] 77.245.32.68 #67: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): both are NATed
09:54:10.24331 "MAI0874627901_1"[1] 77.245.32.68 #67: transition from state STATE_MAIN_R1 to state STATE_MAIN_R2
09:54:10.24355 "MAI0874627901_1"[1] 77.245.32.68 #67: STATE_MAIN_R2: sent MR2, expecting MI3
09:55:18.23344 "MAI0874627901_1"[1] 77.245.32.68 #66: max number of retransmissions (2) reached STATE_MAIN_R2
09:55:20.23351 "MAI0874627901_1"[1] 77.245.32.68 #67: max number of retransmissions (2) reached STATE_MAIN_R2
```

The **responder** is in STATE_MAIN_R2 and is expecting the third *Main Mode* message (MI3) from the **initiator** but did not receive it. Thus the **responder** keeps on retransmitting MR2.

Reason:

– Some entity in-between the two VPN peers blocks UDP traffic directed to port 4500.
– If the **initiator** is located behind a firewall, most likely this firewall drops outgoing traffic to UDP port 4500.
– If the **responder** is located behind a NAT router, either port forwarding for UDP 4500 to the IP address of the **responder** is not configured on the NAT router or it is not configure properly.

15.3.4 Initiator: “ignoring informational payload, type INVALID_ID_INFORMATION”

Initiator Log:

```
10:00:07.10837 "MAI1950251842_1" #61: initiating Main Mode
10:00:09.02070 "MAI1950251842_1" #61: received Vendor ID payload [Openswan (this version) 2.6.24 ]
10:00:09.02074 "MAI1950251842_1" #61: received Vendor ID payload [Dead Peer Detection]
10:00:09.02077 "MAI1950251842_1" #61: received Vendor ID payload [RFC 3947] method set to=109
10:00:09.02081 "MAI1950251842_1" #61: received Vendor ID payload [Innominate IKE Fragmentation]
10:00:09.02085 "MAI1950251842_1" #61: received Vendor ID payload [Innominate always send NAT-OA]
10:00:09.02089 "MAI1950251842_1" #61: enabling possible NAT-traversal with method 4
10:00:09.02093 "MAI1950251842_1" #61: enabling Innominate IKE Fragmentation (main_inR1_outI2)
10:00:09.02108 "MAI1950251842_1" #61: enabling Innominate Always Send NAT-OA (main_inR1_outI2)
10:00:09.34262 "MAI1950251842_1" #61: transition from state STATE_MAIN_I1 to state STATE_MAIN_I2
10:00:09.34270 "MAI1950251842_1" #61: STATE_MAIN_I2: sent MI2, expecting MR2
10:00:11.70805 "MAI1950251842_1" #61: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): both are NATed
10:00:11.70817 "MAI1950251842_1" #61: I am sending my cert
10:00:11.70821 "MAI1950251842_1" #61: I am sending a certificate request
10:00:11.70929 "MAI1950251842_1" #61: transition from state STATE_MAIN_I2 to state STATE_MAIN_I3
10:00:11.70948 "MAI1950251842_1" #61: STATE_MAIN_I3: sent MI3, expecting MR3
10:00:11.71746 "MAI1950251842_1" #61: ignoring informational payload, type INVALID_ID_INFORMATION msgid=00000000
10:00:11.71750 "MAI1950251842_1" #61: received and ignored informational message
```

The **initiator** has sent his third *Main Mode* message (MI3) and expects now the response from the **responder** (*STATE_MAIN_I3: sent MI3, expecting MR3*). The **initiator** has sent with the third message its certificate or hash value of the PSK and expects now the according information from the **responder**.

But the **responder** did not send its certificate or hash value of the PSK, it returns an informational payload of the type INVALID_ID_INFORMATION.

The log of the **responder** will tell us more about the reason.

15.3.4.1 Responder: “no suitable connection for peer‘…’”

Responder Log:

```
10:00:08.88221 packet from 77.245.32.68:500: received Vendor ID payload [Openswan (this version) 2.6.24 ]
10:00:08.88231 packet from 77.245.32.68:500: received Vendor ID payload [Dead Peer Detection]
10:00:08.88238 packet from 77.245.32.68:500: received Vendor ID payload [RFC 3947] method set to=109
10:00:08.88245 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-03] meth=108, but already using method 109
10:00:08.88253 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02_n] meth=106, but already using method 109
10:00:08.88261 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02] meth=107, but already using method 109
10:00:08.88270 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-00]
10:00:08.88277 packet from 77.245.32.68:500: received Vendor ID payload [Innominate IKE Fragmentation]
10:00:08.88295 packet from 77.245.32.68:500: received Vendor ID payload [Innominate always send NAT-OA]
10:00:08.88304 "MAI0874627901_1"[1] 77.245.32.68 #73: responding to Main Mode from unknown peer 77.245.32.68
10:00:08.88312 "MAI0874627901_1"[1] 77.245.32.68 #73: enabling Innominate IKE Fragmentation (main_inI1_outR1)
10:00:08.88320 "MAI0874627901_1"[1] 77.245.32.68 #73: enabling Innominate Always Send NAT-OA (main_inI1_outR1)
10:00:08.88389 "MAI0874627901_1"[1] 77.245.32.68 #73: transition from state STATE_MAIN_R0 to state STATE_MAIN_R1
10:00:08.88433 "MAI0874627901_1"[1] 77.245.32.68 #73: STATE_MAIN_R1: sent MR1, expecting MI2
10:00:09.45098 "MAI0874627901_1"[1] 77.245.32.68 #73: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): both are NATed
10:00:11.23116 "MAI0874627901_1"[1] 77.245.32.68 #73: transition from state STATE_MAIN_R1 to state STATE_MAIN_R2
10:00:11.23140 "MAI0874627901_1"[1] 77.245.32.68 #73: STATE_MAIN_R2: sent MR2, expecting MI3
10:00:11.71884 "MAI0874627901_1"[1] 77.245.32.68 #73: Main mode peer ID is ID_DER_ASN1_DN: 'O=Innominate, OU=Support, CN=mGuard 3'
10:00:11.71893 "MAI0874627901_1"[1] 77.245.32.68 #73: issuer cacert not found
10:00:11.71900 "MAI0874627901_1"[1] 77.245.32.68 #73: X.509 certificate rejected
10:00:11.71908 "MAI0874627901_1"[1] 77.245.32.68 #73: no suitable connection for peer 'O=Innominate, OU=Support, CN=mGuard 3'
10:00:11.71916 "MAI0874627901_1"[1] 77.245.32.68 #73: sending encrypted notification INVALID_ID_INFORMATION to 77.245.32.68:500
```

The **responder** has received the third *Main Mode* message (MI3) but there is not VPN connection configured with a certificate matching to the subject of the received certificate.

Possible Reasons:

– Certificate or PSK mismatch. If PSK is used for authentication, ensure that the same Pre-Shared Secret Key was entered on both sides (menu **/Psec VPN >> Connections >> (Edit) >> Authentication**, parameter *Pre-Shared Secret Key (PSK)*). If certificates are used for authentication, compare the MD5 or SHA1 fingerprint of the machine certificate of the **initiator** (menu **Authentication >> Certificates >> Machine Certificates**) with the fingerprint of the Remote Certificate in the corresponding VPN connection of the **responder** (menu **IPsec VPN >> Connections >> (Edit) >> Authentication**).

– Mismatch of the specified VPN identifier (VPN connection tab *Authentication*), log entry e.g. “no suitable connection for peer '@mGuard 1'”

15.3.4.2 Responder: “Signature check (on …) failed (wrong key?)”

Responder Log:

```
10:30:56.12114 packet from 77.245.32.68:500: received Vendor ID payload [Openswan (this version) 2.6.24 ]
10:30:56.12123 packet from 77.245.32.68:500: received Vendor ID payload [Dead Peer Detection]
10:30:56.12130 packet from 77.245.32.68:500: received Vendor ID payload [RFC 3947] method set to=109
10:30:56.12138 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-03] meth=108, but already using method 109
10:30:56.12146 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02_n] meth=106, but already using method 109
10:30:56.12154 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02] meth=107, but already using method 109
10:30:56.12162 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-00]
10:30:56.12169 packet from 77.245.32.68:500: received Vendor ID payload [Innominate IKE Fragmentation]
10:30:56.12187 packet from 77.245.32.68:500: received Vendor ID payload [Innominate always send NAT-OA]
10:30:56.12196 "MAI0874627901_1"[1] 77.245.32.68 #94: responding to Main Mode from unknown peer 77.245.32.68
10:30:56.12204 "MAI0874627901_1"[1] 77.245.32.68 #94: enabling Innominate IKE Fragmentation (main_inI1_outR1)
10:30:56.12212 "MAI0874627901_1"[1] 77.245.32.68 #94: enabling Innominate Always Send NAT-OA (main_inI1_outR1)
10:30:56.12324 "MAI0874627901_1"[1] 77.245.32.68 #94: transition from state STATE_MAIN_R0 to state STATE_MAIN_R1
10:30:56.12371 "MAI0874627901_1"[1] 77.245.32.68 #94: STATE_MAIN_R1: sent MR1, expecting MI2
10:30:56.71292 "MAI0874627901_1"[1] 77.245.32.68 #94: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): both are NATed
10:30:58.51165 "MAI0874627901_1"[1] 77.245.32.68 #94: transition from state STATE_MAIN_R1 to state STATE_MAIN_R2
10:30:58.51189 "MAI0874627901_1"[1] 77.245.32.68 #94: STATE_MAIN_R2: sent MR2, expecting MI3
10:30:59.00185 "MAI0874627901_1"[1] 77.245.32.68 #94: Main mode peer ID is ID_DER_ASN1_DN: 'O=Innominate, OU=Support, CN=mGuard 1'
10:30:59.00233 "MAI0874627901_1"[1] 77.245.32.68 #94: issuer cacert not found
10:30:59.00241 "MAI0874627901_1"[1] 77.245.32.68 #94: X.509 certificate rejected
10:30:59.00248 "MAI0874627901_1"[1] 77.245.32.68 #94: Signature check (on O=Innominate, OU=Support, CN=mGuard 1) failed (wrong key?);
                                                      tried *AwEAAcBS4
```

Reason:

The machine certificate has been replaced by a new one on the **initiator**. The new certificate has the same subject attributes as the previous certificate. On the **responder**, the certificate of the **initiator** specified as remote certificate in the VPN connection (VPN connection tab *Authentication*) is still the previous one.

15.3.5 Initiator: “Signature Check (on …) failed (wrong key?)”

Initiator Log:

