# D-Link®
**Building Networks for People**

# Network Security Firewall
# CLI Reference Guide

## NetDefendOS

**Ver.** 12.00.13

Security

# NETDEFEND

# CLI Reference Guide

## *DFL-260E/860E/870/1660/2560/2560G*

## *NetDefendOS version 12.00.13*

# CLI Reference Guide
**DFL-260E/860E/870/1660/2560/2560G**

## NetDefendOS version 12.00.13

Published 2018-11-07

Copyright © 2018

# Table of Contents

# List of Examples

# Preface

## Audience

The target audience for this reference guide is:

- Administrators that are responsible for configuring and managing the D-Link Firewall.

- Administrators that are responsible for troubleshooting the D-Link Firewall.

This guide assumes that the reader is familiar with the D-Link Firewall, and has the necessary basic knowledge in network security.

## Notation

The following notation is used throughout this reference guide when specifying the options of a command:

| | |
|---|---|
| **Angle brackets <name> or -option=<description>** | Used for specifying the *name* of an option or a description of a value. |
| **Square brackets [option] or -option[=value]** | Used for specifying that an option or a value for an option is *optional* and can be omitted. |
| **Curly brackets {value1 \| value2 \| value3}** | Used for specifying the *available values* for an option. |
| **Ellipsis ...** | Used for specifying that *more than one* value can be specified for the option. |

---

**Example 1. Command Option Notation**

One of the usages for the **help** command looks like this:

```
help -category={COMMANDS | TYPES} [<Topic>]
```

This means that help has an option called `category` which has two possible values which are *COMMANDS* and *TYPES*. There is also an optional option called `Topic` which in this case is a search string used to specify what help topic to display. Since the topic is optional, it is possible to exclude it when running the command.

Both of the following examples are valid for the usage described above:

```
gw-world:/> help -category=COMMANDS
gw-world:/> help -category=COMMANDS activate
```

The usage for the **routes** command is:

```
routes [-all] [-switched] [-flushl3cache[=<percent>]] [-num=<n>]
       [-nonhost] [-tables] [-lookup=<ip address>] [-verbose]
       [-setmtu=<mtu>] [-cacheinfo] [<table name>]...
```

None of the options of this command are mandatory. The `flushl3cache` option also has an optional value. This is because that option has a default value, *100*, which will be used if no value

---

is specified.

The following two examples will yield the same result:

```
gw-world:/> routes -flushl3cache=100
gw-world:/> routes -flushl3cache
```

Because the `table name` option is followed by ellipses it is possible to specify more than one routing table. Since `table name` is optional as well, the user can specify zero or more policy-based routing tables.

```
gw-world:/> routes Virroute Virroute2
```

# Chapter 1: Introduction

This guide is a reference for all commands and configuration object types that are available in the command line interface for NetDefendOS.

The CLI is case-sensitive. However, the tab-completion feature of the CLI does not require the correct case to perform completion and will alter the typed case if it is required.

## 1.1. Basic CLI Usage

### Entering Commands

The commands described in this guide can be run by typing the command name at the system prompt and then pressing the return key. Many commands require options to be set to run. If a required option is missing a brief syntax help will be displayed.

### User roles

Some commands and options cannot be used unless the logged-in user has administrator privileges. This is indicated in this guide by a note following the command or **Admin only** written next to an option.

### Function keys

There are a number of function keys that are used in the CLI.

| | |
|---|---|
| **Backspace** | Delete the character to the left of the cursor. |
| **Tab** | Complete current word. |
| **Ctrl-A or Home** | Move the cursor to the beginning of the line. |
| **Ctrl-B or Left Arrow** | Move the cursor one character to the left. |

| | |
|---|---|
| **Ctrl-C** | Clear line or cancel page view if more than one page of information is shown. |
| **Ctrl-D or Delete** | Delete the character to the right of the cursor. |
| **Ctrl-E or End** | Move the cursor to the end of the line. |
| **Ctrl-F or Right Arrow** | Move the cursor one character to the right. |
| **Ctrl-K** | Delete from the cursor to the end of the line. |
| **Ctrl-N or Down Arrow** | Show the next entry in the command history. |
| **Ctrl-P or Up Arrow** | Show the previous entry in the command history. |
| **Ctrl-T** | Transpose the current and the previous character. |
| **Ctrl-U** | Delete from the cursor to the beginning of line. |
| **Ctrl-W** | Delete word backwards. |

### The CLI History

Every time a command is run, the command line is added to a history list. The up and down arrow keys are used to access previous command lines (up arrow for older command lines and down arrow to move back to a newer command line). See also Section 2.4.4, "history".

---

**Example 1.1. The CLI History**

Using the command line history via the arrow keys:

```
gw-world:/> show Address
gw-world:/> (up arrow)
gw-world:/> show Address (the previous commandline is displayed)
```

---

### Adding and Removing IP Address Group Members

With IP address groups, it is often useful to be able to add new members to a group or remove existing group members. This is easily done with the web interface which provides an intuitive display showing the available objects and the objects in the group. It can also be done with the CLI but requires a special command syntax.

Suppose there already exists an *IP4Group* object called *my_ip_group*. It has three member *IP4Address* objects called *my_ip_1*, *my_ip_2* and *my_ip_3*. Suppose that the object *my_ip_2* is to be removed from the group. The command would be:

```
gw-world:/> set Address IP4Group my_ip_group Members-=my_ip_2
```

The option **Members-=** can remove one or more members of the group. To add one or more members to a group, the option **Members+=** can be used. Suppose that the *IP4Address* objects *my_ip_4* and *my_ip_5* are to be added to the group. The command would be:

```
gw-world:/> set Address IP4Group my_ip4_group Members+=my_ip_4,my_ip_5
```

*14*

# 1.2. CLI Tab Completion

By using the tab function key in the CLI the names of commands, options, objects and object properties can be automatically completed. If the text entered before pressing tab only matches one possible item, e.g. "activate" is the only match for "acti", and a command is expected, the name will be autocompleted. Should there be more than one match the part common to all matches will be completed. At this point the user can either enter more characters or press tab again, which will display a list of the possible completions. This can also be done without entering any characters, but the resulting list might be long if there are many possible completions, e.g. all commands.

---

**Example 1.2. Tab completion**

An example of tab completion when using the **add** command:

```
gw-world:/> add Add (tab)
gw-world:/> add Address ("ress" was autocompleted)
gw-world:/> add Address i (tab)
gw-world:/> add Address IP4 ("IP4" was autocompleted)
gw-world:/> add Address IP4
        (tab, or double tab if IP4 were entered manually)
A list of all types starting with IP4 is listed.
gw-world:/> add Address IP4a (tab)
gw-world:/> add Address IP4Address ("Address" was autocompleted)
gw-world:/> add Address IP4Address example_ip a (tab)
gw-world:/> add Address IP4Address example_ip Address=
        ("Address=" was autocompleted)
gw-world:/> add Address IP4Address example_ip Address=1.2.3.4
```

Tab completion of references:

```
gw-world:/> set Address IP4Group examplegroup Members= (tab, tab)
A list of valid objects is displayed.
gw-world:/> set Address IP4Group examplegroup Members=e (tab)
gw-world:/> set Address IP4Group examplegroup Members=example_ip
("example_ip" was autocompleted)
```

---

# 1.2.1. Inline Help with Tab Completion

It is possible to get help about available properties of configuration objects while a command line is being typed by using the *?* character. Write *?* instead of a property name and press tab and a help text for the available properties is shown. If *?* is typed in stead of a property value and tab is pressed a help text for that property which contains more information such as data type, default value, etc. is displayed.

---

**Example 1.3. Inline help**

Get inline help for all properties of an IP4Address:

```
gw-world:/> set IP4Address example_ip ? (tab)
A help text describing all available properties is displayed.
```

Getting inline help for the Address property:

---

```
gw-world:/> set IP4Address example_ip Address=? (tab)
A more detailed help text about Address is displayed.
```

## 1.2.2. Autocompleting Current and Default Values

Another special character that can be used together with tab completion is the period "."
character. If "." is entered instead of a property value and tab is pressed it will be replaced by the
current value of that property. This is useful when editing an existing list of items or a long text
value.

The "<" character before a tab can be used to automatically fill in the default value for a
parameter if no value has yet been set. If the "." character is used, all possible values will be
shown and these can then be edited with the back arrrow and backspace keys.

**Example 1.4. Edit an Existing Property Value in the CLI**

Edit the current value:

```
gw-world:/> add IP4Address example_ip Address=1.2.3.4
gw-world:/> set IP4Address example_ip Address=. (tab)
gw-world:/> set IP4Address example_ip Address=1.2.3.4
        (the value was inserted)
The value can now be edited by using the arrow keys or backspace.

gw-world:/> set IP4Group examplegroup Members=ip1,ip2,ip3,ip5
gw-world:/> set IP4Group examplegroup Members=. (tab)
gw-world:/> set IP4Group examplegroup Members=ip1,ip2,ip3,ip5
(the value was inserted)
It is now possible to add or remove a member to the list without
having to enter all the other members again.
```

Edit the default value:

```
gw-world:/> add LogReceiverSyslog example Address=example_ip
        LogSeverity=. (tab)
gw-world:/> add LogReceiverSyslog example Address=example_ip
  LogSeverity=Emergency,Alert,Critical,Error,Warning,Notice,Info
```

It is now easy to remove a log severity.

## 1.2.3. Configuration Object Type Categories

Some object types are grouped together in a category in the CLI. This only matters when using
tab completion as they are used to limit the number of possible completions when tab
completing object types. The category can always be omitted when running commands if the
type name is entered manually.

**Example 1.5. Using categories with tab completion**

Accessing an IP4Address object with the use of categories:

```
gw-world:/> show ad (tab)
gw-world:/> show Adress (the category is autocompleted)
```

```
gw-world:/> show Adress ip4a (tab)
gw-world:/> show Adress IP4Address (the type is autocompleted)
gw-world:/> show Adress IP4Address example_ip
```

Accessing an IP4Address object without the use of categories:

```
gw-world:/> show IP4Address example_ip
```

# 1.3. CLI Help Options

## 1.3.1. Help for Commands

There are two ways of getting help about a command. A brief help is displayed if the command name is typed followed by -? or -h. This applies to all commands and is therefore not listed in the option list for each command in this guide. Using the **help** command gives a more detailed help corresponding to the information found in this guide. In most cases it is possible to simply type **help** followed by the command name to get the full help. See Section 2.4.3, "help" for a more detailed description. To list the available commands, just type **help** and press return.

---

**Example 1.6. Help for Commands**

Brief help for the **activate** command:

```
gw-world:/> activate -?
gw-world:/> activate -h
```

Full help for **activate**:

```
gw-world:/> help activate
```

Help for the **arp** command. Arp is also the name of a configuration object type, so it is necessary to specify that the help text for the command should be displayed:

```
gw-world:/> help -category=COMMANDS arp
```

List all available commands:

```
gw-world:/> help
```

---

## 1.3.2. Help for Object Types

To get help about configuration object types, use the **help** command. It is also possible to get information about each property in an object type, such as data type, default value, etc. by entering the ? character when entering the value of a property and pressing tab. More on this in Section 1.2.1, "Inline Help with Tab Completion".

---

**Example 1.7. Help for Object Types**

Full help for IP4Address:

```
gw-world:/> help IP4Address
```

Help for the ARP configuration object type, which collides with the **arp** command:

```
gw-world:/> help -category=TYPES ARP
```

---

# Chapter 2: Command Reference

## 2.1. Configuration

### 2.1.1. activate

Activate changes.

**Description**

Activate the latest changes.

This will issue a reconfiguration, using the new configuration. If the reconfiguration is successful a **commit** command must be issued within the configured timeout interval in order to save the changes to media. If not, the system will revert to using the previous version of the configuration.

**Usage**

```
activate
```

> ***Note***
> *Requires Administrator privileges.*

### 2.1.2. add

Create a new object.

### Description

Create a new object and add it to the configuration.

Specify the type of object you want to create and the identifier, if the type has one, unless the object is identified by an index. Set the properties of the object by writing the propertyname equals (=) and then the value. An optional category can be specified for some object types when using tab completion.

If a mandatory property isn't specified a list of errors will be shown after the object is created. If an invalid property or value type is specified or if the identifier is missing the command will fail and not create an object.

Adjustments can be made after the object is created by using the **set** command.

---

**Example 2.1. Create a new object**

```
Add objects with an identifier property (not index):
 gw-world:/> add Address IP4Address example_ip Address=1.2.3.4
 Comments="This is an example"
 gw-world:/> add IP4Address example_ip2 Address=2.3.4.5
Add an object with an index:
 gw-world:/main> add Route Interface=lan
Add an object without identifier:
 gw-world:/> add DynDnsClientDyndnsOrg DNSName=example Username=example
```

---

### Usage

```
add [<Category>] <Type> [<Identifier>] [-force] [-silent]
    [<key-value pair>]...
```

### Options

| | |
|---|---|
| **-force** | Add object, even if it has errors. |
| **-silent** | Do not show any errors. |
| **<Category>** | Category that groups object types. |
| **<Identifier>** | The property that identifies the configuration object. May not be applicable depending on the specified <Type>. |
| **<key-value pair>** | One or more property-value pairs, i.e. <property name>=<value> or <property name>="<value>". |
| **<Type>** | Type of configuration object to perform operation on. |

---

***Note***
*Requires Administrator privileges.*

---

*21*

## 2.1.3. cancel

Cancel ongoing commit.

### Description

Cancel commit operation immediately, without waiting for the timeout.

### Usage

```
cancel
```

***Note***
*Requires Administrator privileges.*

## 2.1.4. cc

Change the current context.

### Description

Change the current configuration context.

A context is a group of objects that are dependent on and grouped by a parent object. Many objects lie in the "root" context and do not have a specific parent. Other objects, e.g. User objects lie in a sub-context (or child context) of the root - in this case in a LocalUserDatabase. In order to add or modify users you have to be in the correct context, e.g. a LocalUserDatabase called "exampledb". Only objects in the current context can be accessed.

**Example 2.2. Change context**

```
Change to a sub/child context:
 gw-world:/> cc LocalUserDatabase exampledb
 gw-world:/exampledb>
Go back to the parent context:
 gw-world:/ospf1/area1> cc ..
 gw-world:/ospf1> cc ..
 gw-world:/>
Go back to the root context:
 gw-world:/ospf1/area1> cc
 gw-world:/>
or
 gw-world:/ospf1/area1> cc /
 gw-world:/>
```

### Usage

```
cc [<Category>] <Type> <Identifier>
```

*22*

Change the current context.

```
cc -print
```

Print the current context.

```
cc
```

Change to root context (same as "cc /").

**Options**

| | |
|---|---|
| **-print** | Print the current context. |
| **<Category>** | Category that groups object types. |
| **<Identifier>** | The property that identifies the configuration object. May not be applicable depending on the specified <Type>. |
| **<Type>** | Type of configuration object to perform operation on. |

## 2.1.5. commit

Save new configuration to media.

### Description

Save the new configuration to media. This command can only be issued after a successful activate command.

### Usage

```
commit
```

*Note*
*Requires Administrator privileges.*

## 2.1.6. delete

Delete specified objects.

### Description

Delete the specified object, removing it from the configuration.

Add the force flag to delete the object even if it is referenced by other objects or if it is a context that has child objects that aren't deleted. This may cause objects referring to the specified object or one of its children to get errors that must be corrected before the configuration can be

activated.

See also: **undelete**

---

**Example 2.3. Delete an object**

```
Delete an unreferenced object:
 gw-world:/> delete Address IP4Address example_ip
Delete a referenced object:
(will cause error in examplerule)
 gw-world:/> set IPRule examplerule SourceNetwork=examplenet
 gw-world:/> delete Address IP4Address examplenet -force
```

---

**Usage**

```
delete [<Category>] <Type> [<Identifier>] [-force]
```

**Options**

| | |
|---|---|
| **-force** | Force object to be deleted even if it's used by other objects or has children. |
| **<Category>** | Category that groups object types. |
| **<Identifier>** | The property that identifies the configuration object. May not be applicable depending on the specified <Type>. |
| **<Type>** | Type of configuration object to perform operation on. |

---

> ***Note***
> *Requires Administrator privileges.*

---

## 2.1.7. pskgen

Generate random pre-shared key.

### Description

Generate a pre-shared key of specified size, containing randomized key data. If a key with the specified name exists, the existing key is modified. Otherwise a new key object is created.

### Usage

```
pskgen <Name> [-comments=<String>] [-size={64 | 128 | 256 | 512 |
        1024 | 2048 | 4096}]
```

**Options**

| | |
|---|---|
| **-comments=<String>** | Comments for this key. |
| **-size={64 \| 128 \| 256 \| 512 \| 1024 \| 2048 \| 4096}** | Number of bits of data in the generated key. (Default: 64) |
| **<Name>** | Name of key. |

> ***Note***
> *Requires Administrator privileges.*

# 2.1.8. reject

Reject changes.

## Description

Reject the changes made to the specified object by reverting to the values of the last committed configuration.

All changes made to the object will be lost. If the object is added after the last commit, it will be removed.

To reject the changes in more than one object, use either the `-recursive` flag to delete a context and all its children recursively or the `-all` flag to reject the changes in *all* objects in the configuration.

See also: **activate**, **commit**

**Example 2.4. Reject changes**

```
Reject changes in individual objects:
 gw-world:/> set Address IP4Address example_ip
 Comments="This comment will be rejected"
 gw-world:/> reject Address IP4Address example_ip
 gw-world:/> add Address IP4Address example_ip2 Address=1.2.3.4
 Comments="This whole object will be removed"
 gw-world:/> reject Address IP4Address example_ip2
Reject changes recursively:
(will reject changes in the user database and all users)
 gw-world:/exampledb> set User user1 Comments="Something"
 gw-world:/exampledb> set User user2 Comments="that will be"
 gw-world:/exampledb> set User user3 Comments="rejected"
 gw-world:/exampledb> cc ..
 gw-world:/> reject LocalUserDatabase exampledb -recursive
Reject all changes:
 gw-world:/anycontext> reject -all
All changes since the last commit will be rejected:
(example_ip will be removed since it is newly added)
 gw-world:/> add IP4Address example_ip Address=1.2.3.4
 gw-world:/> delete IP4Address example_ip
 gw-world:/> reject IP4Address example_ip
```

**Usage**

```
reject [<Category>] <Type> [<Identifier>] [-recursive]
```

Reject changes made to the specified object.

```
reject -all
```

Reject all changes in the configuration.

**Options**

| | |
|---|---|
| **-all** | Reject all changes in the configuration. |
| **-recursive** | Recursively reject changes. |
| **<Category>** | Category that groups object types. |
| **<Identifier>** | The property that identifies the configuration object. May not be applicable depending on the specified <Type>. |
| **<Type>** | Type of configuration object to perform operation on. |

***Note***
*Requires Administrator privileges.*

## 2.1.9. reset

Reset unit configuration and/or binaries.

**Description**

Reset configuration or binaries to factory defaults.

**Usage**

```
reset -configuration
```

Reset the configuration to factory defaults.

```
reset -unit
```

Reset the unit to factory defaults.

**Options**

| | |
|---|---|
| **-configuration** | Reset configuration to factory default. |
| **-unit** | Reset unit to factory defaults. |

***Note***
*Requires Administrator privileges.*

# 2.1.10. set

Set property values.

## Description

Set property values of configuration objects.

Specify the type of object you want to modify and the identifier, if the type has one. Set the properties of the object by writing the propertyname equals (=) and then the value. An optional category can be specified for some object types when using tab completion.

If a mandatory property hasn't been specified or if a property has an error a list of errors will be shown after the specified properties have been set. If an invalid property or value type is specified the command will fail and not modify the object.

See also: **add**

**Example 2.5. Set property values**

```
Set properties for objects that have an identifier property:
 gw-world:/> set Address IP4Address example_ip Address=1.2.3.4
 Comments="This is an example"
 gw-world:/> set IP4Address example_ip2 Address=2.3.4.5
 Comments=comment_without_whitespace
 gw-world:/main> set Route 1 Comment="A route"
 gw-world:/> set IPRule 12 Index=1
Set properties for an object without identifier:
 gw-world:/> set DynDnsClientDyndnsOrg Username=example
```

## Usage

```
set [<Category>] <Type> [<Identifier>] [-disable] [-enable]
    [-force] [<key-value pair>]...
```

## Options

**-disable**                          Disable object. This option is not available if the object is already disabled.

**-enable**                           Enable object. This option is not available if the object is already enabled.

**-force**                            Set values, even if they contain errors.

**<Category>**                        Category that groups object types.

**<Identifier>**                      The property that identifies the configuration

|  |  |
|---|---|
|  | object. May not be applicable depending on the specified <Type>. |
| **<key-value pair>** | One or more property-value pairs, i.e. <property name>=<value> or <property name>="<value>". |
| **<Type>** | Type of configuration object to perform operation on. |

> ***Note***
> *Requires Administrator privileges.*

## 2.1.11. show

Show objects.

### Description

Show objects.

Show the properties of a specified object. There are a number of flags that can be specified to show otherwise hidden properties. To show a list of object types and categories available in the current context, just type **show**. Show a table of all objects of a type by specifying a type or a category. Use the `-errors` or `-changes` flags to show what objects have been changed or have errors in the configuration.

When showing a table of all objects of a certain type, the status of each object since the last time the configuration was committed is indicated by a flag. The flags used are:

**-**    The object is deleted.

**o**    The object is disabled.

**!**    The object has errors.

**+**    The object is newly created.

**\***    The object is modified.

Additional flags:

**D**    The object has dynamic properties which are updated by the system.

When listing categories and object types, categories are indicated by [] and types where objects may be contexts by /.

---

**Example 2.6. Show objects**

```
Show the properties of an individual object:
 gw-world:/> show Address IP4Address example_ip
 gw-world:/main> show Route 1
 gw-world:/> show Client DynDnsClientDyndnsOrg
Show a table of all objects of a type and a selection of their
```

---

```
properties as well as their status:
 gw-world:/> show Address IP4Address
 gw-world:/> show IP4Address
Show a table of all objects for each type in a category:
 gw-world:/> show Address
Show objects with changes and errors:
 gw-world:/> show -changes
 gw-world:/> show -errors
Show what objects use (refer to) a certain object:
 gw-world:/> show Address IP4Address example_ip -references
```

**Usage**

```
show
```

Show the types and categories available in the current context.

```
show [<Category>] [<Type> [<Identifier>]] [-disabled] [-references]
```

Show an object or list a type or category.

```
show -errors [-verbose]
```

Show all errors.

```
show -changes
```

Show all changes.

**Options**

| | |
|---|---|
| **-changes** | Show all changes in the current configuration. |
| **-disabled** | Show disabled properties. |
| **-errors** | Show all errors in the current configuration. |
| **-references** | Show all references to this object from other objects. |
| **-verbose** | Show error details. |
| **<Category>** | Category that groups object types. |
| **<Identifier>** | The property that identifies the configuration object. May not be applicable depending on the specified <Type>. |
| **<Type>** | Type of configuration object to perform operation on. |

## 2.1.12. undelete

Restore previously deleted objects.

### Description

Restore a previously deleted object.

This is possible as long as the **activate** command has not been called.

See also: **delete**

---

**Example 2.7. Undelete an object**

```
Undelete an unreferenced object:
 gw-world:/> delete Address IP4Address example_ip
 gw-world:/> undelete Address IP4Address example_ip
Undelete a referenced object:
(will remove the error in examplerule)
 gw-world:/> set IPRule examplerule SourceNetwork=examplenet
 gw-world:/> delete Address IP4Address examplenet -force
 gw-world:/> undelete Address IP4Address examplenet
```

---

### Usage

```
undelete [<Category>] <Type> [<Identifier>]
```

### Options

**<Category>**                      Category that groups object types.

**<Identifier>**                    The property that identifies the configuration
                                    object. May not be applicable depending on the
                                    specified <Type>.

**<Type>**                          Type of configuration object to perform operation
                                    on.

---

> ***Note***
> *Requires Administrator privileges.*

---

# 2.2. Runtime

## 2.2.1. about

Show copyright/build information.

### Description

Show copyright and build information.

### Usage

```
about
```

## 2.2.2. alarm

Show alarm information.

### Description

Show list of currently active alarms.

### Usage

```
alarm [-history] [-active]
```

### Options

**-active**                                Show the currently active alarms.

**-history**                               Show the 20 latest alarms.

## 2.2.3. appcontrol

Show application control status.

### Description

Browse the applications defined in the Application Control functionality. Saved browsing results as filters that can be later used to define IPPolicies.

### Usage

```
appcontrol
```

Show general information about application control system.

```
appcontrol -show_lists
```

List information about specified application.

```
appcontrol -delete_lists={ALL | <Integer>}
```

List information about specified application.

```
appcontrol <Name>
```

List information about specified application.

```
appcontrol -application=<String> [-save_list]
```

Define a filter selecting individual applications.

```
appcontrol -filter [-name=<String>] [-family=<String>]
           [-risk={VERY_LOW | LOW | MEDIUM | HIGH | VERY_HIGH}]
           [-tag=<String>] [-save_list]
```

Define a filter selecting families, tags, risks and a matching expression for the applications names.