```
10:33:56.63023 "MAI1950251842_1" #85: initiating Main Mode
10:33:58.47973 "MAI1950251842_1" #85: received Vendor ID payload [Openswan (this version) 2.6.24 ]
10:33:58.47977 "MAI1950251842_1" #85: received Vendor ID payload [Dead Peer Detection]
10:33:58.47981 "MAI1950251842_1" #85: received Vendor ID payload [RFC 3947] method set to=109
10:33:58.47985 "MAI1950251842_1" #85: received Vendor ID payload [Innominate IKE Fragmentation]
10:33:58.47989 "MAI1950251842_1" #85: received Vendor ID payload [Innominate always send NAT-OA]
10:33:58.47993 "MAI1950251842_1" #85: enabling possible NAT-traversal with method 4
10:33:58.47997 "MAI1950251842_1" #85: enabling Innominate IKE Fragmentation (main_inR1_outI2)
10:33:58.48012 "MAI1950251842_1" #85: enabling Innominate Always Send NAT-OA (main_inR1_outI2)
10:33:58.81901 "MAI1950251842_1" #85: transition from state STATE_MAIN_I1 to state STATE_MAIN_I2
10:33:58.81909 "MAI1950251842_1" #85: STATE_MAIN_I2: sent MI2, expecting MR2
10:34:01.19738 "MAI1950251842_1" #85: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): both are NATed
10:34:01.19750 "MAI1950251842_1" #85: I am sending my cert
10:34:01.19753 "MAI1950251842_1" #85: I am sending a certificate request
10:34:01.19861 "MAI1950251842_1" #85: transition from state STATE_MAIN_I2 to state STATE_MAIN_I3
10:34:01.19880 "MAI1950251842_1" #85: STATE_MAIN_I3: sent MI3, expecting MR3
10:34:01.24550 "MAI1950251842_1" #85: Main mode peer ID is ID_DER_ASN1_DN: 'O=Innominate, OU=Support, CN=mGuard 2'
10:34:01.24554 "MAI1950251842_1" #85: issuer cacert not found
10:34:01.24558 "MAI1950251842_1" #85: X.509 certificate rejected
10:34:01.24561 "MAI1950251842_1" #85: Signature check (on O=Innominate, OU=Support, CN=mGuard 2) failed (wrong key?); tried *AwEAAbns8
10:34:01.24566 "MAI1950251842_1" #85: sending encrypted notification INVALID_KEY_INFORMATION to 77.245.33.67:4500
```

Reason:

The machine certificate has been replaced by a new one on the **responder**. The new certificate has the same subject attributes as the previous certificate. On the **initiator**, the certificate of the **responder** specified as remote certificate in the VPN connection (VPN connection tab *Authentication*) is still the previous one.

15.3.6 Initiator: “we require peer to have ID ‘…’, but peer declares ‘…’”

Initiator Log:

```
10:06:12.36092 "MAI1950251842_1" #67: initiating Main Mode
10:06:14.17361 "MAI1950251842_1" #67: received Vendor ID payload [Openswan (this version) 2.6.24 ]
10:06:14.17365 "MAI1950251842_1" #67: received Vendor ID payload [Dead Peer Detection]
10:06:14.17369 "MAI1950251842_1" #67: received Vendor ID payload [RFC 3947] method set to=109
10:06:14.17373 "MAI1950251842_1" #67: received Vendor ID payload [Innominate IKE Fragmentation]
10:06:14.17377 "MAI1950251842_1" #67: received Vendor ID payload [Innominate always send NAT-OA]
10:06:14.17381 "MAI1950251842_1" #67: enabling possible NAT-traversal with method 4
10:06:14.17385 "MAI1950251842_1" #67: enabling Innominate IKE Fragmentation (main_inR1_outI2)
10:06:14.17400 "MAI1950251842_1" #67: enabling Innominate Always Send NAT-OA (main_inR1_outI2)
10:06:14.48008 "MAI1950251842_1" #67: transition from state STATE_MAIN_I1 to state STATE_MAIN_I2
10:06:14.48016 "MAI1950251842_1" #67: STATE_MAIN_I2: sent MI2, expecting MR2
10:06:16.85786 "MAI1950251842_1" #67: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): both are NATed
10:06:16.85798 "MAI1950251842_1" #67: I am sending my cert
10:06:16.85801 "MAI1950251842_1" #67: I am sending a certificate request
10:06:16.85848 "MAI1950251842_1" #67: transition from state STATE_MAIN_I2 to state STATE_MAIN_I3
10:06:16.85867 "MAI1950251842_1" #67: STATE_MAIN_I3: sent MI3, expecting MR3
10:06:16.90526 "MAI1950251842_1" #67: Main mode peer ID is ID_DER_ASN1_DN: 'O=Innominate, OU=Support, CN=mGuard 3'
10:06:16.90531 "MAI1950251842_1" #67: issuer cacert not found
10:06:16.90534 "MAI1950251842_1" #67: X.509 certificate rejected
10:06:16.90538 "MAI1950251842_1" #67: we require peer to have ID 'O=Innominate, OU=Support, CN=mGuard 2', but peer declares 'O=Innominate,
                                      OU=Support, CN=mGuard 3'
10:06:16.90543 "MAI1950251842_1" #67: sending encrypted notification INVALID_ID_INFORMATION to 77.245.33.67:4500
10:06:16.90933 "MAI1950251842_1" #67: received 1 malformed payload notifies
```

The **initiator** has received the third *Main Mode* response from the **responder** (MR3) with the certificate of the remote side but the certificate’s subject does not match to the one specified in the VPN connection as remote certificate (VPN connection tab *Authentication*).

Responder Log:

```
10:06:14.03024 packet from 77.245.32.68:500: received Vendor ID payload [Openswan (this version) 2.6.24 ]
10:06:14.03033 packet from 77.245.32.68:500: received Vendor ID payload [Dead Peer Detection]
10:06:14.03040 packet from 77.245.32.68:500: received Vendor ID payload [RFC 3947] method set to=109
10:06:14.03047 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-03] meth=108, but already using method 109
10:06:14.03055 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02_n] meth=106, but already using method 109
10:06:14.03063 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-02] meth=107, but already using method 109
10:06:14.03071 packet from 77.245.32.68:500: received Vendor ID payload [draft-ietf-ipsec-nat-t-ike-00]
10:06:14.03078 packet from 77.245.32.68:500: received Vendor ID payload [Innominate IKE Fragmentation]
10:06:14.03096 packet from 77.245.32.68:500: received Vendor ID payload [Innominate always send NAT-OA]
10:06:14.03105 "MAI0874627901_1"[1] 77.245.32.68 #79: responding to Main Mode from unknown peer 77.245.32.68
10:06:14.03113 "MAI0874627901_1"[1] 77.245.32.68 #79: enabling Innominate IKE Fragmentation (main_inI1_outR1)
10:06:14.03120 "MAI0874627901_1"[1] 77.245.32.68 #79: enabling Innominate Always Send NAT-OA (main_inI1_outR1)
10:06:14.03188 "MAI0874627901_1"[1] 77.245.32.68 #79: transition from state STATE_MAIN_R0 to state STATE_MAIN_R1
10:06:14.03232 "MAI0874627901_1"[1] 77.245.32.68 #79: STATE_MAIN_R1: sent MR1, expecting MI2
10:06:14.65862 "MAI0874627901_1"[1] 77.245.32.68 #79: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): both are NATed
10:06:16.39205 "MAI0874627901_1"[1] 77.245.32.68 #79: transition from state STATE_MAIN_R1 to state STATE_MAIN_R2
10:06:16.39228 "MAI0874627901_1"[1] 77.245.32.68 #79: STATE_MAIN_R2: sent MR2, expecting MI3
10:06:16.90888 "MAI0874627901_1"[1] 77.245.32.68 #79: Main mode peer ID is ID_DER_ASN1_DN: 'O=Innominate, OU=Support, CN=mGuard 1'
10:06:16.90896 "MAI0874627901_1"[1] 77.245.32.68 #79: issuer cacert not found
10:06:16.90904 "MAI0874627901_1"[1] 77.245.32.68 #79: X.509 certificate rejected
10:06:16.90911 "MAI0874627901_1"[1] 77.245.32.68 #79: I am sending my cert
10:06:16.91022 "MAI0874627901_1"[1] 77.245.32.68 #79: transition from state STATE_MAIN_R2 to state STATE_MAIN_R3
10:06:16.91038 "MAI0874627901_1"[1] 77.245.32.68 #79: new NAT mapping for #79, was 77.245.32.68:500, now 77.245.32.68:4500
10:06:16.91091 "MAI0874627901_1"[1] 77.245.32.68 #79: new NAT mapping for #78, was 77.245.32.68:500, now 77.245.32.68:4500
10:06:16.91111 "MAI0874627901_1"[1] 77.245.32.68 #79: STATE_MAIN_R3: sent MR3, ISAKMP SA established {auth=OAKLEY_RSA_SIG
                                                      cipher=oakley_3des_cbc_192 prf=oakley_md5 group=modp8192}
10:06:16.91121 "MAI0874627901_1"[1] 77.245.32.68 #79: Dead Peer Detection (RFC 3706): enabled
10:06:16.91576 "MAI0874627901_1"[1] 77.245.32.68 #79: next payload type of ISAKMP Hash Payload has an unknown value: 234
10:06:16.91604 "MAI0874627901_1"[1] 77.245.32.68 #79: next payload type of ISAKMP Hash Payload has an unknown value: 234
```

Due to the certificate failure, the **initiator** responds with INVALID_ID_INFORMATION.

The *ISAKMP SA* was established successfully for the **responder**. Thus he is expecting now the first packet for the establishment of the *IPsec SA* but did not receive it.

Possible reasons:

– Certificate mismatch. Compare the MD5 or SHA1 fingerprint of the machine certificate of the **responder** (menu **Authentication >> Certificates >> Machine Certificates**) with the fingerprint of the Remote Certificate in the corresponding VPN connection of the **initiator** (menu **IPsec VPN >> Connections >> (Edit) >> Authentication**).

– Mismatch of the specified VPN identifier (VPN connection tab *Authentication*), log entry e.g. “we require peer to have ID 'O=Innominate, OU=Support, CN=mGuard 2', but peer declares '@mGuard 2'”.

15.4 IPSec SA (phase II) can not be established

The *IPsec SA* is established using the *Quick Mode* provided by the *Internet Key Exchange* (IKE) protocol. Basically three messages are exchanged in this mode.

If the establishment of the *IPsec SA* fails, it is caused by a configuration mismatch. Either the specified VPN networks do not match, or there is a mismatch of the specified encryption and/or hash algorithm for the *IPsec SA* (tab *IKE Options*), or *Perfect Forward Secrecy* is enabled on the **responder** but not on the **initiator**.

15.4.1 Initiator: “ignoring informational payload, type NO_PROPOSAL_CHOSEN”

Initiator Log:

```
15:50:00.48413 "MAI1950251842_1" #80: initiating Main Mode
---------- Establishment of the ISAKMP SA ----------
15:50:05.34633 "MAI1950251842_1" #80: STATE_MAIN_I4: ISAKMP SA established {auth=OAKLEY_RSA_SIG cipher=oakley_3des_cbc_192
                    prf=oakley_md5 group=modp8192}
15:50:05.34638 "MAI1950251842_1" #80: Dead Peer Detection (RFC 3706): enabled