**Options**

| | |
|---|---|
| **-application=<String>** | Exact application name. |
| **-delete_lists={ALL \| <Integer>}** | Free saved Strings. |
| **-family=<String>** | Application family. |
| **-filter** | Shows applications matching certain criteria. |
| **-name=<String>** | Application name (wildcards allowed). |
| **-risk={VERY_LOW \| LOW \| MEDIUM \| HIGH \| VERY_HIGH}** | Application risk level. |
| **-save_list** | Saved filter result. |
| **-show_lists** | List saved strings. |
| **-tag=<String>** | Application tag. |
| **<Name>** | Application name. |

## 2.2.4. arp

Show ARP entries for given interface.

**Description**

List the ARP cache entries of specified interfaces.

If no interface is given the ARP cache entries of all interfaces will be presented.

The presented list can be filtered using the *ip* and *hw* options.

**Usage**

```
arp
```

Show all ARP entries.

```
arp -show [<Interface>] [-ip=<pattern>] [-hw=<pattern>] [-num=<n>]
```

Show ARP entries.

```
arp -hashinfo [<Interface>]
```

Show information on hash table health.

```
arp -flush [<Interface>]
```

Flush ARP cache of specified interface.

```
arp -notify=<ip> [<Interface>] [-hwsender=<Ethernet Address>]
```

Send gratuitous ARP for IP.

**Options**

| | |
|---|---|
| **-flush** | Flush ARP cache of all specified interfaces. (Admin only) |
| **-hashinfo** | Show information on hash table health. |
| **-hw=<pattern>** | Show only hardware addresses matching pattern. |
| **-hwsender=<Ethernet Address>** | Sender ethernet address. |
| **-ip=<pattern>** | Show only IP addresses matching pattern. |
| **-notify=<ip>** | Send gratuitous ARP for <ip>. |
| **-num=<n>** | Show only the first <n> entries per interface. (Default: 20) |
| **-show** | Show ARP entries for given interface(s). |
| **<Interface>** | Interface name. |

## 2.2.5. arpsnoop

Toggle snooping and displaying of ARP requests.

**Description**

Toggle snooping and displaying of ARP queries and responses on-screen.

The snooped messages are displayed before the access section validates the sender IP addresses in the ARP data.

*33*

**Usage**

```
arpsnoop
```

Show snooped interfaces.

```
arpsnoop {ALL | NONE | <interface>} [-verbose]
```

Snoop specified interface.

**Options**

**-verbose**                              Verbose.

**{ALL | NONE | <interface>}**             Interface name.

> ***Note***
> *Requires Administrator privileges.*

## 2.2.6. ats

Show active ARP Transaction States.

**Description**

Show active ARP Transaction States.

**Usage**

```
ats [-num=<n>]
```

**Options**

**-num=<n>**                           Limit list to <n> entries. (Default: 20)

## 2.2.7. authagent

Shows the state of the Authentication Agents.

**Description**

Shows the state of the Authentication Agents.

**Usage**

```
authagent -version
```

Shows the state of the configured Authentication Agents including the protocol version.

```
authagent
```

Shows the state of the configured Authentication Agents.

```
authagent {ALL | <AuthAgent>}
```

Shows the state of the configured Authentication Agents.

```
authagent -reconnect {ALL | <AuthAgent>}
```

Closes the connection with the Agent and attempst to reconnect.

**Options**

| | |
|---|---|
| **-reconnect** | Closes the connection with the Agent and attempst to reconnect. (Admin only) |
| **-version** | Show protocol version. |
| **{ALL | <AuthAgent>}** | Authentication Agent name. |

## 2.2.8. authagentsnoop

Toggle snooping and displaying of Authentication Agents traffic.

**Description**

Toggle snooping and displaying of Authentication Agents queries and responses on-screen.

**Usage**

```
authagentsnoop
```

Show snooped Authentication Agents.

```
authagentsnoop {ALL | NONE | <AuthAgent>} [-verbose]
```

Snoop specified Authentication Agent.

**Options**

| | |
|---|---|
| **-verbose** | Verbose. |
| **{ALL | NONE | <AuthAgent>}** | Authentication Agent name. |

> ***Note***
> *Requires Administrator privileges.*

## 2.2.9. avcache

Control the anti-virus cache.

### Description

Show anti-virus cache statistics or remove all entries in it.

### Usage

```
avcache -clear
```

Remove all entries in the anti-virus cache.

```
avcache
```

Show anti-virus cache count.

#### Options

**-clear**                                  Remove all entries in the anti-virus cache.

## 2.2.10. blacklist

Blacklist.

### Description

Block and unblock hosts on the black and white list.

Note: Static blacklist hosts cannot be unblocked.

If *-force* is not specified, only the exact host with the service, protocol/port and destiny specified is unblocked.

---

**Example 2.8. Block hosts**

```
blacklist -show -black -listtime -info
blacklist -block 100.100.100.0/24 -serv=FTP -dest=50.50.50.1 -time=6000
```

---

### Usage

```
blacklist
```

Show the current blacklist and whitelist content.

```
blacklist -show [-num={ALL | <Integer>}] [-alerttype={IDS |
          HOST_THRESHOLD | NETWORK_THRESHOLD | CLI | REST |
```

```
            DOS_GENERAL | DOS_GEOIP | BOTNET | SCANNER | ALL}]
            [-creationtime] [-dynamic] [-listtime] [-info] [-black]
            [-white] [-all]
```

Show information about the blacklisted hosts.

```
blacklist -block <host> [-serv=<service>] [-prot={TCP | UDP | ICMP
            | OTHER | TCPUDP | ALL}] [-port=<port number>] [-dest=<ip
            address>] [-time=<seconds>]
```

Block specified netobject.

```
blacklist -unblock <host> [-serv=<service>] [-prot={TCP | UDP |
            ICMP | OTHER | TCPUDP | ALL}] [-port=<port number>]
            [-dest=<ip address>] [-force]
```

Unblock specified netobject.

```
blacklist -purge {IDS | HOST_THRESHOLD | NETWORK_THRESHOLD | CLI |
            REST | DOS_GENERAL | DOS_GEOIP | BOTNET | SCANNER}
```

Unblock all netobject of specific alert type.

**Options**

| | |
|---|---|
| **-alerttype={IDS \| HOST_THRESHOLD \| NETWORK_THRESHOLD \| CLI \| REST \| DOS_GENERAL \| DOS_GEOIP \| BOTNET \| SCANNER \| ALL}** | Alert types to show (default: ALL). |
| **-all** | Show all the information. |
| **-black** | Show blacklist hosts only. |
| **-block** | Block specified netobject. (Admin only) |
| **-creationtime** | Show creation time. |
| **-dest=<ip address>** | Destination address to block/unblock (ExceptExtablished flag is set on). |
| **-dynamic** | Show dynamic hosts only. |
| **-force** | Unblock all services for the host that matches to options. |
| **-info** | Show detailed information. |
| **-listtime** | Show time in list (for dynamic whitelist hosts). |
| **-num={ALL \| <Integer>}** | Maximum number of entries to show (default: 20). |
| **-port=<port number>** | Number of the port to block/unblock. |
| **-prot={TCP \| UDP \| ICMP \| OTHER \| TCPUDP \| ALL}** | Protocol to block/unblock. |
| **-purge** | Unblock all object of specific type. (Admin only) |
| **-serv=<service>** | Service to block/unblock. |
| **-show** | Show information about the blacklisted hosts. |
| **-time=<seconds>** | The time that the host will remain blocked. |
| **-unblock** | Unblock specified netobject. (Admin only) |

*37*

| | |
|---|---|
| **-white** | Show whitelist hosts only. |
| **<host>** | IP address range. |
| **{IDS \| HOST_THRESHOLD \| NETWORK_THRESHOLD \| CLI \| REST \| DOS_GENERAL \| DOS_GEOIP \| BOTNET \| SCANNER}** | Alert types to purge. |

## 2.2.11. buffers

List packet buffers or the contents of a buffer.

### Description

Lists the 20 most recently freed packet buffers, or in-depth information about a specific buffer.

### Usage

```
buffers
```

List the 20 most recently freed buffers.

```
buffers -recent
```

Decode the most recently freed buffer.

```
buffers <Num>
```

Decode buffer number <Num>.

### Options

| | |
|---|---|
| **-recent** | Decode most recently freed buffer. |
| **<Num>** | Decode given buffer number. |

## 2.2.12. cam

CAM table information.

### Description

Show information about the CAM table(s) and their entries.

### Usage

```
cam -num=<n>
```

Show CAM table information.

```
cam <Interface> [-num=<n>]
```

*38*

Show interface-specified CAM table information.

```
cam <Interface> [-flush]
```

Flush CAM table information of specified interface.

```
cam -flush
```

Flush CAM table information.

**Options**

| | |
|---|---|
| **-flush** | Flush CAM table. If interface is specified, only entries using this interface are flushed. (Admin only) |
| **-num=<n>** | Limit list to <n> entries per CAM table. (Default: 20) |
| **<Interface>** | Interface. |

## 2.2.13. certcache

Show the contents of the certificate cache.

**Description**

Show all certificates in the certificate cache.

**Usage**

```
certcache [-verbose] [-flush]
```

**Options**

| | |
|---|---|
| **-flush** | Flush certificate cache. |
| **-verbose** | Show verbose information. |

## 2.2.14. cfglog

Display configuration log.

**Description**

Display the log of the last configuration read attempt.

**Usage**

```
cfglog
```

## 2.2.15. connections

List current state-tracked connections.

### Description

List current state-tracked connections.

### Usage

```
connections -show [-extended] [-num=<n>] [-verbose]
            [-srciface=<interface>] [-destiface=<interface>]
            [-ipver={IPV6 | IPV4}] [-srcip=<ip address>]
            [-destip=<ip address>] [-protocol=<name/num>]
            [-srcport=<port>] [-destport=<port>]
            [-dataless=<bytes>] [-datamore=<bytes>]
```

List connections.

```
connections
```

Same as "connections -show".

```
connections -close [-all] [-srciface=<interface>]
            [-destiface=<interface>] [-ipver={IPV6 | IPV4}]
            [-srcip=<ip address>] [-destip=<ip address>]
            [-protocol=<name/num>] [-srcport=<port>]
            [-destport=<port>] [-dataless=<bytes>]
            [-datamore=<bytes>]
```

Close connections.

### Options

| | |
|---|---|
| **-all** | Mark all connections. |
| **-close** | Close all connections that match the filter expression. (Admin only) |
| **-dataless=<bytes>** | Filter on amount of data transferred below specified limit. Acceptable suffixes are 'k', 'M' and 'G'. |
| **-datamore=<bytes>** | Filter on amount of data transferred above specified limit. Acceptable suffixes are 'k', 'M' and 'G'. |
| **-destiface=<interface>** | Filter on destination interface. |
| **-destip=<ip address>** | Filter on destination IP address. |
| **-destport=<port>** | Filter on TCP/UDP destination port. |
| **-extended** | Show connections with extended information. |
| **-ipver={IPV6 \| IPV4}** | Filter on IP version. |
| **-num=<n>** | Limit list to <n> connections. (Default: 20) |

| | |
|---|---|
| **-protocol=<name/num>** | Filter in IP protocol. |
| **-show** | Show connections. |
| **-srciface=<interface>** | Filter on source interface. |
| **-srcip=<ip address>** | Filter on source IP address. |
| **-srcport=<port>** | Filter on TCP/UDP source port. |
| **-verbose** | Verbose (more information). |

## 2.2.16. cpuid

Display info about the cpu.

### Description

Display the make and model of the machine's CPU.

### Usage

```
cpuid
```

Display the make and model of the machine's CPU.

## 2.2.17. crashdump

Show the contents of the crash.dmp file.

### Description

Show the contents of the crash.dmp file, if it exists.

### Usage

```
crashdump
```

## 2.2.18. cryptostat

Show information about crypto accelerators.

### Description

Show information about active crypto accelerators.

**Usage**

```
cryptostat [-all] [-hashinfo]
```

**Options**

**-all**                                    Show information about all devices.

**-hashinfo**                               Show information about the hardware fastpath hash.

## 2.2.19. dconsole

Displays the content of the diagnose console.

**Description**

The diagnose console is used to help troubleshooting internal problems within the firewall

**Usage**

```
dconsole [-clean] [-flush] [-date=<date>] [-onlyhigh]
```

**Options**

**-clean**                                  Remove all diagnose entries. (Admin only)

**-date=<date>**                            YYYY-MM-DD. Only show entries from this date and forward.

**-flush**                                  Flush all diagnose entries to disk. (Admin only)

**-onlyhigh**                               Only show entries with severity high. (Admin only)

## 2.2.20. dhcp

Display information about DHCP-enabled interfaces or modify/update their leases.

**Description**

Display information about a DHCP-enabled interface.

**Usage**

```
dhcp
```

List DHCP enabled interfaces.

```
dhcp -list
```

List DHCP enabled interfaces.

```
dhcp -show [<interface>]
```

Show information about DHCP enabled interface.

```
dhcp -lease={RENEW | RELEASE} <interface>
```

Modify interface lease.

**Options**

| | |
|---|---|
| **-lease={RENEW \| RELEASE}** | Modify interface lease. (Admin only) |
| **-list** | List all DHCP enabled interfaces. |
| **-show** | Show information about DHCP enabled interface. |
| **<interface>** | DHCP Interface. |

## 2.2.21. dhcprelay

Show DHCP/BOOTP relayer ruleset.

**Description**

Display the content of the DHCP/BOOTP relayer ruleset and the current routed DHCP relays.

Display filter filters relays based on interface/ip (example: if1 192.168.*)

**Usage**

```
dhcprelay
```

Show the currently relayed DHCP sessions.

```
dhcprelay -show [-num={ALL | <Integer>}] [-rules] [-routes]
         [<display filter>]...
```

Show DHCP/BOOTP relayer ruleset.

```
dhcprelay -release <ip address> [-interface=<Interface>]
```

Terminate relayed session.

**Options**

| | |
|---|---|
| **-interface=<Interface>** | Interface. |
| **-num={ALL \| <Integer>}** | Maximum number of entries to show (default: 20). |
| **-release** | Terminate relayed session <[interface:]ip>. (Admin |

only)

| | |
|---|---|
| **-routes** | Show the currently relayed DHCP sessions. |
| **-rules** | Show the DHCP/BOOTP relayer ruleset. |
| **-show** | Show ruleset. |
| **\<display filter\>** | Display filter, filters relays based on interface/ip. |
| **\<ip address\>** | IP address. |

## 2.2.22. dhcpserver

Show content of the DHCP server ruleset.

### Description

Show the content of the DHCP server ruleset and various information about active/inactive leases.

Display filter filters entries based on Interface/MAC/IP (example: If1 192.168.*)

### Usage

```
dhcpserver
```

Show DHCP server leases.

```
dhcpserver -show [-rules] [-leases] [-num=<Integer>]
           [-fromentry=<Integer>] [-mappings] [-utilization]
           [<Display filter>]...
```

Show DHCP server ruleset.

```
dhcpserver -release={BLACKLIST}
```

Release a specific types of IPs.

```
dhcpserver -releaseip <Interface> <IP address>
```

Release an active IP.

### Options

| | |
|---|---|
| **-fromentry=\<Integer\>** | Show entry list from offset \<n\>. |
| **-leases** | Show DHCP server leases. |
| **-mappings** | Show DHCP server IP mappings. |
| **-num=\<Integer\>** | Limit list to \<n\> entries. |
| **-release={BLACKLIST}** | Release specific type of IPs. (Admin only) |
| **-releaseip** | Release an active IP. (Admin only) |

**44**

| | |
|---|---|
| **-rules** | Show DHCP server rules. |
| **-show** | Show ruleset. |
| **-utilization** | Show IP pool utilization. |
| **<Display filter>** | Display filter based on Interface/MAC/IP (eg. If1 192.168.*). |
| **<Interface>** | Interface. |
| **<IP address>** | IP address. |

## 2.2.23. dhcpv6

Display information about DHCPv6-enabled interfaces or modify/update their leases.

### Description

Display information about a DHCPV6-enabled interface.

### Usage

```
dhcpv6
```

List DHCPv6 enabled interfaces.

```
dhcpv6 -list
```

List DHCPv6 enabled interfaces.

```
dhcpv6 -show [<interface>]
```

Show information about DHCPv6 enabled interface.

```
dhcpv6 -lease={RENEW | RELEASE} <interface>
```

Modify interface lease.

### Options

| | |
|---|---|
| **-lease={RENEW | RELEASE}** | Modify interface lease. (Admin only) |
| **-list** | List all DHCPv6 enabled interfaces. |
| **-show** | Show information about DHCPv6 enabled interface. |
| **<interface>** | DHCPv6 Interface. |

## 2.2.24. dhcpv6server

Show content of the DHCPv6 server ruleset.

**Description**

Show the content of the DHCPv6 server ruleset and various information about active/inactive leases.

Display filter filters leases based on interface/mac/ip (example: if1 2001:DB8::*)

**Usage**

```
dhcpv6server
```

Show DHCPv6 server leases.

```
dhcpv6server -releaseip <interface> <IPv6 address>
```

Release an active IP6.

```
dhcpv6server -show [-rules] [-leases] [-num=<Integer>]
             [-fromentry=<Integer>] [<display filter>]...
```

Show DHCP server ruleset.

**Options**

| | |
|---|---|
| **-fromentry=<Integer>** | Shows dhcp server lease list from offset <n>. |
| **-leases** | Show DHCPv6 server leases. |
| **-num=<Integer>** | Limit list to <n> leases. |
| **-releaseip** | Release an active IP. (Admin only) |
| **-rules** | Show DHCPv6 server rules. |
| **-show** | Show ruleset. |
| **<display filter>** | Display filters for leases based on interface/mac/ip (eg. if1 2001:DB8::*). |
| **<interface>** | Interface. |
| **<IPv6 address>** | IPv6 address. |

## 2.2.25. dns

DNS client and queries.

**Description**

Show status of the DNS client and manage pending DNS queries.

**Usage**

```
dns -cache [<FQDNAddress>] [-num=<n>]
```

Show contents of DNS cache.

```
dns -flush
```

Flush the contents of DNS cache.

```
dns
```

Show status of the DNS client.

```
dns -query <domain name> [-type={A | AAAA}]
```

Resolve domain name.

```
dns -list
```

List pending DNS queries.

```
dns -remove
```

Remove all pending DNS queries.

**Options**

| | |
|---|---|
| **-cache** | Show contents of the DNS cache. |
| **-flush** | Flush entire contents of the DNS cache. |
| **-list** | List pending DNS queries. |
| **-num=<n>** | Limit list to <n> addresses. (Default: 20) |
| **-query** | Resolve domain name. |
| **-remove** | Remove all pending DNS queries. |
| **-type={A | AAAA}** | Query type. |
| **<domain name>** | Resolve domain name. |
| **<FQDNAddress>** | FQDN Address object name. |

## 2.2.26. dnsbl

DNSBL.

**Description**

Show status of DNSBL.

**Usage**

```
dnsbl [-show] [<SMTP ALG>] [-clean]
```

**Options**

| | |
|---|---|
| **-clean** | Clear DNSBL statistics for ALG. |
| **-show** | Show DNSBL statistics for ALG. |
| **<SMTP ALG>** | Name of SMTP ALG. |

## 2.2.27. dnscontrol

DNS Control ALG commands.

### Description

Show status for DNS Control ALG sessions.

### Usage

```
dnscontrol
```

List DNS Control Sessions.

```
dnscontrol -list [-num[=<Integer>]] [-verbose]
```

List DNS Control Sessions (Advanced).

```
dnscontrol -stats
```

Show DNS control statistics.

**Options**

| | |
|---|---|
| **-list** | List all DNS Control sessions. |
| **-num[=<Integer>]** | Sessions to list. (Default: 40) |
| **-stats** | Show DNS Control statistics. |
| **-verbose** | Verbose output. |

## 2.2.28. dynroute

Show dynamic routing policy.

### Description

Show the dynamic routing policy filter ruleset and current exports.

In the "Flags" field of the dynrouting exports, the following letters are used:

**o**    Route describe the optimal path to the network

**u**     Route is unexported

**Usage**

```
dynroute [-rules] [-exports]
```

**Options**

**-exports**                             Show current exports.

**-rules**                               Show dynamic routing, filter ruleset.

## 2.2.29. filedownload

File download stats.

### Description

Show statistics of the File Download engine.

### Usage

```
filedownload
```
Show active downloads.

```
filedownload -inactive
```
Show inactive downloads.

```
filedownload -active
```
Show active downloads.

### Options

**-active**                              Show active downloads.

**-inactive**                            Show inactive downloads.

## 2.2.30. frags

Show active fragment reassemblies.

### Description

List active fragment reassemblies.

More detailed information can optionally be obtained for specific reassemblies:

**NEW**      Newest reassembly

**ALL**       All reassemblies

**0..1023**    Assembly 'N'

---

**Example 2.9. frags**

```
frags NEW
frags 254
```

---

**Usage**

```
frags [{NEW | ALL | <reassembly id>}] [-free] [-done] [-num=<n>]
```

**Options**

**-done**                                        List done (lingering) reassemblies.

**-free**                                        List free instead of active.

**-num=<n>**                                     List <n> entries. (Default: 20)

**{NEW | ALL | <reassembly id>}**                Show in-depth info about reassembly <n>. (Default: all)

## 2.2.31. ha

Show and change HA status.

**Description**

Show current HA status.

**Usage**

```
ha [-activate] [-deactivate]
```

**Options**

**-activate**                                    Go active. (Admin only)

**-deactivate**                                  Go inactive. (Admin only)

## 2.2.32. hostmon

Show Host Monitor statistics.

### Description

Show active Host Monitor sessions.

### Usage

```
hostmon [-verbose] [-num=<n>]
```

### Options

**-num=<n>**                            Limit list to <n> entries. (Default: 20)

**-verbose**                            Verbose output.

## 2.2.33. httpalg

Commands related to the HTTP Application Layer Gateway.

### Description

Show information about the WCF cache or list the overridden WCF hosts.

### Usage

```
httpalg -override [-flush]
```

List or flush hosts that have overridden the wcf filter.

```
httpalg -wcfcache [-show] [-url=<String>] [-flush] [-verbose]
        [-count] [-server[={STATUS | CONNECT | DISCONNECT}]]
        [-num=<n>]
```

Display URL cache information.

### Options

**-count**                             Only display cache count.

**-flush**                             Removes all entries.

**-num=<n>**                           Limit list to <n> entries. (Default: 20)

**-override**                          List hosts that have overridden the wcf filter.

**-server[={STATUS | CONNECT | DISCONNECT}]**    Web Content Filtering Server options. (Default: status)

*51*

| | |
|---|---|
| **-show** | Show Web Content Filtering cache data. |
| **-url=<String>** | Limits the output from the show command to only match the specified characters. |
| **-verbose** | Verbose. |
| **-wcfcache** | Show statistics of WCF functionality. |

## 2.2.34. httpposter

Display HTTP Poster status.

### Description

Display configuration and status of configured HTTPPoster_URLx targets.

### Usage

```
httpposter [-repost=<Integer>]
```

### Options

| | |
|---|---|
| **-repost=<Integer>** | Re-post URL now. (Admin only) |

## 2.2.35. hwm

Show hardware monitor sensor status.

### Description

Show hardware monitor sensor status.

### Usage

```
hwm [-all] [-verbose]
```

### Options

| | |
|---|---|
| **-all** | Show ALL sensors, WARNING: use at own risk, may take long time for highspeed ifaces to cope. |
| **-verbose** | Show sensor number, type and limits. |

## 2.2.36. idppipes

Show and remove hosts that are piped by IDP.

**Description**

Show list of currently piped hosts.

**Usage**

```
idppipes
```

List all idppipes.

```
idppipes -show [-host=<ip addr>]
```

Lists hosts for which new connections are piped by IDP.

```
idppipes -unpipe [-all] [-host=<ip addr>]
```

Remove piping for the specified host.

**Options**

| | |
|---|---|
| **-all** | mark all hosts. |
| **-host=<ip addr>** | Filter on source IP address. |
| **-show** | Lists hosts for which new connections are piped by IDP. |
| **-unpipe** | Remove piping for the specified host. (Admin only) |

## 2.2.37. ifstat

Show interface statistics.

**Description**

Show list of attached interfaces, or in-depth information about a specific interface.