15:50:05.34642 "MAI1950251842_1" #81: initiating Quick Mode RSASIG+ENCRYPT+TUNNEL+PFS+UP {using isakmp#80 msgid:738f09c4
                    proposal=AES(12)_128-MD5(1)_128 pfsgroup=OAKLEY_GROUP_MODP8192}
15:50:05.64835 "MAI1950251842_1" #80: ignoring informational payload, type NO_PROPOSAL_CHOSEN msgid=00000000
15:50:05.64839 "MAI1950251842_1" #80: received and ignored informational message
```

Responder Log:

```
15:50:00.94309 "MAI0874627901_1"[1] 77.245.32.68 #90: responding to Main Mode from unknown peer 77.245.32.68
---------- Establishment of the ISAKMP SA ----------
15:50:03.83320 "MAI0874627901_1"[1] 77.245.32.68 #90: STATE_MAIN_R3: sent MR3, ISAKMP SA established {auth=OAKLEY_RSA_SIG
                                                      cipher=oakley_3des_cbc_192 prf=oakley_md5 group=modp8192}
15:50:03.83330 "MAI0874627901_1"[1] 77.245.32.68 #90: Dead Peer Detection (RFC 3706): enabled

15:50:04.32312 "MAI0874627901_1"[1] 77.245.32.68 #90: the peer proposed: 192.168.20.0/24:0/0 -> 192.168.10.0/24:0/0
15:50:04.32337 "MAI0874627901_1"[1] 77.245.32.68 #91: IPsec Transform [ESP_AES (128), AUTH_ALGORITHM_HMAC_MD5] refused due to strict
                                                      flag
15:50:04.32424 "MAI0874627901_1"[1] 77.245.32.68 #91: no acceptable Proposal in IPsec SA
```

Reason:

Mismatch of the specified encryption and/or hash algorithms for the *IPsec SA*. Check the specified encryption and hash algorithms for the *IPsec SA* on the **initiator** and on the **responder** (menu **IPsec VPN >> Connections >> (Edit) >> IKE Options**, section *IPsec SA* (*Data Exchange*)). Both VPN connections need to support the same encryption and hash algorithm.

15.4.2 Initiator: “ignoring informational payload, type INVALID_ID_INFORMATION”

Initiator Log:

```
16:08:21.07207 "MAI1950251842_1" #104: initiating Main Mode
---------- Establishment of the ISAKMP SA ----------
16:08:25.85346 "MAI1950251842_1" #104: STATE_MAIN_I4: ISAKMP SA established {auth=OAKLEY_RSA_SIG
                cipher=oakley_3des_cbc_192 prf=oakley_md5 group=modp8192}
16:08:25.85351 "MAI1950251842_1" #104: Dead Peer Detection (RFC 3706): enabled
16:08:25.85354 "MAI1950251842_1" #105: initiating Quick Mode RSASIG+ENCRYPT+TUNNEL+PFS+UP {using isakmp#104 msgid:ed708573
                proposal=3DES(3)_192-MD5(1)_128 pfsgroup=OAKLEY_GROUP_MODP8192}
16:08:26.20417 "MAI1950251842_1" #104: ignoring informational payload, type INVALID_ID_INFORMATION msgid=00000000
16:08:26.20422 "MAI1950251842_1" #104: received and ignored informational message
```

Responder Log:

```
16:08:21.51698 "MAI0874627901_1"[1] 77.245.32.68 #126: responding to Main Mode from unknown peer 77.245.32.68
---------- Establishment of the ISAKMP SA ----------
16:08:24.41158 "MAI0874627901_1"[1] 77.245.32.68 #126: STATE_MAIN_R3: sent MR3, ISAKMP SA established {auth=OAKLEY_RSA_SIG
                cipher=oakley_3des_cbc_192 prf=oakley_md5 group=modp8192}
16:08:24.41169 "MAI0874627901_1"[1] 77.245.32.68 #126: Dead Peer Detection (RFC 3706): enabled
16:08:24.87992 "MAI0874627901_1"[1] 77.245.32.68 #126: the peer proposed: 192.168.20.0/24:0/0 -> 192.168.10.0/24:0/0
16:08:24.88001 "MAI0874627901_1"[1] 77.245.32.68 #126: cannot respond to IPsec SA request because no connection is known for
                192.168.20.0/24===192.168.3.1[O=Innominate, OU=Support, CN=mGuard 2]...77.245.32.68[O=Innominate, OU=Support,
                CN=mGuard 1]===192.168.10.0/24}
16:08:24.88012 "MAI0874627901_1"[1] 77.245.32.68 #126: sending encrypted notification INVALID_ID_INFORMATION to 77.245.32.68:4500
```

Reason:

The specified VPN networks (VPN connection tab *General*) do not match on both sides. The local network specified on one side must be specified as remote network on the other side and vice versa.

15.4.3 Initiator: “No acceptable response to our first Quick Mode message: perhaps peer likes no proposal”

Initiator Log:

```
09:20:12.96824 "MAI1950251842_1" #15: initiating Main Mode
---------- Establishment of the ISAKMP SA ----------
09:20:17.64568 "MAI1950251842_1" #15: STATE_MAIN_I4: ISAKMP SA established {auth=OAKLEY_RSA_SIG
                                      cipher=oakley_3des_cbc_192 prf=oakley_md5 group=modp8192}
09:20:17.64573 "MAI1950251842_1" #15: Dead Peer Detection (RFC 3706): enabled
09:20:17.64577 "MAI1950251842_1" #16: initiating Quick Mode RSASIG+ENCRYPT+TUNNEL+UP {using isakmp#15 msgid:1acc17dd
                                      proposal=3DES(3)_192-MD5(1)_128 pfsgroup=no-pfs}
09:21:27.63790 "MAI1950251842_1" #16: max number of retransmissions (2) reached STATE_QUICK_I1. No acceptable response to our
                                      first Quick Mode message: perhaps peer likes no proposal
```

Responder Log:

```
09:20:14.74888 packet from 77.245.32.68:500: received Vendor ID payload [Openswan (this version) 2.6.24 ]
---------- Establishment of the ISAKMP SA ----------
09:20:17.63925 "MAI0874627901_1"[1] 77.245.32.68 #5: STATE_MAIN_R3: sent MR3, ISAKMP SA established {auth=OAKLEY_RSA_SIG
                                                     cipher=oakley_3des_cbc_192 prf=oakley_md5 group=modp8192}
09:20:17.63935 "MAI0874627901_1"[1] 77.245.32.68 #5: Dead Peer Detection (RFC 3706): enabled
09:20:17.65065 "MAI0874627901_1"[1] 77.245.32.68 #5: the peer proposed: 192.168.20.0/24:0/0 -> 192.168.10.0/24:0/0
09:20:17.65090 "MAI0874627901_1"[1] 77.245.32.68 #6: we require PFS but Quick I1 SA specifies no GROUP_DESCRIPTION
```

Reason:

Perfect Forward Secrecy (PFS) is enabled on the **responder** but not on the **initiator** (VPN connection tab *IKE Options*, section *IPsec SA (Data Exchange)*).

15.5 Remote network clients can not be reached through established VPN tunnel

If the VPN connection was established successfully, problems related to transferring data through the VPN tunnel are usually not caused by the mGuard devices and have external reasons.

If VPN masquerading is configured on one mGuard, connections can only be established from the masqueraded network to the other network, not vice versa.

The following steps help to narrow down the reason for the problem, assuming the VPN firewall does not block ICMP traffic.

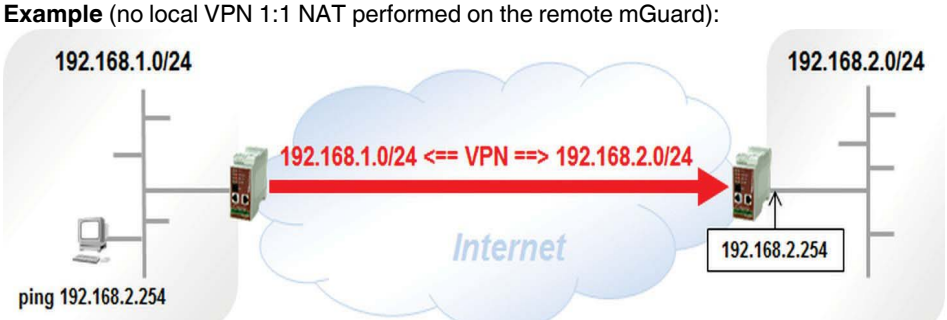
Step 1: Is the internal IP of the local mGuard reachable from the internal Client?

- The first step is to check if the internal IP of the local mGuard is reachable from the client from which the remote VPN network should be accessed.
- From the client, send a “ping” to the internal IP of the local mGuard.
- If the “ping” is not replied, the reason is located in the internal network. If the “ping” is replied, proceed with the next step.

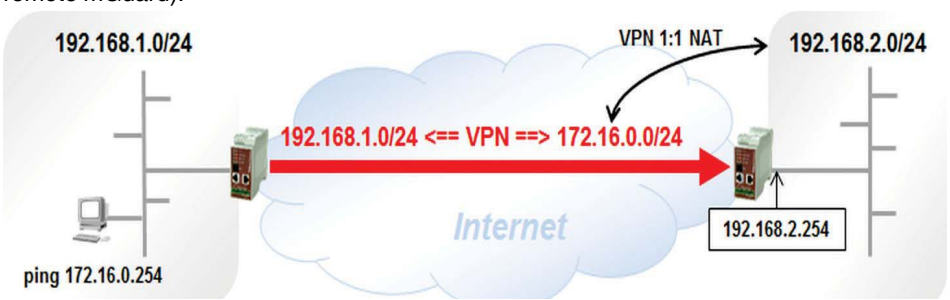
Step 2: Internal IP of the Remote mGuard reachable through VPN?

- The next step is to check if a “ping” to the internal IP of the remote mGuard through the VPN connection is replied.

Example (no local VPN 1:1 NAT performed on the remote mGuard):



Example (Local VPN 1:1 NAT from 172.16.0.0/24 to 192.168.2.0/24 performed on the remote mGuard):



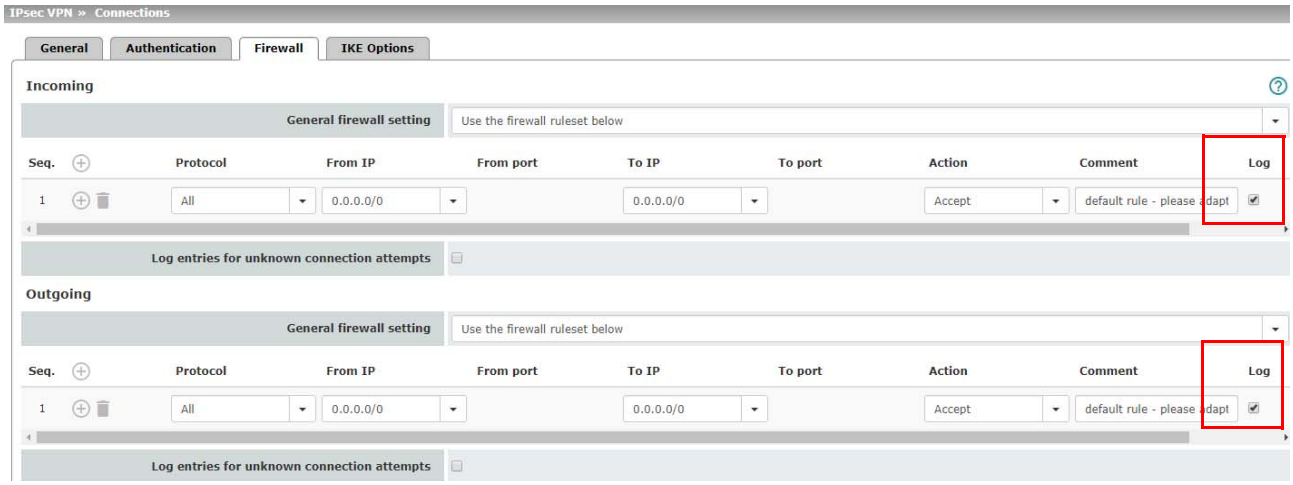
If the “ping” is replied, the reason for the problem why clients of the remote VPN network cannot be accessed is located in the remote network. Maybe the internal IP of the remote mGuard is not specified as default gateway on the remote clients.

If the “ping” is not replied, proceed with the next step.

Step 3: Do the Packets enter the VPN and are they received on the remote Side?

The next step is to verify if the sent packets enter the VPN connection and if they are received by the remote side. To check this, enable the VPN firewall logging on both sides.

- Edit the VPN connection (menu **IPsec VPN >> Connections**).
- Switch to the tab *Firewall*.
- Enable the logging.



- Click the icon <*Save*>.

The VPN connection will be interrupted due to the configuration change. Wait until the connection is up again (menu **IPsec VPN >> IPsec Status**), send the data and then check the VPN firewall logs.The log entries are display in the menu **Logging >> Browse local logs**, option *Network Security*.

Example, ICMP echo requests entering the VPN tunnel (fw-vpn_...-out-…):