**Usage**

```
ifstat [<Interface>] [-filter=<expr>] [-pbr=<table name>]
       [-num=<n>] [-restart] [-allindepth] [-maclist]
       [-snmpnewindexes]
```

**Options**

| | |
|---|---|
| **-allindepth** | Show in-depth information about all interfaces. |
| **-filter=<expr>** | Filter list of interfaces. |

| | |
|---|---|
| **-maclist** | Show MAC addresses for all interfaces. |
| **-num=\<n\>** | Limit list to \<n\> lines. (Default: 20) |
| **-pbr=\<table name\>** | Only list members of given PBR table(s). |
| **-restart** | Stop and restart the interface. (Admin only) |
| **-snmpnewindexes** | Renumber persistent SNMP interface indexes for all interfaces. A reconfigure must follow this command in order to generate the new indexes. |
| **\<Interface\>** | Name of interface. |

## 2.2.38. igmp

IGMP Interfaces.

### Description

Show information about the current state of the IGMP interfaces.

Send simulated messages to test configuration of the interface.

### Usage

```
igmp
```

Prints the current IGMP state.

```
igmp -state [<Interface>]
```

Prints the current IGMP state. If an interface is specified, more details are provided.

```
igmp -query <Interface> [<MC address> [<router address>]]
```

Simulate an incoming IGMP query message.

```
igmp -join <Interface> <MC address> [<host address>]
```

Simulate an incoming IGMP join message.

```
igmp -leave <Interface> <MC address> [<host address>]
```

Simulate an incoming IGMP leave message.

### Options

| | |
|---|---|
| **-join** | Simulate an incoming IGMP join message. |
| **-leave** | Simulate an incoming IGMP leave message. |
| **-query** | Simulate an incoming IGMP query message. |
| **-state** | Show the current IGMP state. |

*54*

| | |
|---|---|
| **\<host address\>** | Host IP address. |
| **\<Interface\>** | Interface. |
| **\<MC address\>** | Multicast Address. |
| **\<router address\>** | Router IP address. |

## 2.2.39. ihs

Alias for **ipsechastat**.

## 2.2.40. ike

Initiate/delete/show IKE negotiated SAs.

### Description

Command to do various operations on IKE negotiated Security Associations.

### Usage

```
ike -stat [<IPsecTunnelBase>] [-cfgmode]
```

Show global or interface statistics about IKE SAs.

```
ike -mem
```

Show memory statistics about the IKE enigne.

```
ike -delete [<ip address>] [-srcif=<Interface>]
    [-tunnel=<IPsecTunnelBase>] [-force]
```

Delete IKE SAs.

```
ike -connect [<IPsecTunnelBase>]
```

Setup IKE and IPsec SAs for a specified tunnel.

```
ike -tunnels [<IPsecTunnelBase>] [-num={ALL | <Integer>}] [-force]
```

Show configured tunnels.

```
ike -show [<ip address>] [-num={ALL | <Integer>}]
    [-srcif=<Interface>] [-verbose] [-force]
    [-tunnel=<IPsecTunnelBase>]
```

Show current IKE SAs.

```
ike -snoop [<ip address>] [-match] [-brief] [-off]
```

Enable/disable IKE snooping.

```
ike -ha [-clear]
```

Shows statistics about IKE/IPsec SAs synchronized and how many that failed to import. Sent

statistics shows how many packets that has been sent to the other cluster member when this node was active and receive statistics show how many packets/failures it got as inactive.

```
ike
```

Show current IKE SAs.

**Options**

| | |
|---|---|
| **-brief** | Show only header information. |
| **-cfgmode** | Show statistics for config mode pool. |
| **-clear** | Reset all statistics. |
| **-connect** | Setup IKE and IPsec SAs for a specified tunnel. |
| **-delete** | Delete IKE SAs. (Admin only) |
| **-force** | Don't send notifications. Delete without delay. |
| **-ha** | Show HA synchronizing statistics for IKE/IPsec SAs. |
| **-match** | Turn on snooping of tunnel matching. |
| **-mem** | Show memory statistics. |
| **-num={ALL | <Integer>}** | Maximum number of entries to show (default: 40/8). |
| **-off** | Turn off IKE snoop. |
| **-show** | Show information on current IKE SAs. |
| **-snoop** | Enable/disable snooping of IKE messages. (Admin only) |
| **-srcif=<Interface>** | Interface used to reach the remote endpoint. |
| **-stat** | Show verbose information. |
| **-tunnel=<IPsecTunnelBase>** | IPsec interface. |
| **-tunnels** | Show information on configured tunnels. |
| **-verbose** | Show verbose information. |
| **<ip address>** | IP address of remote SG/peer. |
| **<IPsecTunnelBase>** | IPsec interface. |

## 2.2.41. ikesnoop

Enable or disable IKE-snooping.

**Description**

Turn IKE on-screen snooping on/off. Useful for troubleshooting IPsec connections.

**Usage**

```
ikesnoop
```

Show IKE snooping status.

```
ikesnoop -on [<ip address>] [-verbose]
```

Enable IKE snooping.

```
ikesnoop -off
```

Disable IKE snooping.

**Options**

| | |
|---|---|
| **-off** | Turn IKE snooping off. |
| **-on** | Turn IKE snooping on. |
| **-verbose** | Enable IKE snooping with verbose output. |
| **<ip address>** | IP address to snoop. |

> *Deprecated*
> *(2014-05-27) Replaced by command **ike -snoop**. Deprecated commands may be removed in future releases.*

## 2.2.42. ippool

Show IP pool information.

**Description**

Show information about the current state of the configured IP pools.

**Usage**

```
ippool
```

Show IP pool information.

```
ippool -release [<ip address>] [-all]
```

Forcibly free IP assigned to subsystem.

```
ippool -renew [<ip address>] [-all]
```

Try to renew IP leases through DHCP Server.

```
ippool -show [-verbose] [-num=<n>]
```

Show IP pool information.

**Options**

| | |
|---|---|
| **-all** | Free or renew all IP addresses. |
| **-num=<n>** | Limit list to <n> entries. (Default: 100) |
| **-release** | Forcibly free IP assigned to subsystem. (Admin only) |
| **-renew** | Try to renew IP leases through DHCP Server. (Admin only) |
| **-show** | Show IP pool information. |
| **-verbose** | Verbose output. |
| **<ip address>** | IP address to free or renew. |

# 2.2.43. ipreputation

IP Reputation stats.

**Description**

Show IP Reputation engine information and perform IP Reputation operations.

**Usage**

```
ipreputation -query <ip address> [-category[={ALL | SPAM_SOURCES |
            WINDOWS_EXPLOITS | WEB_ATTACKS | BOTNETS | SCANNERS |
            DOS | REPUTATION | PHISHING | PROXY | NETWORK |
            CLOUD_PROVIDERS | MOBILE_THREATS | <String>}]]
            [-lookup[={ALLMETHODS | LOCAL | CLOUD | CACHE}]]
```

Perform an advanced IP Reputation Query.

```
ipreputation -query <ip address>
```

Perform an IP Reputation Query.

```
ipreputation -show [-updates] [-verbose]
```

Show IP Reputation update information.

```
ipreputation -updates [-update] [-verbose]
```

Update IP Reputation Database.

```
ipreputation
```

Show engine information.

```
ipreputation -cache [-show] [-flush] [-num=<n>] [-verbose]
```

IP Reputation cache.

```
ipreputation -subsystems [-verbose]
```

Show subsystem information.

```
ipreputation -statistics[={TOTAL | 24H | 2M | 30D}]
```

Show IP Reputation statistics.

**Options**

| | |
|---|---|
| **-cache** | IP Reputation cache. |
| **-category[={ALL \| SPAM_SOURCES \| WINDOWS_EXPLOITS \| WEB_ATTACKS \| BOTNETS \| SCANNERS \| DOS \| REPUTATION \| PHISHING \| PROXY \| NETWORK \| CLOUD_PROVIDERS \| MOBILE_THREATS \| <String>}]** | IP Reputation category. (Default: all) |
| **-flush** | Remove IP Reputation cache entries. |
| **-lookup={ALLMETHODS \| LOCAL \| CLOUD \| CACHE}]** | Query lookup method. (Default: allmethods) |
| **-num=<n>** | Limit list to <n> entries. (Default: 40) |
| **-query** | Perform an IP Reputation query. |
| **-show** | Show IP Reputation update information. |
| **-statistics[={TOTAL \| 24H \| 2M \| 30D}]** | IP Reputation statistics. (Default: 24h) |
| **-subsystems** | Show subsystem information. |
| **-update** | IP Reputation updates. |
| **-updates** | Update the IP Reputation database. |
| **-verbose** | Verbose output. |
| **<ip address>** | IP address. |

## 2.2.44. ipsec

Show the IPsec SAs in use.

**Description**

List the currently active IPsec SAs, optionally only showing SAs matching the pattern given for the argument "iface".

**Usage**

```
ipsec -stat [<IPsecTunnelBase>]
```

Show global or interface statistics about IPsec SAs.

```
ipsec -show [<IPsecTunnelBase>] [-verbose] [-num={ALL | <Integer>}]
      [-srcif=<Interface>] [-force] [-usage]
```

Show SA information.

```
ipsec
```

Show SA information.

**Options**

| | |
|---|---|
| **-force** | Bypass confirmation question. |
| **-num={ALL | <Integer>}** | Maximum number of entries to show (default: 40/8). |
| **-show** | Show SA information. |
| **-srcif=<Interface>** | Interface used to reach the remote endpoint. |
| **-stat** | Show IPsec statistics. |
| **-usage** | Show detailed SA statistics information. |
| **-verbose** | Show verbose information. |
| **<IPsecTunnelBase>** | IPsec interface. |

## 2.2.45. ipsecdefines

Display various DEFINES that specify the system performance.

### Description

Display various DEFINES that specify the system performance.

### Usage

```
ipsecdefines
```

## 2.2.46. ipsecglobalstats

Show global ipsec statistics.

### Description

List global IPsec statistics.

### Usage

```
ipsecglobalstats -mem [-verbose]
```

Start IKE test.

```
ipsecglobalstats -verbose
```

Start IKE test.

```
ipsecglobalstats
```

Show interfaces.

**Options**

**-mem**                                        Show memory statistics.

**-verbose**                                    Show all statistics.

> ### *Deprecated*
> *(2014-05-27) Replaced by command **ike -stat**. Deprecated commands may be removed in future releases.*

## 2.2.47. ipsechastat

Show statistics about HA synchronization for IPsec.

### Description

Shows statistics about IKE/IPsec SAs synchronized and how many that failed to import. Sent statistics shows how many packets that has been sent to the other cluster member when this node was active and receive statistics show how many packets/failures it got as inactive.

### Usage

```
ipsechastat [-clear]
```

### Options

**-clear**                                      Reset all statistics.

## 2.2.48. ipsecstats

Show the SAs in use.

### Description

List the currently active IKE and IPsec SAs, optionally only showing SAs matching the pattern given for the argument "tunnel".

### Usage

```
ipsecstats [-ike] [<tunnel>] [-ipsec] [-usage] [-verbose]
           [-num={ALL | <Integer>}] [-force]
```

**Options**

| | |
|---|---|
| **-force** | Bypass confirmation question. |
| **-ike** | Show IKE SAs. |
| **-ipsec** | Show IPsec SAs. |
| **-num={ALL | <Integer>}** | Maximum number of entries to show (default: 40/8). |
| **-usage** | Show detailed SA statistics information. |
| **-verbose** | Show verbose information. |
| **<tunnel>** | Only show SAs matching pattern. |

> **Deprecated**
> *(2014-05-27) Replaced by command **ipsec -show**. Deprecated commands may be removed in future releases.*

# 2.2.49. ipsectunnels

Lists the current IPsec configuration.

## Description

Lists the current IPsec configuration,

## Usage

```
ipsectunnels -iface=<recv iface>
```

Show specific interface.

```
ipsectunnels -num={ALL | <Integer>} [-force]
```

Show specific number if interface.

```
ipsectunnels
```

Show interfaces.

**Options**

| | |
|---|---|
| **-force** | Bypass confirmation question. |
| **-iface=<recv iface>** | IPsec interface to show information about. |
| **-num={ALL | <Integer>}** | Maximum number of entries to show (default: 40). |

***Deprecated***
*(2014-05-27) Replaced by command **ike -tunnels**. Deprecated commands may be removed in future releases.*

## 2.2.50. killsa

Kill all SAs belonging to the given remote SG/peer.

### Description

Kill all (IPsec and IKE) SAs associated with a given remote IKE peer IP or optional all SA:s in the system. IKE delete messages are sent.

### Usage

```
killsa <ip address> [-iface=<interface>]
```

Delete SAs belonging to provided remote SG/peer.

```
killsa -all [-iface=<interface>]
```

Delete all SAs.