```
14:57:33.68468 kernel: fw-vpn_MAI1950251842-out-1-123bacb5-b892-103f-88ac-000cbe020f10 act=ACCEPT IN=eth1 OUT=eth0 SRC=192.168.1.100
                       DST=192.168.20.1 LEN=60 TOS=0x00 PREC=0x00 TTL=127 ID=20250 PROTO=ICMP TYPE=8 CODE=0 ID=512 SEQ=5632
14:57:38.95374 kernel: fw-vpn_MAI1950251842-out-1-123bacb5-b892-103f-88ac-000cbe020f10 act=ACCEPT IN=eth1 OUT=eth0 SRC=192.168.1.100
                       DST=192.168.20.1 LEN=60 TOS=0x00 PREC=0x00 TTL=127 ID=20251 PROTO=ICMP TYPE=8 CODE=0 ID=512 SEQ=5888
```

Example, ICMP echo requests received through the VPN tunnel (fw-vpn_...-in-…):

```
14:57:33.68384 kernel: fw-vpn_MAI0874627901-in-1-2a407f3f-1020-1141-a3a4-000cbe020e08 act=ACCEPT IN=eth0 OUT=eth1 SRC=192.168.10.100
                       DST=192.168.1.1 LEN=60 TOS=0x00 PREC=0x00 TTL=126 ID=20250 PROTO=ICMP TYPE=8 CODE=0 ID=512 SEQ=5632
14:57:38.95130 kernel: fw-vpn_MAI0874627901-in-1-2a407f3f-1020-1141-a3a4-000cbe020e08 act=ACCEPT IN=eth0 OUT=eth1 SRC=192.168.10.100
                       DST=192.168.1.1 LEN=60 TOS=0x00 PREC=0x00 TTL=126 ID=20251 PROTO=ICMP TYPE=8 CODE=0 ID=512 SEQ=5888
```

Possible results of this test:

1. Local mGuard does not display according outgoing (entering the VPN tunnel) log messages (fw-vpn-…-out-…).
   - The internal IP address of the local mGuard is not specified as default gateway on the client on which the command has been issued.
   - If the client has to use a different default gateway than the local mGuard, no route is defined to direct the packets for the remote VPN network to the local mGuard.

- A firewall between the client and the local mGuard blocks the traffic.
- Reason located somewhere in the internal network.

2. Remote mGuard does not display according incoming (received through the VPN tunnel) log messages (fw-vpn-…-in-…).
   - Some entity (gateway, router) between the two VPN peers blocks the encrypted traffic. The following cases already have been observed:
     - Some provider only allow incoming encrypted packets from the Internet to their network if outgoing encrypted packets have already been seen for this connection. This was observed with a satellite network provider as well as with a telephone network provider. To verify this, try to access the clients of the local network from the remote VPN network.
     - A router has blocked ESP traffic between the mGuard devices. This problem could be solved by forcing UDP encapsulation on the mGuard device. This option can only be activated from the command line (*gaiconfig --set VPN_CONNECTION.x.FORCE_UDP_ENCAPS yes, 'x'* standing for the number of the configured VPN connection (0, 1, 2, 3, …)). The router has blocked the ESP traffic but not UDP packets encapsulating the ESP packets.

15.6 Other Problems

15.6.1 VPN connections fails after 24 hours

This problem usually happens if the **responder** has a dynamic public IP address which changes every 24h, registers the current IP address under a specific name in a DynDNS service, the **initiator** refers to this DNS name as "*Address of the remote VPN gateway*", but DynDNS monitoring is not enabled on the **initiator**.

- On the **initiator**, switch to the menu **IPsec VPN >> Global >> DynDNS Monitoring**.
- Set *Watch hostnames of remote VPN Gateways* to *Yes*.
- Click the icon <*Save*>.

15.6.2 Problems transferring huge Data

A remote client responds to small packets (e.g. "pings") without problems but transferring huge data (e.g. Remote Desktop Application) fails. This problem is usually caused by routers in the Internet, which reduce the MTU size but do not support UDP fragmentation. The mGuard device receives fragments of encrypted UDP packets and cannot decode them.

This problem can be solved by reducing the IPsec MTU size on the mGuard device. Thus encrypted packets have a smaller size and will not be fragmented when passing the router which reduces the MTU size.

The IPsec MTU size needs to be reduced on the mGuard device where the huge data enter the VPN connection.

- Switch to the menu **IPsec VPN >> Global >> Options**.
- Reduce the size of the *IPsec MTU* in the section *IP Fragmentation*.
- Click the icon <*Save*>.

You need to reduce the IPsec MTU size successively until the huge data get through the tunnel.

15.7 Quick Reference: VPN Log Error Messages

Table 15-2 Quick Reference: VPN Log Error Messages

VPN log error messages	Refer to chapter
ikelifetime […] must be greater than rekeymargin*(100+rekeyfuzz)/100	Section 15.2
tunnel ignored: local address 'w.x.y.x' within remote network 'a.b.c.d/e'	Section 15.2
Initiator Error Messages	
pending Quick Mode with w.x.y.z took too long – replacing phase 1	Section 15.3.2
Possible authentication failure: no acceptable response to our first encrypted message	Section 15.3.3
discarding duplicate packet; already STATE_MAIN_I3	Section 15.3.3
ignoring informational payload, type INVALID_ID_INFORMATION (during establishment of ISAKMP SA)	Section 15.3.4
Signature Check (on ...) failed (wrong key?)	Section 15.3.5
we require peer to have ID '...', but peer declares '...'	Section 15.3.6
ignoring informational payload, type NO_PROPOSAL_CHOSEN	Section 15.4.1
ignoring informational payload, type INVALID_ID_INFORMATION (during establishment of IPsec SA)	Section 15.3.4
No acceptable response to our first Quick Mode message: perhaps peer likes no proposal	Section 15.4.3
Responder Error Messages	