### Options

**-all**                                                    Kill all SAs.

**-iface=<interface>**                        Remote interface for SG/peer.

**<ip address>**                                   IP address of remote SG/peer.

***Note***
*Requires Administrator privileges.*

***Deprecated***
*(2014-05-27) Replaced by command **ike -delete**. Deprecated commands may be removed in future releases.*

## 2.2.51. l2tp

Show L2TP information.

### Description

Shows L2TP information and statistics.

**Usage**

```
l2tp -state={ALL | ACTIVE | LISTENING} [-child] [-num=<Integer>]
```

Show all L2TP sessions.

```
l2tp -l2tpserver=<PPTP/L2TP Server> [-l2tpv3server=<L2TPv3 Server>]
     [-l2tpv3client=<L2TPv3 Client>] [-l2tpclient=<PPTP/L2TP
     Client>] [-state={ALL | ACTIVE | LISTENING}] [-child]
     [-num=<Integer>]
```

List L2TP sessions.

```
l2tp -l2tpv3server=<L2TPv3 Server> [-l2tpserver=<PPTP/L2TP Server>]
     [-state={ALL | ACTIVE | LISTENING}] [-child] [-num=<Integer>]
```

List L2TP sessions.

```
l2tp -l2tpclient=<PPTP/L2TP Client> [-l2tpv3client=<L2TPv3 Client>]
     [-state={ALL | ACTIVE | LISTENING}] [-child] [-num=<Integer>]
```

List L2TP sessions.

```
l2tp -l2tpv3client=<L2TPv3 Client> [-l2tpclient=<PPTP/L2TP Client>]
     [-state={ALL | ACTIVE | LISTENING}] [-child] [-num=<Integer>]
```

List L2TP sessions.

**Options**

| | |
|---|---|
| **-child** | Include child sessions. |
| **-l2tpclient=<PPTP/L2TP Client>** | Only show sessions belonging to this L2TPClient. |
| **-l2tpserver=<PPTP/L2TP Server>** | Only show sessions belonging to this L2TPServer. |
| **-l2tpv3client=<L2TPv3 Client>** | Only show sessions belonging to this L2TPv3Client. |
| **-l2tpv3server=<L2TPv3 Server>** | Only show sessions belonging to this L2TPv3Server. |
| **-num=<Integer>** | Number of entries to list. |
| **-state={ALL \| ACTIVE \| LISTENING}** | Show sessions with specified state. (Default: active) |

## 2.2.52. languagefiles

Manage language files on disk.

**Description**

Manage language files on disk

**Usage**

```
languagefiles
```

Show all language files on disk.

```
languagefiles -remove=<String>
```

Remove a language file from disk.

**Options**

**-remove=<String>**                Specify language file to delete.

## 2.2.53. ldap

LDAP information.

**Description**

Status and statistics for the configured LDAP databases.

**Usage**

```
ldap
```

List all LDAP databases.

```
ldap -list
```

List all LDAP databases.

```
ldap -show [<LDAP Server>]
```

Show LDAP database status and statistics.

```
ldap -reset [<LDAP Server>]
```

Reset LDAP database.

**Options**

**-list**                          List all LDAP databases.

**-reset**                         Reset status for LDAP database. (Admin only)

**-show**                          Show status and statistics.

**<LDAP Server>**                  LDAP database.

## 2.2.54. license

License management.

**Description**

Display the current license.

**Usage**

```
license
```

Show the contents of the current license.

```
license -show
```

Show the contents of the current license.

**Options**

**-show**                                          Show current status and credentials.

## 2.2.55. linkmon

Display link montitoring statistics.

**Description**

If link monitor hosts have been configured, linkmon will monitor host reachability to detect link/NIC problems.

**Usage**

```
linkmon
```

## 2.2.56. logout

Logout user.

**Description**

Logout current user.

**Usage**

```
logout
```

## 2.2.57. lwhttp

Commands related to the Light-Weight HTTP inspection engine.

### Description

The lwhttp CLI command prints information about the Light-Weight HTTP inspection engine a.k.a. LW-HTTP ALG.

The LW-HTTP inspection engine is automaticlaly enabled for IP policies with HTTP protocol validation or a web profile configured.

Compared to the ordinary HTTP-ALG, the LW-HTTP inspector provides better throughput performance without affecting network security.

### Usage

```
lwhttp
```

## 2.2.58. macstorage

The MAC address storage.

### Description

The mac address storage keeps mac addresses persistent for SR-IOV interfaces when used in virtual environments.

### Usage

```
macstorage
```

## 2.2.59. memory

Show memory information.

### Description

Show core memory consumption. Also show detailed memory use of some components and lists.

### Usage

```
memory
```

*67*

## 2.2.60. natpool

Show current NAT Pools.

### Description

Show current NAT Pools and in-depth information.

### Usage

```
natpool [-verbose] [<pool name> [<IP4 Address>]] [-num=<Integer>]
```

### Options

| | |
|---|---|
| **-num=<Integer>** | Maximum number of items to list (default: 20). |
| **-verbose** | Verbose (more information). |
| **<IP4 Address>** | Translated IP. |
| **<pool name>** | NAT Pool name. |

## 2.2.61. nd

Show Neighbor Discovery entries for given interface.

### Description

List the Neighbor Discovery cache entries of specified interfaces.

If no interface is given the Neighbor Discovery cache entries of all interfaces will be presented.

The presented list can be filtered using the *ip* and *hw* options.

### Usage

```
nd -routerdiscovery [<Interface>] [-num=<n>]
```

Show Router Discovery enabled interfaces.

```
nd
```

Show all Neighbor Discovery entries.

```
nd -show [<Interface>] [-ip=<pattern>] [-hw=<pattern>] [-num=<n>]
```

Show Neighbor Discovery entries.

```
nd -hashinfo [<Interface>]
```

Show information on hash table health.

```
nd -flush [<Interface>]
```

Flush Neighbor Discovery cache of specified interface.

```
nd -query=<ip> <Interface>
```

Send Neighbor Solicitation for IP.

```
nd -del=<ip> <Interface>
```

Delete ND cache entry.

**Options**

| | |
|---|---|
| **-del=\<ip\>** | Delete ND cache entry \<ip\>. |
| **-flush** | Flush Neighbor Discovery cache of all specified interfaces. (Admin only) |
| **-hashinfo** | Show information on hash table health. |
| **-hw=\<pattern\>** | Show only hardware addresses matching pattern. |
| **-ip=\<pattern\>** | Show only IP addresses matching pattern. |
| **-num=\<n\>** | Show only the first \<n\> entries per interface. (Default: 20) |
| **-query=\<ip\>** | Send Neighbor Solicitation for \<ip\>. |
| **-routerdiscovery** | Show Router Discovery enabled interfaces. |
| **-show** | Show Neighbor Discovery entries for given interface(s). |
| **\<Interface\>** | Interface name. |

## 2.2.62. ndsnoop

Toggle snooping and displaying of ARP requests.

**Description**

Toggle snooping and displaying of Neighbor Discovery queries and responses on-screen.

The snooped messages are displayed before the access section validates the sender IP addresses in the ARP data.

**Usage**

```
ndsnoop
```

Show snooped interfaces.

```
ndsnoop {ALL | NONE | <interface>} [-verbose]
```

Snoop specified interface.

**Options**

**-verbose**                                    Verbose.

**{ALL | NONE | <interface>}**                  Interface name.

> ***Note***
> *Requires Administrator privileges.*

# 2.2.63. neighborcache

Shows the default contents of the neighbor cache.

### Description

Contains information such as hostname, configured name, hardware address and ip4 address, for the firewall's network neighbors.

### Usage

```
neighborcache [-show] [-names] [-users] [-ipv6] [-filter={INACTIVE
            | ACTIVE}]
```

### Options

**-filter={INACTIVE | ACTIVE}**                 Shows the filtered contents of the neighbor cache, based on state.

**-ipv6**                                       Shows the ipv6 addresses for entries in the neighbor cache.

**-names**                                      Shows the host name and configured name for entries in the neighbor cache.

**-show**                                       Shows the default contents of the neighbor cache.

**-users**                                      Shows any authenticated users against their neighbor cache entry.

# 2.2.64. netobjects

Show runtime values of network objects.

### Description

Displays named network objects and their contents.

**Example 2.10. List network objects which have names containing "net".**

```
netobjects *net*
```

**Usage**

```
netobjects [<String>] [-num=<num>]
```

**Options**

**-num=<num>**                                    Number of entries to show. (Default: 20)

**<String>**                                      Name or pattern.

---

# 2.2.65. ospf

Show runtime OSPF information.

**Description**

Show runtime information about the OSPF router process(es).

Note: $-process$ is only required if there are >1 OSPF router processes.

**Usage**

```
ospf
```

Show runtime information.

```
ospf -iface [<interface>] [-process=<OSPF Router Process>]
```

Show interface information.

```
ospf -area [<OSPF Area>] [-process=<OSPF Router Process>]
```

Show area information.

```
ospf -neighbor [<OSPF Neighbor>] [-process=<OSPF Router Process>]
```

Show neighbor information.

```
ospf -route [{HA | ALT}] [-process=<OSPF Router Process>]
```

Show the internal OSPF process routingtable.

```
ospf -database [-verbose] [-process=<OSPF Router Process>]
```

Show the LSA database.

```
ospf -lsa <lsaID> [-process=<OSPF Router Process>]
```

Show details for a specified LSA.

```
ospf -snoop={ON | OFF} [-process=<OSPF Router Process>]
```

Show troubleshooting messages on the console.

```
ospf -ifacedown <interface> [-process=<OSPF Router Process>]
```

Take specified interface offline.

```
ospf -ifaceup <interface> [-process=<OSPF Router Process>]
```

Take specified interface online.

```
ospf -execute={STOP | START | RESTART} [-process=<OSPF Router
      Process>]
```

Start/stop/restart OSPF process.

**Options**

| | |
|---|---|
| **-area** | Show area information. |
| **-database** | Show the LSA database. |
| **-execute={STOP \| START \| RESTART}** | Start/stop/restart OSPF process. (Admin only) |
| **-iface** | Show interface information. |
| **-ifacedown** | Take specified interface offline. (Admin only) |
| **-ifaceup** | Take specified interface online. (Admin only) |
| **-lsa** | Show details for a specified LSA <lsaID>. |
| **-neighbor** | Show neighbor information. |
| **-process=<OSPF Router Process>** | Required if there are >1 OSPF router processes. |
| **-route** | Show the internal OSPF process routingtable. |
| **-snoop={ON \| OFF}** | Show troubleshooting messages on the console. (Admin only) |
| **-verbose** | Increase amount of information to display. |
| **<interface>** | OSPF enabled interface. |
| **<interface>** | OSPF enabled interface. |
| **<lsaID>** | LSA ID. |
| **<OSPF Area>** | OSPF Area. |
| **<OSPF Neighbor>** | Neighbor. |
| **{HA \| ALT}** | Show HA routingtable. |

## 2.2.66. pcapdump

Packet capturing.

### Description

Packet capture engine

### Usage

```
pcapdump
```

Show capture status.

```
pcapdump -start [<interface(s)>] [-size=<value>] [-snaplen=<value>]
         [-count=<value>] [-out] [-out-nocap] [-eth=<Ethernet
         Address>] [-ethsrc=<Ethernet Address>] [-ethdest=<Ethernet
         Address>] [-ip=<IP4 Address>] [-ipsrc=<IP4 Address>]
         [-ipdest=<IP4 Address>] [-port=<String>]
         [-srcport=<String>] [-destport=<String>]
         [-proto=<0...255>] [-icmp] [-tcp] [-udp] [-promisc]
         [-ipversion=<1...15>]
```

Start capture.

```
pcapdump -stop [<interface(s)>]
```

Stop capture.

```
pcapdump -status
```

Show capture status.

```
pcapdump -show [<interface(s)>] [-num={ALL | <Integer>}]
```

Show a captured packets brief.

```
pcapdump -write [<interface(s)>] [-filename=<String>]
```

Write the captured packets to disk.

```
pcapdump -wipe
```

Remove all captured packets from memory.

```
pcapdump -cleanup
```

Remove all captured packets, release capture mode and delete all written capture files from disk.