initial Main Mode message received on w.x.y.z:500 but no connection has been authorized	Section 15.3.2.2
max number of retransmissions (2) reached STATE_MAIN_R2	Section 15.3.3
no suitable connection for peer '...'	Section 15.3.4.1
Signature check (on ...) failed (wrong key?)	Section 15.3.4.2
next payload type of ISAKMP Hash Payload has an unknown value	Section 15.3.6
IPsec Transform [...] refused due to strict flag	Section 15.4.1
no acceptable Proposal in IPsec SA	Section 15.4.1
cannot respond to IPsec SA request because no connection is known for ...	Section 15.3.4
sending encrypted notification INVALID_ID_INFORMATION to …	Section 15.3.4
we require PFS but Quick I1 SA specifies no GROUP_DESCRIPTION	Section 15.4.3

16 Using CIFS Integrity Monitoring

Document ID: 108419_en_00
Document designation: AH EN MGUARD CIFS

Make sure you always use the latest documentation.
It is available for download at phoenixcontact.net/products.

The use of CIFS IM is subject to licensing and is not available on all devices.

Contents of this document

This document describes how to use the *CIFS Integrity Monitoring* mGuard function.

16.1 Introduction

CIFS stands for *Common Internet File System*, better known as *Windows File Sharing*.

CIFS Integrity Monitoring (CIFS-IM) is antivirus protection – or an antivirus sensor – for use in industrial applications that is able to detect whether a Windows-based system (machine controller, operator interface, PC) has been infected with malicious software, without the need to load virus signatures.

As a part of the CIFS integrity check, the Windows shares are checked to determine whether certain files (e.g. *.exe, *.dll) have been changed. Changes to these files indicate a possible virus or unauthorized intervention.

CIFS-IM can also be used for version control and monitoring.

16.1.1 Purpose

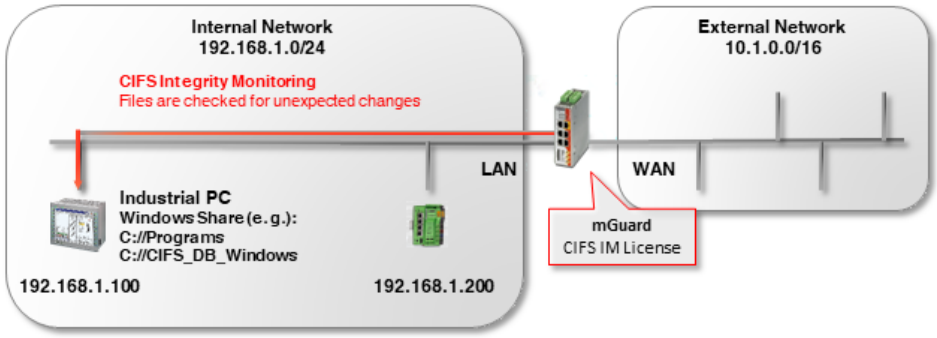


Figure 16-1 CIFS Integrity Monitoring – diagram

CIFS-IM is generally used in conjunction with the firewall function on mGuard devices for protecting *non-patchable systems*.

Non-patchable systems are primarily Window-based systems which either

a) **have an outdated operating system** for which security updates are no longer being provided (e.g. Windows 2000 / Windows XP),

b) **may no longer be modified** because the delivery state has been certified by the manufacturer or an authority and the manufacturer's warranty or the authority's approval would be forfeited in the event of a software modification,

c) **cannot be equipped with a virus scanner**, e.g. due to time-critical industrial applications (*real-time* capacity); or there is no way to update a virus signature because there is no connection to the Internet.

Non-patchable systems can be found in a number of different branches of industry. These include medicine (e.g. MRI, CT), the chemical and pharmaceutical industry (e.g. analysis systems), but also in production (e.g. PC-based machine controllers, plant data collection).

16.1.2 Method of operation

As a part of the **CIFS integrity check**, Windows shares are regularly checked to determine whether certain (executable) files (e.g. *.exe, *.dll) have been changed compared to a reference status in the integrity database.

The **integrity database** contains the checksums (hash values) for all the files that are checked. If the checksum of a file has changed, this indicates that the file has been modified, which in turn indicates a possible virus/worm attack or unauthorized intervention. It also detects whether new files have been added or files have been deleted.

The integrity database is created either when a share is checked for the first time or upon explicit request (e.g. after intentionally changing one or more files on the share). It is signed with an mGuard device machine certificate which protects it against tampering.

If the CIFS integrity check detects a deviation, an alarm can be sent out via e-mail or SNMP (SNMP trap).

16.1.3 Advantages over other antivirus systems

CIFS Integrity Monitoring offers the following advantages in the industrial environment:

a) There is no or almost no burden on the system being monitored (CPU performance, network load).
b) A connection to the Internet or to an update server is not required.
c) There is no need to reinstall virus signatures.
d) There are generally no false alarms (*false positives*) – and if one does occur, it has no effect on the system being monitored, since no data is deleted or moved to quarantine.

16.2 Configuration Example

On a Windows PC, the directory that is monitored is *C://Programs*. A user with the user name *CIFS* is created on the PC being monitored who has read access to the *C://Programs* directory.

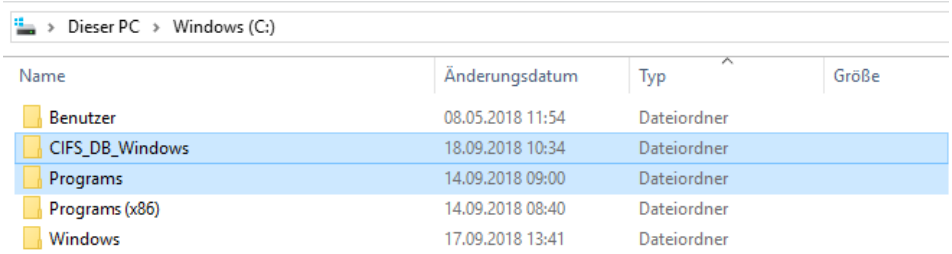


Figure 16-2 Creating directories / the integrity database

The integrity database should be saved to the *CIFS_DB_Windows* directory on the PC being monitored. The *CIFS* user also has read/write access to this directory.

16.3 Requirements

– The PC to be monitored must be situated in the network 192.168.1.0/24 and be accessible at the IP address 192.168.1.100.
– The mGuard device must be accessible at the IP address 192.168.1.1.
– The optional *CIFS Integrity Monitoring* license is present and available for purchase on the device.

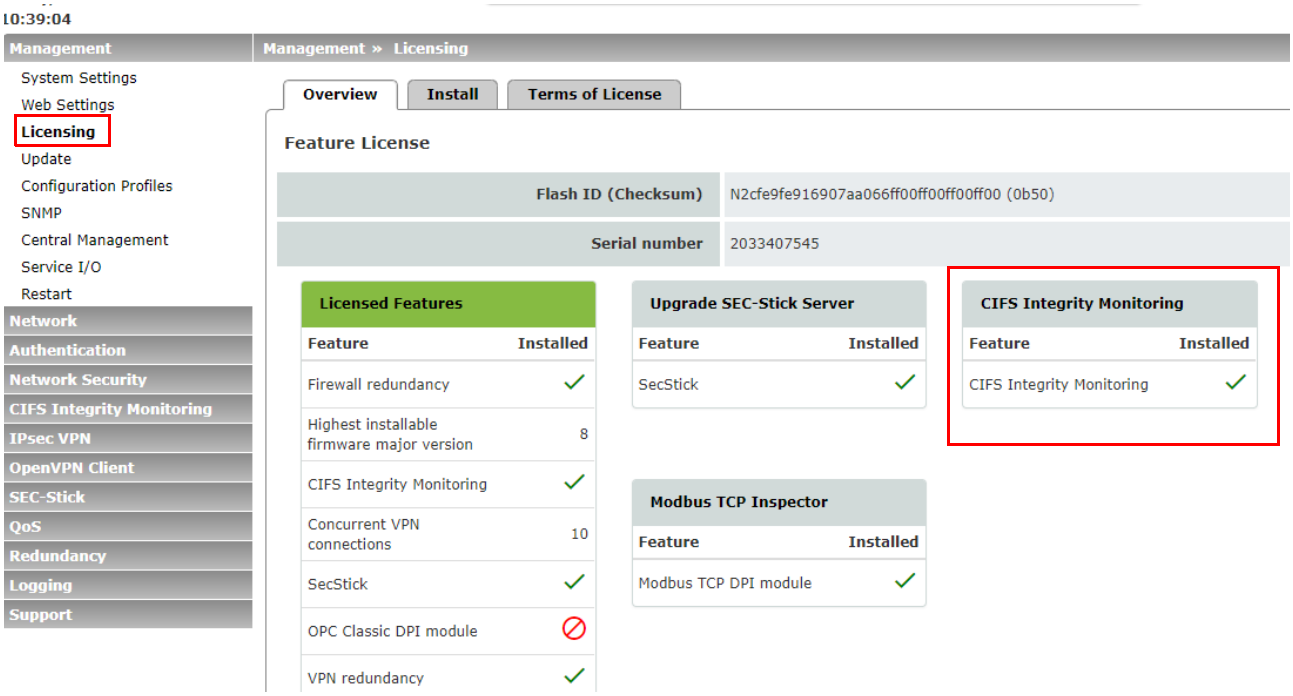


Figure 16-3 CIFS Integrity Monitoring license on the device

CIFS-IM is configured using the web-based management tool on the mGuard device (shown here: firmware version 8.7.0).

16.4 Importing a machine certificate

The machine certificate selected in the CIFS IM menu as the *integrity certificate* is used to sign and check the integrity database so that it cannot be replaced or tampered with by an intruder without being detected.

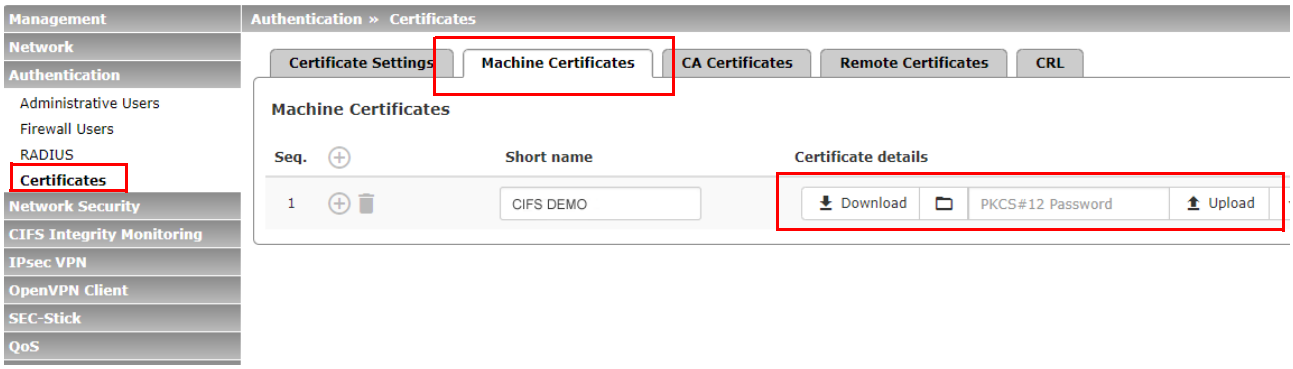


Figure 16-4 Installed machine certificate for use with CIFS IM

To import a machine certificate, proceed as follows:

1. Log on to the mGuard device web-based management.
2. Go to **Authentication >> Certificates** (*Machine certificates* tab).
3. Click on the ⊕ icon to add a new machine certificate.
4. Click on the 🗀 icon to select the certificate file (PKCS#12) on the installation computer.
5. Enter the PKCS#12 password issued when generating the certificate.
6. Give the certificate a unique short name. If you leave this field empty, the *common name (CN)* of the certificate is used automatically.
7. Click on the **Upload** button to import the certificate into the mGuard device.
8. Click on the "*Save*" icon 💾 to complete the import.

16.5 Configuring/importing shares

The Windows shares to be monitored are configured or imported on the mGuard device. The location where the integrity database and the test report should be stored is also configured/imported as a share.

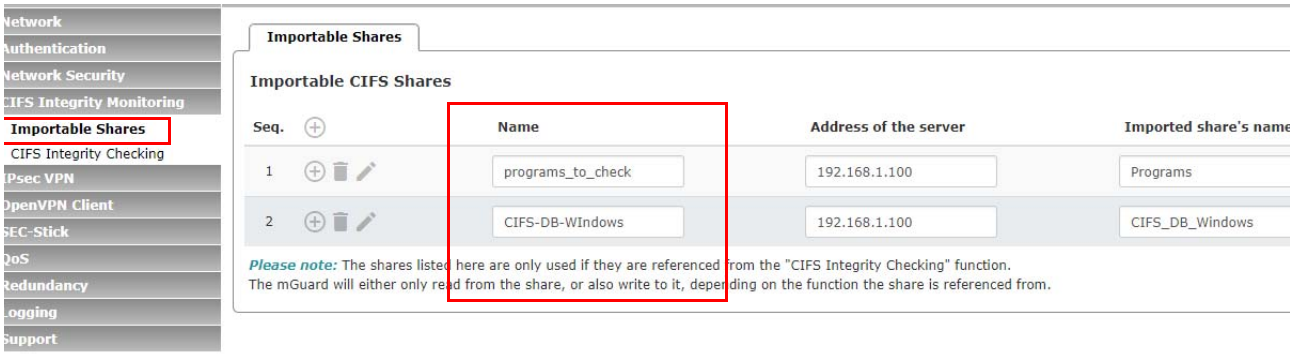


Figure 16-5 Imported shares for use with CIFS-IM

To import shares into the mGuard device, proceed as follows:

- Go to **CIFS Integrity Monitoring >> Importable Shares**.
- Click on the ⊕ icon to add a new share.
- Click on the ✎ icon to configure the share.

The designations that the mGuard device uses to internally manage the shares is indicated under **Name**. **Imported share's name** is the name of the approved Windows directory and must be adopted exactly as is:

  - The **name** *"programs_to_check"* is the internal mGuard designation for the **Imported share's name** *"C:\Programs"*.
  - The **name** *"CIFS- DB-Windows"* is the internal mGuard designation for the **Imported share's name** *"C:\CIFS_DB_Windows"*.

⇒ The mGuard device now knows the shares and can check them.

16.6 Configuring parameters for integrity checks

The integrity certificate used to sign the integrity databases must now be selected. If you wish to receive e-mail notification of integrity checks when they are done, you must configure the settings here accordingly.

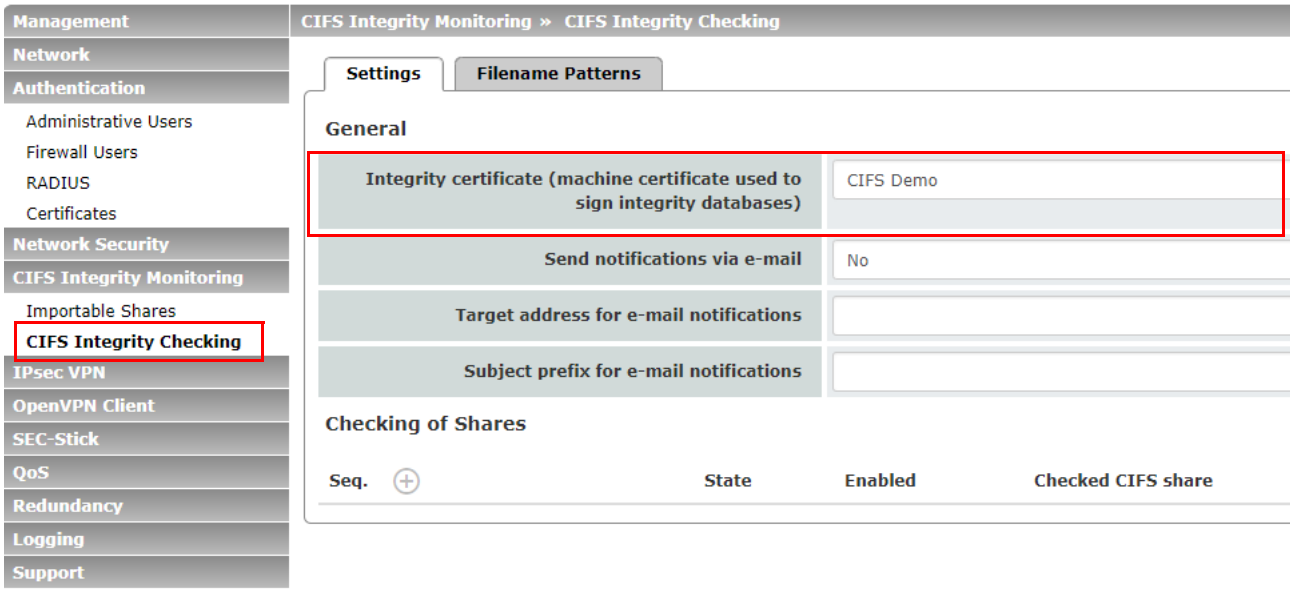


Figure 16-6 Selecting the machine certificate and configuring e-mail notification

- Go to **CIFS Integrity Monitoring >> CIFS Integrity Check** (*Settings tab*).
- Select the machine certificate to be used for the CIFS IM.
- **Optional**: Specify whether an e-mail notification should be sent (with every integrity check or only if errors/deviations are found).
  The mGuard device must have access to an e-mail server for this option. Configure this under **Management >> System Settings** (*E-Mail* tab).

16.7 Specifying the files to check

The file types and/or file directories to be included or excluded from monitoring are specified on the *Filename Patterns* tab.

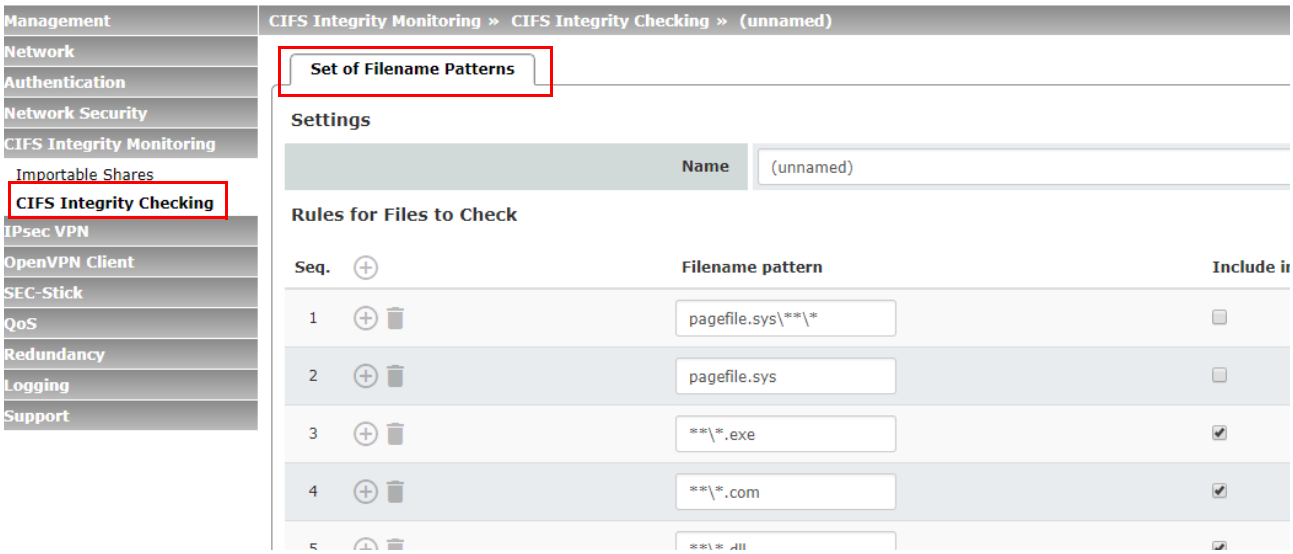


Figure 16-7 The files to be checked are specified using patterns

Proceed as follows:

- Go to **CIFS Integrity Monitoring >> CIFS Integrity Checking** (*Filename Patterns* tab).
- Specify the file types or file patterns to be checked.
  The mGuard device starts by offering a file pattern that can be either adopted or modified.

Patterns for filenames

***.exe means that the files located in a specific directory and with file extension **.exe* are checked (or excluded).

** at the start means that any directory is searched, even those at the top level, if this is empty. This cannot be combined with other characters (e.g., *c*** is not permitted).

Placeholders (*) represent any characters, e.g. *win**.exe* returns files with the extension **.exe* that are located in a directory that begins with *win*... Only one placeholder is permitted per directory or file name.

Example: *Name***.exe* refers to all files with the extension *.exe* that are located in the "*Name*" directory and any subdirectories.

Include in check

Activate function (include): files are included in the check.
Deactivate function (exclude): files are excluded from the check.

(Each file name is compared with the patterns in sequence. The first hit determines whether the file is to be included in the integrity check. The file is not included if no hits are found.)

16.8 Creating check sequences

You can create one or more check sequences that check different shares, directories, or file types.

A time-controlled check is configured for each check sequence (see also the mGuard firmware manual, available at phoenixcontact.net/products or help.mguard.com).

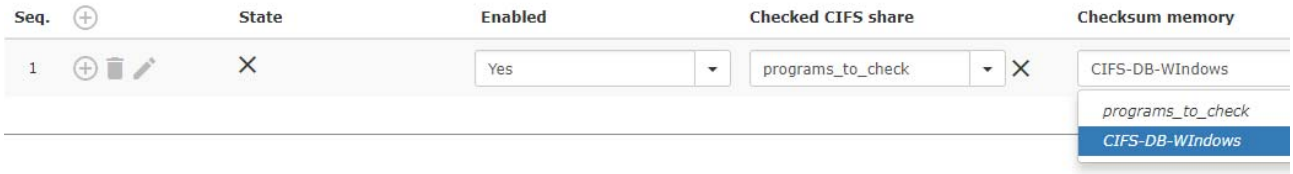


Figure 16-8 Creating a check sequence and selecting shares

Proceed as follows to create and configure a check sequence:

- Go to **CIFS Integrity Monitoring >> CIFS Integrity Check** (*Settings tab*).
- **Checking of Shares** section: Click on the ⊕ icon to create a new check sequence.
- Select the share to be checked from the drop-down list.