### Options

**-cleanup**
Remove all captured packets, release capture mode and delete all written capture files from disk.

**-count=<value>**
Number of packets to capture.

**-destport=<String>**
Destination TCP/UDP port filter.

| | |
|---|---|
| **-eth=<Ethernet Address>** | Ethernet address filter. |
| **-ethdest=<Ethernet Address>** | Ethernet destination address filter. |
| **-ethsrc=<Ethernet Address>** | Ethernet source address filter. |
| **-filename=<String>** | Filename for capture file. |
| **-icmp** | ICMP filter. |
| **-ip=<IP4 Address>** | IP address filter. |
| **-ipdest=<IP4 Address>** | Destination IP address filter. |
| **-ipsrc=<IP4 Address>** | Source IP address filter. |
| **-ipversion=<1...15>** | IP version filter. |
| **-num={ALL \| <Integer>}** | Maximum number of entries to show (default: 20). |
| **-out** | Realtime packet brief dumped to console. |
| **-out-nocap** | Unbuffered (not stored in memory) realtime packet brief dumped to console. |
| **-port=<String>** | TCP/UDP port filter. |
| **-promisc** | Set iface in promiscuous mode. |
| **-proto=<0...255>** | IP protocol filter. |
| **-show** | Show a captured packets brief. |
| **-size=<value>** | Size (kb) of buffer to store captured packets in memory (default 512kb). |
| **-snaplen=<value>** | Maximum length of each packet to capture. |
| **-srcport=<String>** | Source TCP/UDP port filter. |
| **-start** | Start capture. |
| **-status** | Show capture status. |
| **-stop** | Stop capture. |
| **-tcp** | TCP filter. |
| **-udp** | UDP filter. |
| **-wipe** | Remove all captured packets from memory. |
| **-write** | Write the captured packets to disk. |
| **<interface(s)>** | Name of interface(s). |

> ***Note***
> *Requires Administrator privileges.*

## 2.2.67. pipes

Show pipes information.

### Description

Show list of configured pipes / pipe details / pipe users.

Note: The "pipes" command is not executed right away; it is queued until the end of the second, when pipe values are calculated.

### Usage

```
pipes
```

List all pipes.

```
pipes -users [<Pipe>] [-expr=<String>]
```

List users of a given pipe.

```
pipes -show [<Pipe>] [-expr=<String>]
```

Show pipe details.

### Options

| | |
|---|---|
| **-expr=\<String\>** | Pipe wildcard(*) expression. |
| **-show** | Show pipe details. |
| **-users** | List users of a given pipe. |
| **\<Pipe\>** | Show pipe details. |

## 2.2.68. pptp

Show PPTP information.

### Description

Shows PPTP information and statistics.

### Usage

```
pptp -state={ALL | ACTIVE | LISTENING | CHILDONLY} [-child]
     [-num=<Integer>]
```

Show all PPTP sessions.

```
pptp -pptpserver=<PPTP/L2TP Server> [-state={ALL | ACTIVE |
     LISTENING | CHILDONLY}] [-child] [-num=<Integer>]
```

*75*

List PPTP sessions.

```
pptp -pptpclient=<PPTP/L2TP Client> [-state={ALL | ACTIVE |
      LISTENING | CHILDONLY}] [-child] [-num=<Integer>]
```

List PPTP sessions.

**Options**

| | |
|---|---|
| **-child** | Include child sessions. |
| **-num=<Integer>** | Number of entries to list. |
| **-pptpclient=<PPTP/L2TP Client>** | Only show sessions belonging to this PPTP client (L2TPClient with TunnelProtocol == PPTP). |
| **-pptpserver=<PPTP/L2TP Server>** | Only show sessions belonging to this PPTP server (L2TPServer with TunnelProtocol == PPTP). |
| **-state={ALL | ACTIVE | LISTENING | CHILDONLY}** | Show sessions with specified state. (Default: active) |

# 2.2.69. pptpalg

Show PPTP ALG information.

**Description**

Shows information and statistics of the PPTP ALGs.

**Usage**

```
pptpalg
```

Show all configured PPTP ALGs.

```
pptpalg -sessions <PPTP ALG> [-verbose] [-num=<Integer>]
```

List all PPTP sessions.

```
pptpalg -services <PPTP ALG>
```

List all services attached to PPTP ALG.

**Options**

| | |
|---|---|
| **-num=<Integer>** | Number of entries to list. |
| **-services** | List all services attached to PPTP ALG. |
| **-sessions** | List all session using a PPTP tunnel. |
| **-verbose** | Verbose output. |
| **<PPTP ALG>** | PPTP ALG. |

## 2.2.70. reconfigure

Initiates a configuration re-read.

### Description

Restart the firewall using the currently active configuration.

### Usage

**Note**
*Requires Administrator privileges.*

## 2.2.71. rekeysa

Rekey IPsec or IKE SAs established with given remote peer.

### Description

Rekey IPsec or IKE SAs associated with a given remote IKE peer, or optionally all IPsec or IKE SAs in the system.

### Usage

```
rekeysa -ike <ip address>
```

Rekey IKE SAs.

```
rekeysa -ipsec <ip address>
```

Rekey IPsec SAs.

```
rekeysa <ip address>
```

Rekey IPsec SAs.

### Options

**-ike**                                       Rekey IKE SAs.

**-ipsec**                                     Rekey IPsec SAs.

**<ip address>**                               IP address of remote peer.

**Note**
*Requires Administrator privileges.*

## 2.2.72. route

Alias for **routes**.

## 2.2.73. routemon

List the currently monitored interfaces and gateways.

### Description

List the currently monitored interfaces and/or gateways.

### Usage

```
routemon
```

## 2.2.74. routes

Display routing lists.

### Description

Display information about the routing table(s):

- Contents of a (named) routing table.

- The list of routing tables, along with a total count of route entries in each table, as well as how many of the entries are single-host routes.

Note that "core" routes for interface IP addresses are not normally shown. Use the `-all` switch to show core routes also.

Use the `-switched` switch to show only switched routes.

Explanation of Flags field of the routing tables:

**O**    Learned via OSPF

**X**    Route is Disabled

**M**    Route is Monitored

**A**    Published via Proxy ARP

**D**    Dynamic (from e.g. DHCP relay, IPsec, L2TP/PPP servers, etc.)

**H**    HA synced from cluster peer

### Usage

```
routes [-all] [<table name>] [-switched] [-flushl3cache] [-num=<n>]
       [-nonhost] [-tables] [-lookup=<ip address>] [-verbose]
```

**Options**

| | |
|---|---|
| **-all** | Also show routes for interface addresses. |
| **-flushl3cache** | Flush Layer 3 Cache. (Admin only) |
| **-lookup=<ip address>** | Lookup the route for the given IP address. |
| **-nonhost** | Do not show single-host routes. |
| **-num=<n>** | Limit display to <n> entries. (Default: 20) |
| **-switched** | Only show switched routes and L3C entries. |
| **-tables** | Display list of named (PBR) routing tables. |
| **-verbose** | Verbose. |
| **<table name>** | Name of routing table. |

## 2.2.75. rtmonitor

Real-time monitor information.

**Description**

Show information about real-time monitor objects, and real-time monitor alerts.

All objects matching the specified filter are displayed. The filter can be the name of an object, or the beginning of a name. If no filter is specified, all objects are displayed.

If the option "monitored" is specified, only objects that have an associated real-time monitor alert are displayed.

---

**Example 2.11. Show all monitored objects in the alg/http category**

```
gw-world:/> rtmonitor alg/http -m
```

---

**Usage**

```
rtmonitor [<filter>] [-terse] [-monitored] [-num={ALL | <Integer>}]
```

**Options**

| | |
|---|---|
| **-monitored** | Only show monitored objects. |
| **-num={ALL \| <Integer>}** | Maximum number of entries to show (default: 20). |

| | |
|---|---|
| **-terse** | Only show object name. |
| **<filter>** | Object filter. |

## 2.2.76. rules

Show rules lists.

### Description

Shows the content of the various types of rules, i.e. main ruleset, pipe ruleset, etc.

---

**Example 2.12. Show a range of rules**

```
rules -verbose 1-5 7-9
```

---

### Usage

```
rules -type=IP [-ruleset={* | MAIN | <IP Rule Set>}] [-verbose]
      [-schedule] [-usageless=<usageless>] [-usagemore=<usagemore>]
      [<rules>]...
```

Show IP rules.

```
rules -type={ROUTING | PIPE | IDP | THRESHOLD | IGMP} [-verbose]
      [-schedule] [-usageless=<usageless>] [-usagemore=<usagemore>]
      [<rules>]...
```

Show a specific type of rules.

### Options

| | |
|---|---|
| **-ruleset={* \| MAIN \| <IP Rule Set>}** | Show a specified IP ruleset. |
| **-schedule** | Filter out rules that are not currently allowed by selected schedules. |
| **-type={IP \| ROUTING \| PIPE \| IDP \| THRESHOLD \| IGMP}** | Type of rules to display. (Default: IP) |
| **-usageless=<usageless>** | Filter on usage below(<=) specified limit. |
| **-usagemore=<usagemore>** | Filter on usage above(>=) specified limit. |
| **-verbose** | Verbose: show all parameters of the rules. |
| **<rules>** | Range of rules to display. (default: all rules). |

## 2.2.77. selftest

Run appliance self tests.

## Description

The appliance self tests are used to verify the correct function of hardware components.

IMPORTANT: In order for a selftest result to be reliable the test must be run using a default configuration and having the SGW disconnected from any networks.

IMPORTANT: Normal SGW operations might be disrupted during the test(s).

The outcome of the throughput crypto accelerator tests are dependent on configuration values. If the number of large buffers (LocalReassSettings->LocalReass_NumLarge) too low, it might lower throughput result. In the field 'Drop/Fail', the 'Drop' column contains the number of packets that were dropped before ever reaching the crypto accelerator and the 'Fail' column contains the number of packets that for some reason failed encryption. The 'Pkt In/Out' field shows the total number of packets sent to, and returned from the accelerator.

The interface tests 'traffic' and 'throughput' are dependent on the settings for the NIC ring sizes and possibly also license limitations. The 'traffic' test uses a uniform random distribution of six packet sizes between 60 and 1518 bytes. The content of each received packet is validated. The 'throughput' test uses only the largest packet size, and does not validate the contents of the received packets.

---

**Example 2.13. Interface ping test between all interfaces**

```
selftest -ping
```

---

**Example 2.14. Interface ping test between interfaces 'if1' and 'if2'**

```
selftest -ping -interfaces=if1,if2
```

---

**Example 2.15. Start 30 min burn-in, testing RAM, storage media and crypto accelerator**

```
selftest -burnin -minutes 30 -media -memory -cryptoaccel
```

---

## Usage

```
selftest -memory [-num=<Integer>]
```

Check the sanity of the RAM.

```
selftest -media [-size=<Integer>]
```

Check the sanity of the disk drive.

```
selftest -mac
```

Check if there are MAC address collisions on the interfaces.

```
selftest -ping [-interfaces=<Interface>]
```

Run a ping test over the interfaces.

```
selftest -throughput [-interfaces=<Interface>]
```

Run a throughput test over the interfaces.

```
selftest -traffic [-interfaces=<Interface>]
```

Run a traffic test over the interfaces.

```
selftest -cryptoaccel
```

Verify the correct functioning of the accelerator cards.

```
selftest -burnin [-hours[=<Integer>]] [-minutes[=<Integer>]]
        [-memory] [-media] [-ping] [-throughput] [-traffic]
        [-cryptoaccel] [-size=<Integer>]
```

Run burn-in tests for a set of sub tests. If no sub tests are specified the following are included: -memory, -ping, -traffic, -cryptoaccel.

```
selftest -abort
```

Abort a running self test.

```
selftest
```

Show the status of a running test.

**Options**

| | |
|---|---|
| **-abort** | Abort a running self test. |
| **-burnin** | Run burn-in tests for a selected set of sub tests. |
| **-cryptoaccel** | Verify the correct functioning of available crypto accelerator cards. |
| **-hours[=<Integer>]** | Test duration in hours. (Default: 48) |
| **-interfaces=<Interface>** | Ethernet interface(s). |
| **-mac** | Check if there are MAC address collisions on the interfaces. |
| **-media** | Check the sanity of the disk drive. |
| **-memory** | Check the sanity of the RAM. |
| **-minutes[=<Integer>]** | Test duration in minutes. (Default: 0) |
| **-num=<Integer>** | Number of times to execute the test. (Default: 1) |
| **-ping** | Run a ping test over the interfaces. |
| **-size=<Integer>** | Size of media space to utilize in the test. Set in MB. (Default: 1) |
| **-throughput** | Run a throughput test over the interfaces. This will show the maximal achievable interface |

throughput.

**-traffic**                                    Run a traffic test over the interfaces. The traffic test uses mixed frame sizes and verifies the content of each received frame.

***Note***
*Requires Administrator privileges.*

## 2.2.78. services

Show runtime values of configured services.

### Description

Shows the runtime values of all configured services.