- Select the share to be used as the checksum memory from the drop-down list.
- Click on the ✎ icon to configure the parameters for a check sequence.

The parameters are all preset to defaults on the *Checked Share* tab. If need be, however, you can make changes here.

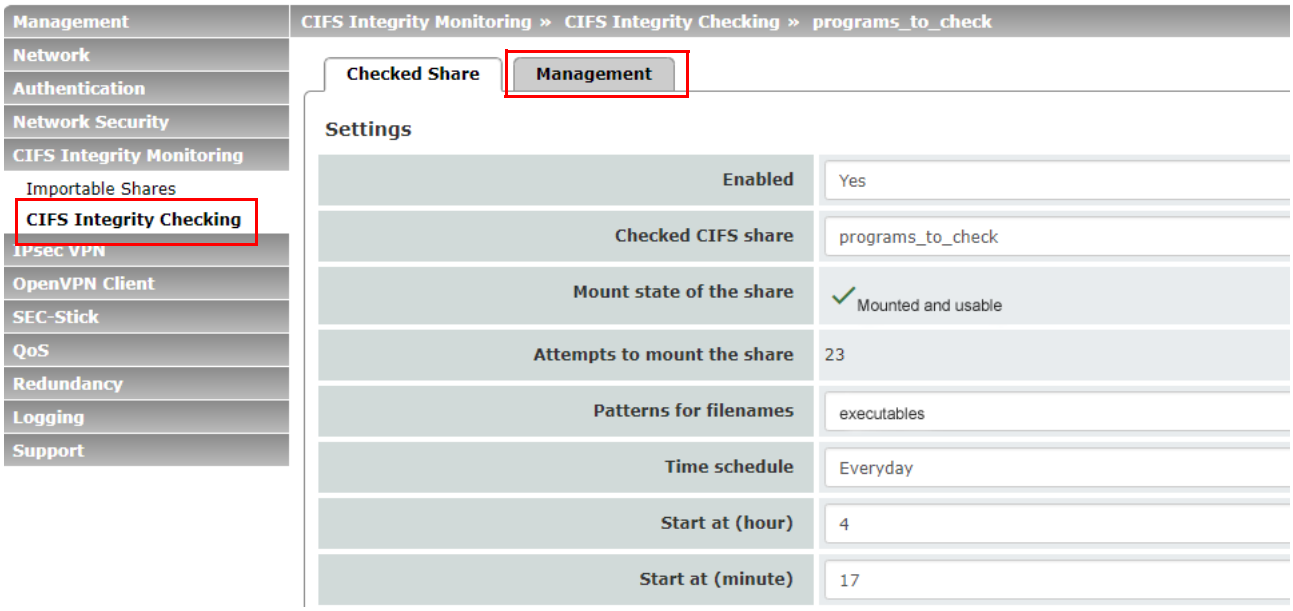


Figure 16-9 Parameter settings for checking the share

16.9 Initializing the integrity database

If a share to be checked is reconfigured, a corresponding integrity database must be created. This integrity database is used as the basis for comparison when checking the share regularly. It stores the checksums for all of the files to be monitored. The integrity database itself is signed with the integrity certificate to protect it against manipulation.

The integrity database is initialized on the *Management* tab.

First run a check to determine whether the mGuard device has read access to all of the files and directories on the monitored share (*Start an access check*).

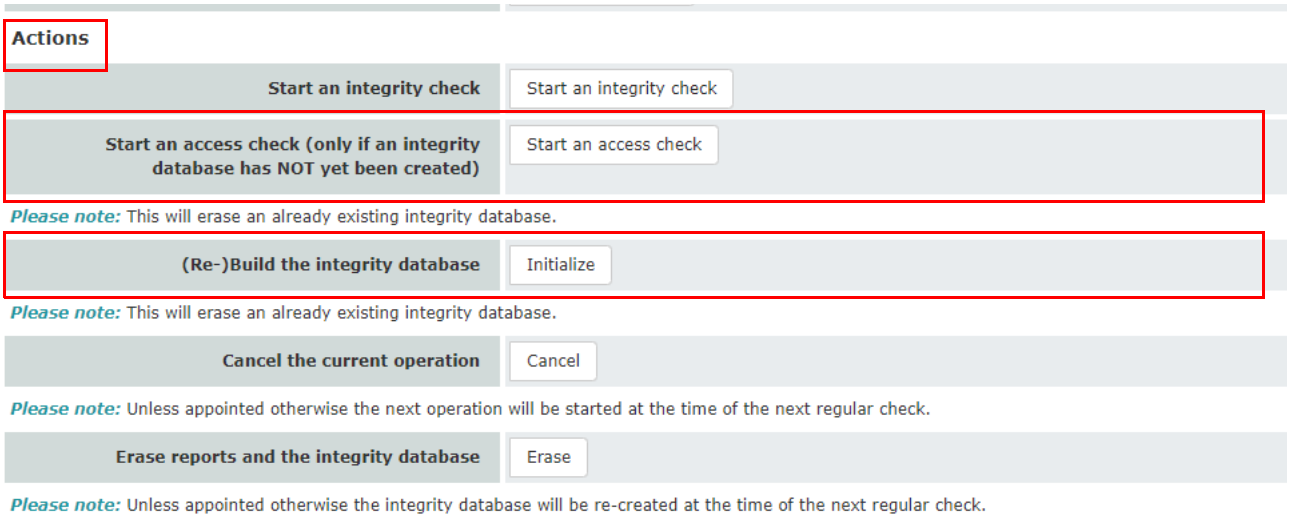


Figure 16-10 Preparing and starting an integrity check

Proceed as follows to (re)initialize the integrity database:

- Go to **CIFS Integrity Monitoring >> CIFS Integrity Check** (*Settings tab*).
- In the **Checking of Shares** section, click on the icon to configure check sequence parameters.
- The parameters are all preset to defaults on the *Checked Share* tab. If need be, you can make changes here.
- Switch to the *Management* tab.
- Click the **Start an access check** button (see Table 16-1).

⇒ The system checks to determine whether the access privileges required for the check are in place.

- If the privileges are in place, click on the **Initialize** button (see Table 16-1).

⇒ The integrity database is created and then used as a reference for further checks.

16.10 Options for actions when creating an integrity database

The actions that you can carry out as part of the CIFS Integrity Monitoring are briefly described in Table 16-1.

For a precise description, see also the mGuard firmware manual, available at phoenixcontact.net/products or help.mguard.com.

Table 16-1 Preparing and starting an integrity check – description of functions

Function name	Description
Start an integrity check	Clicking on the *Start an integrity check* button starts the integrity check. The result of the check can be viewed in the report by clicking on the *Download report* button.
Start an access check **(only if an integrity database has NOT yet been created)**	**NOTE: Any existing integrity database will be deleted.** Click on the *Start an access check* button to check whether there are files present on the imported share that the mGuard device cannot access. This prevents a more comprehensive creation of the integrity database from being aborted due to lack of the proper access permissions. The result of the check can be viewed in the report by clicking on the *Download report* button.
(Re-)Build the integrity database	**NOTE: Any existing integrity database will be deleted.** The mGuard device creates a database with checksums so that it can determine later whether files have been changed. A change to executable files indicates a virus. If files have been changed, rebuilt, or deleted intentionally, a new database must be created by clicking on the *Initialize* button in order to prevent false alarms. The creation of an integrity database is also recommended if shares have been newly set up. Otherwise, an integrity database is set up during the first scheduled check instead of a check being performed (if an access check was not performed first).
Cancel the current operation	Click on the Cancel button to stop the integrity check.
Erase reports and the integrity database	**NOTE: Any existing integrity database will be deleted.** Click on the *Erase* button to delete all existing reports/databases. A new integrity database must be created/initialized for any further integrity checks. This can be initiated by clicking on the *Initialize* button. Otherwise, a new integrity database is generated automatically at the next scheduled check (if an access check was not performed first). This procedure is not visible.

16.11 Access check performed successfully

If the access check was performed successfully, the following message displays (see Figure 16-11).

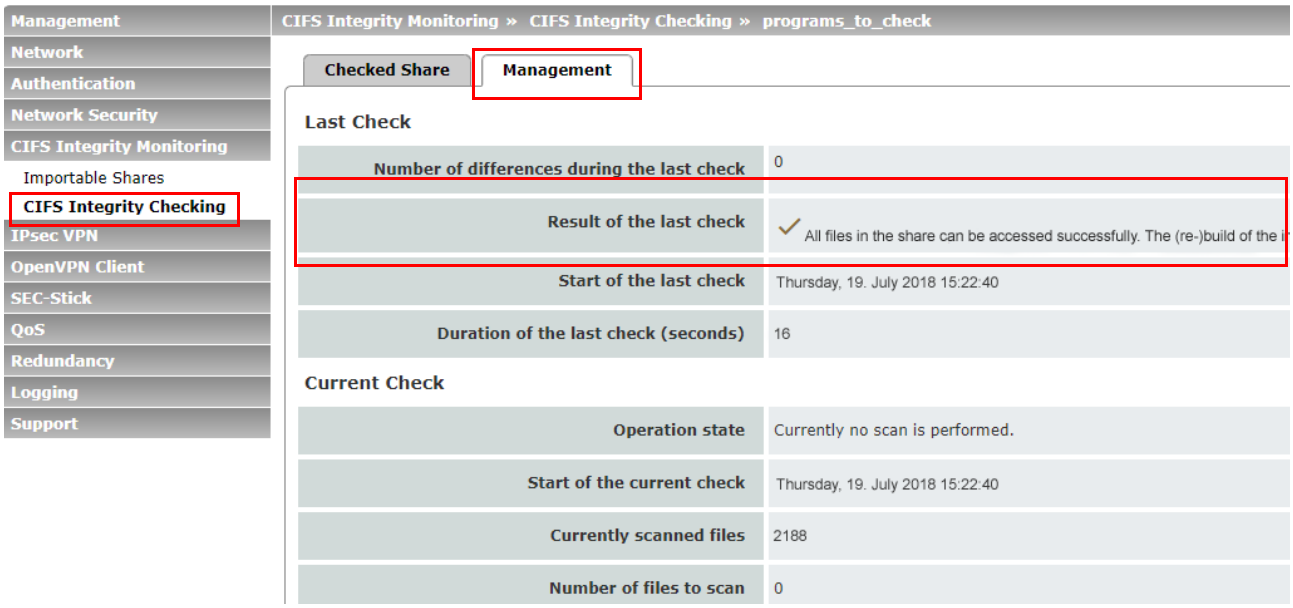


Figure 16-11 Access check successful

⇒ Once an access check has been successfully run, the integrity database can be (re)generated using the "*Initialize*" button under "*(Re-)Build the integrity database*".

16.12 Integrity database build successful

If the integrity database build was successful, the following image is displayed (see Figure 16-12).

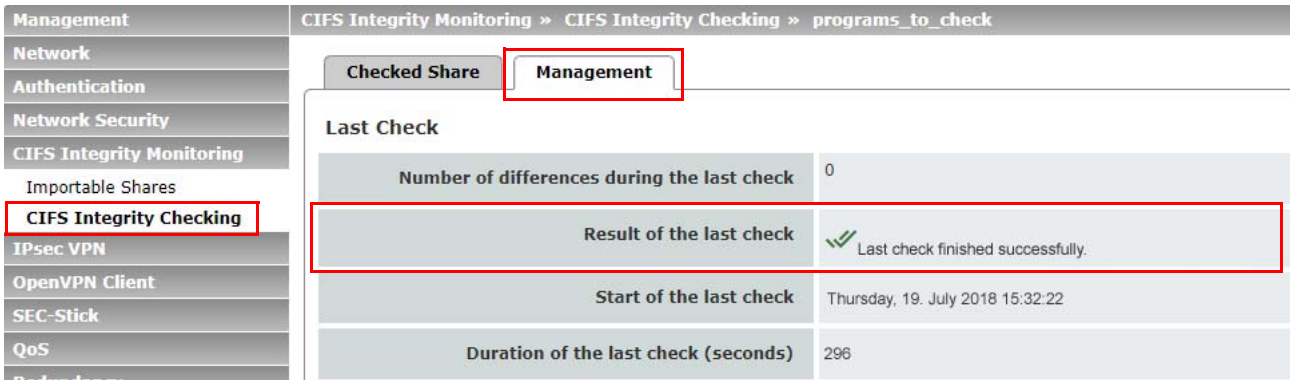


Figure 16-12 Successfully built integrity database

⇒ The integrity database has now been created. The consistency check is then done manually or automatically, depending on the configured time interval.

16.13 Missing access rights (read/write privileges)

If the mGuard device is denied access to any files/directories, the following error message appears.

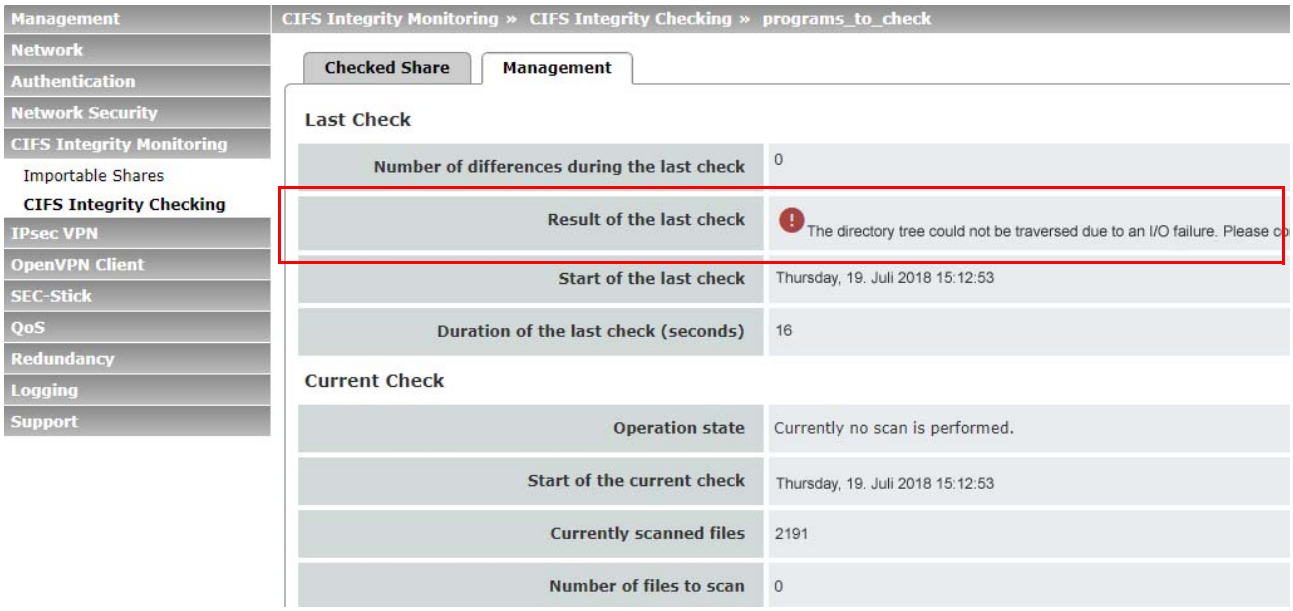


Figure 16-13 Access to files/directories failed

The directories or files in question are listed in the check report. This report is located on the checked PC and can be downloaded there or via the mGuard device's web-based management.

Example:

```
/var/cic/mnt/MAIv042835620-memory/integrity-check-log.txt
START_OF_LOG 2aa83b0b-6484-1787-a2d9-000cbe040098 Thu Jul 19
15:12:53 2018
SUBJECT check-access name=zu-pruefende-Programme
DIR_TRAVERSAL_ERR errno=13 syscall=readdir error="Permission
denied" path=Gemeinsame Dateien type=d
DIR_TRAVERSAL_ERR errno=13 syscall=readdir error="Permission
denied" path=Windows NT/ZubehÃ¶r type=d
ACCESS_CHECK_FAILED
END_OF_LOG
```

Figure 16-14 Example: Entry in report for failed read rights

In this case, Windows prevents access to the following directories:

– *Common Files*
– *Windows NT/Accessories*

16.14 Excluding files and directories from the check

If access to one of more files/directories is not possible, they can be excluded from the check.

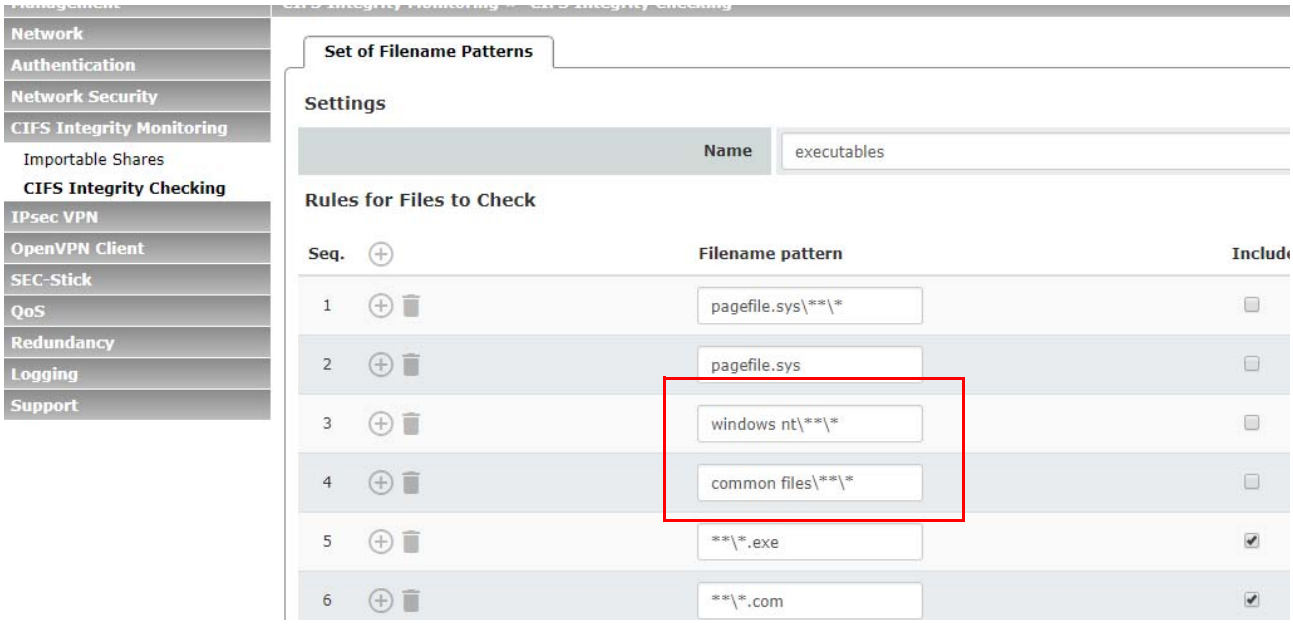


Figure 16-15 Excluding directories from the check

See also Section 16.7, “Specifying the files to check”

Directories that need to be excluded must be inserted in the table in a position before the first ***.

16.15 Performing a CIFS integrity check

Once the integrity database has been successfully created, an integrity check can be performed. This can either be done

– manually via the web-based management or
– via scheduling (see Section 16.8, “Creating check sequences”).

For a description of all of the configuration parameters, see the mGuard firmware manual available at phoenixcontact.net/products or help.mguard.com.

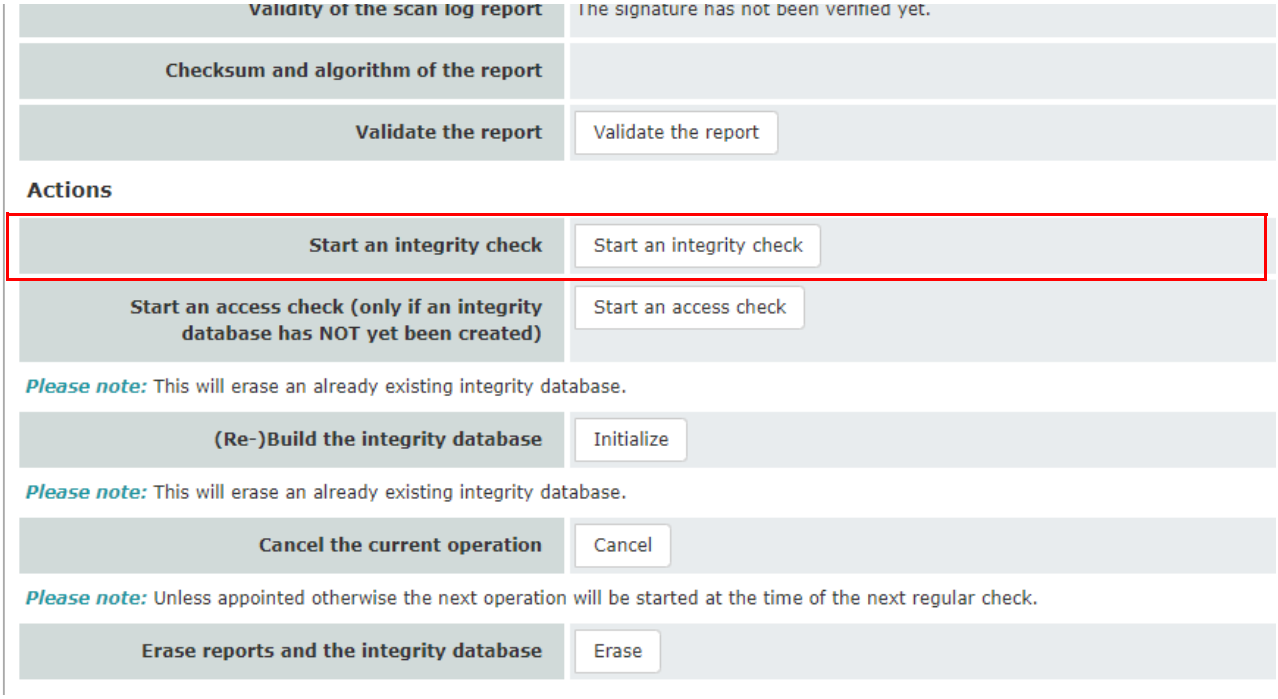


Figure 16-16 Performing an integrity check

Procedure

- Go to **CIFS Integrity Monitoring >> CIFS Integrity Check** (*Settings tab*).
- In the **Checking of Shares** section, click on the icon to configure check sequence parameters.
- The parameters are all preset to defaults on the *Checked Share* tab. If need be, you can make changes here.
- Switch to the *Management* tab.
- Click on the **Start an integrity check** button (see Table 16-1).

⇒ The result of the current check is displayed in the **Current Check** section. A check report is generated.

- Click on the **Validate the report** button to verify the integrity of the check report.
- Click on the **Download report** button to download and analyze the check report.

Please observe the following notes

General terms and conditions of use for technical documentation

Phoenix Contact reserves the right to alter, correct, and/or improve the technical documentation and the products described in the technical documentation at its own discretion and without giving prior notice, insofar as this is reasonable for the user. The same applies to any technical changes that serve the purpose of technical progress.

The receipt of technical documentation (in particular user documentation) does not constitute any further duty on the part of Phoenix Contact to furnish information on modifications to products and/or technical documentation. You are responsible to verify the suitability and intended use of the products in your specific application, in particular with regard to observing the applicable standards and regulations. All information made available in the technical data is supplied without any accompanying guarantee, whether expressly mentioned, implied or tacitly assumed.

In general, the provisions of the current standard Terms and Conditions of Phoenix Contact apply exclusively, in particular as concerns any warranty liability.

Phoenix Contact reserves the right to register its own intellectual property rights for the product identifications of Phoenix Contact products that are used here. Registration of such intellectual property rights by third parties is prohibited.

Other product identifications may be afforded legal protection, even where they may not be indicated as such.

How to contact us

Internet

Up-to-date information on Phoenix Contact products and our Terms and Conditions can be found on the Internet at:
phoenixcontact.com

Make sure you always use the latest documentation.
It can be downloaded at:
phoenixcontact.net/products

Subsidiaries

If there are any problems that cannot be solved using the documentation, please contact your Phoenix Contact subsidiary.
Subsidiary contact information is available at phoenixcontact.com.

Published by

PHOENIX CONTACT GmbH & Co. KG
Flachsmarktstraße 8
32825 Blomberg
GERMANY

PHOENIX CONTACT Development and Manufacturing, Inc.
586 Fulling Mill Road
Middletown, PA 17057
USA

Should you have any suggestions or recommendations for improvement of the contents and layout of our manuals, please send your comments to:
tecdoc@phoenixcontact.com