**Example 2.16.  List all services which names begin with "http"**

```
services http*
```

### Usage

```
services [<String>]
```

### Options

**<String>**                                    Name or pattern.

## 2.2.79. sessionmanager

Session Manager.

### Description

Show information about the Session Manager, and list currently active users.

Explanation of Timeout flags for sessions:

**D**     Session is disabled

**S**     Session uses a timeout in its subsystem

**-**     Session does not use timeout

**Usage**

```
sessionmanager
```

Show Session Manager status.

```
sessionmanager -status
```

Show Session Manager status.

```
sessionmanager -list [-num=<n>]
```

List active sessions.

```
sessionmanager -info <session name> <database>
```

Show in-depth information about session(s).

```
sessionmanager -message <session name> <database> <message text>
```

Send message to session with console.

```
sessionmanager -disconnect <session name> <database> [<IP Address>
              [{LOCAL | SSH | NETCON | HTTP | HTTPS}]]
```

Forcibly terminate session(s).

**Options**

| | |
|---|---|
| **-disconnect** | Forcibly terminate session(s). (Admin only) |
| **-info** | Show in-depth information about session. |
| **-list** | List active sessions. |
| **-message** | Send message to session. |
| **-num=<n>** | List <n> number of session. |
| **-status** | Show Session Manager status. |
| **<database>** | Name of user database. |
| **<IP Address>** | IP address. |
| **<message text>** | Message to send. |
| **<session name>** | Name of session. |
| **{LOCAL | SSH | NETCON | HTTP | HTTPS}** | Session type. |

## 2.2.80. settings

Show settings.

**Description**

Show the contents of the settings section, category by category.

**Usage**

```
settings
```

Show list of categories.

```
settings <category>
```

Show settings in category.

**Options**

| | |
|---|---|
| **<category>** | Show settings in category. |

## 2.2.81. shutdown

Initiate core or system shutdown.

**Description**

Initiate restart of the core/system.

**Usage**

```
shutdown [<seconds>] [-normal] [-reboot]
```

**Options**

| | |
|---|---|
| **-normal** | Initiate core shutdown. |
| **-reboot** | Initiate system reboot. |
| **<seconds>** | Seconds until shutdown. (Default: 5) |

> ***Note***
> *Requires Administrator privileges.*

## 2.2.82. sipalg

SIP ALG.

**Description**

List running SIP-ALG configurations, SIP registration and call information.

The -flags option with -snoop allows any combination of the following values:

- 0x00000001 GENERAL

- 0x00000002 ERRORS

- 0x00000004 OPTIONS

- 0x00000008 PARSE

- 0x00000010 VALIDATE

- 0x00000020 SDP

- 0x00000040 ALLOW_CHANGES

- 0x00000080 SUPPORTED_CHANGES

- 0x00000100 2543COMPLIANCE

- 0x00000200 RECEPTION

- 0x00000400 SESSION

- 0x00000800 REQUEST

- 0x00001000 RESPONSE

- 0x00002000 TOPO_CHANGES

- 0x00004000 MEDIA

- 0x00008000 CONTACT

- 0x00010000 CONN

- 0x00020000 PING

- 0x00040000 TRANSACTION

- 0x00080000 CALLLEG

- 0x00100000 REGISTRY

Flags can be added in the usual way. The default value is 0x00000003 (GENERAL and ERRORS).

NOTE: 'verbose' option outputs a lot of information on the console which may lead to system instability. Use with caution.

**Usage**

```
sipalg -definition [<alg>]
```

Show running ALG configuration parameters.

```
sipalg -registration[={SHOW | FLUSH}] <alg>
```

Show or flush current registration table.

```
sipalg -calls <alg>
```

Show active calls table.

```
sipalg -session <alg>
```

Show active SIP sessions.

```
sipalg -connection <alg> [-num=<n>]
```

Show SIP connections.

```
sipalg -statistics[={SHOW | FLUSH}] <alg>
```

Show or flush SIP counters.

```
sipalg -snoop={ON | OFF | VERBOSE} [<ipaddr>] [-flags=<String>]
```

Control SIP snooping. Useful for troubleshooting SIP transactions. NOTE: 'verbose' option outputs a lot of information on the console which may lead to system instability. Use with caution.

**Options**

| | |
|---|---|
| **-calls** | Show active calls table. |
| **-connection** | Show SIP connections. |
| **-definition** | Show running ALG configuration parameters. |
| **-flags=<String>** | SIP snooping for certain levels. Expected number in hexadecimal notation. |
| **-num=<n>** | Limit list to <n> connections. (Default: 20) |
| **-registration[={SHOW | FLUSH}]** | Show or flush registration table. (Default: show) |
| **-session** | Show active SIP sessions. |
| **-snoop={ON | OFF | VERBOSE}** | Enable or disable SIP snooping. NOTE: 'verbose' option outputs a lot of information on the console which may lead to system instability. Use with caution. (Admin only) |
| **-statistics[={SHOW | FLUSH}]** | Show or flush SIP counters. (Default: show) |
| **<alg>** | SIP-ALG name. |
| **<alg>** | SIP-ALG name. |
| **<ipaddr>** | IP Address to snoop. |

## 2.2.83. slb

Manage and show status for SLB.

**Description**

Display SLB status and preform various related actions

**Usage**

```
slb
```

Display status for all policies.

```
slb -status <SLB Policy>
```

Display status for specific policy.

```
slb -suspend <SLB Policy> <ip address>
```

Suspend load distribution to server.

```
slb -resume <SLB Policy> <ip address>
```

Resume load distribution to server.

**Options**

| | |
|---|---|
| **-resume** | Resume load distribution to SLB server (maintenance off). |
| **-status** | Display status for specific SLB policy. |
| **-suspend** | Suspend load distribution to SLB server (maintenance on). |
| **<ip address>** | IP address. |
| **<SLB Policy>** | SLB policy. |

## 2.2.84. smtp

List SMTP LogReceiver sessions and send test mail.

**Description**

List SMTP sessions for configured SMTP LogReceivers and CLI SMTP sessions created when using "sendmail" to send test mail to SMTP LogReceiver. The temporary CLI sessions, marked with (CLI), has a lifetime of 300s.

**Usage**

```
smtp -list [-num[=<1...1000>]] [-verbose]
```

Show SMTP sessions.

```
smtp -verbose
```

Show SMTP sessions with verbose output.

```
smtp -stat
```

Show SMTP statistics.

```
smtp -sendmail -logreceiver=<Mail Alerting> [-message=<String>]
```

Send mail to specified SMTP LogReceiver.

**Options**

| | |
|---|---|
| **-list** | Show SMTP sessions. |
| **-logreceiver=<Mail Alerting>** | LogReceiver. |
| **-message=<String>** | Mail message. |
| **-num[=<1...1000>]** | Number of entries to list. (Default: 40) |
| **-sendmail** | Send test mail to SMTP LogReceiver. |
| **-stat** | Show SMTP statistics. |
| **-verbose** | Verbose output. |

# 2.2.85. sshserver

SSH Server.

**Description**

Show SSH Server status, or start/stop/restart SSH Server.

**Usage**

```
sshserver
```

Show server status and list all connected clients.

```
sshserver -status [-verbose]
```

Show server status and list all connected clients.

```
sshserver -keygen [-b=<bits>] [-t={RSA | DSA}]
```

Generate SSH Server private keys.

```
sshserver -restart <ssh server>
```

Restart SSH Server.

**Options**

| | |
|---|---|
| **-b=<bits>** | Bitsize. (Default: 1024) |
| **-keygen** | Generate SSH Server private keys. This operation may take a long time to finish, up to several minutes! |

| | |
|---|---|
| **-restart** | Stop and start the SSH Server. |
| **-status** | Show server status and list all connected clients. |
| **-t={RSA \| DSA}** | Type, (default: both RSA and DSA keys will be created). |
| **-verbose** | Verbose output. |
| **\<ssh server\>** | SSH Server. |

> ***Note***
> *Requires Administrator privileges.*

# 2.2.86. sslvpn

SSLVPN tunnels.

### Description

List running SSLVPN configurations, SSLVPN active tunnels and call information.

### Usage

```
sslvpn [-num=<n>]
```

### Options

| | |
|---|---|
| **-num=\<n\>** | Limit display to \<n\> entries. (Default: 20) |

# 2.2.87. stats

Display various general firewall statistics.

### Description

Display general information about the firewall, such as uptime, CPU load, resource consumption and other performance data.

### Usage

```
stats
```

# 2.2.88. sysmsgs

System messages.

### Description

Show contents of the FWLoader sysmsg buffer.

### Usage

```
sysmsgs
```

## 2.2.89. techsupport

Technical Support information.

### Description

Generate information useful for technical support.

Due to the large amount of output, this command might show a truncated result when execute from the local console.

### Usage

```
techsupport
```

## 2.2.90. time

Display current system time.

### Description

Display/set the system date and time.

### Usage

```
time
```

Display current system time.

```
time -verbose
```

Display current system time.

```
time -set <date> <time>
```

Set system local time: <YYYY-MM-DD> <HH:MM:SS>.

```
time -sync [-force]
```

Synchronize time with timeserver(s) (specified in settings).

**Options**

| | |
|---|---|
| **-force** | Force synchronization regardless of the MaxAdjust setting. |
| **-set** | Set system local time: <YYYY-MM-DD> <HH:MM:SS>. (Admin only) |
| **-sync** | Synchronize time with timeserver(s) (specified in settings). |
| **-verbose** | Show more information about time zone and DST. |
| **<date>** | Date YYYY-MM-DD. |
| **<time>** | Time HH:MM:SS. |

## 2.2.91. uarules

Show user authentication rules.

**Description**

Displays the contents of the user authentication ruleset.

**Example 2.17. Show a range of rules**

```
uarules -v 1-2,4-5
```

**Usage**

```
uarules [-verbose] [<Integer Range>]
```

**Options**

| | |
|---|---|
| **-verbose** | Verbose output. |
| **<Integer Range>** | Range of rules to list. |

## 2.2.92. updatecenter

Show autoupdate status and manage IDP/AV databases.

*92*

### Description

Show autoupdate mechanism status or force an update.

### Usage

```
updatecenter
```

Show update status and database information.

```
updatecenter -status[={ANTIVIRUS | IDP | ALL}] [-verbose]
```

Show update status and database information.

```
updatecenter -update[={ANTIVIRUS | IDP | ALL}]
```

Initiate an update check of the specified database.

```
updatecenter -removedb={ANTIVIRUS | IDP}
```

Remove the specified signature database.

```
updatecenter -servers
```

Show status of update servers.

### Options

| | |
|---|---|
| **-removedb={ANTIVIRUS \| IDP}** | Remove the database for the specified service. |
| **-servers** | Show status of update servers. |
| **-status[={ANTIVIRUS \| IDP \| ALL}]** | Show update status and database information. (Admin only; Default: all) |
| **-update[={ANTIVIRUS \| IDP \| ALL}]** | Force an update now for the specified service. (Admin only; Default: all) |
| **-verbose** | Show verbose status information. (Admin only) |

## 2.2.93. userauth

Show logged-on users.

### Description

Show currently logged-on users and other information. Also allows logged-on users to be forcibly logged out.

Note: In the user listing -list, only privileges actually used by the policy are displayed.

### Usage

*93*

```
userauth
```

List all authenticated users.

```
userauth -list [-num=<n>] [-blocked] [-verbose]
```

List all authenticated users.

```
userauth -privilege
```

List all known privileges (usernames and groups).

```
userauth -user [<user ip>]
```

Show all information for user(s) with this IP address.

```
userauth -remove [<user ip> [<Interface>]] [-all]
```

Forcibly log out an authenticated user.

**Options**

| | |
|---|---|
| **-all** | All users. |
| **-blocked** | List all blocked users. |
| **-list** | List all authenticated users. |
| **-num=<n>** | Limit list of authenticated users. (Default: 20) |
| **-privilege** | List all known privileges (usernames and groups). |
| **-remove** | Forcibly log out an authenticated user. (Admin only) |
| **-user** | Show all information for user(s) with this IP address. |
| **-verbose** | List all blocked users history. |
| **<Interface>** | Interface. |
| **<user ip>** | IP address for user(s). |

## 2.2.94. vlan

Show information about VLAN.

**Description**

Show list of attached Virtual LAN Interfaces, or in-depth information about a specified VLAN.

**Usage**

```
vlan
```

List attached VLANs.

```
vlan -num=<n> [-page[=<n>]]
```

Set number of display lines per page and display page.

```
vlan <Interface>
```

Display in-depth information about a VLAN interface, and/or the VLAN interfaces that are based on a specific interface.

**Options**

| | |
|---|---|
| **-num=<n>** | Limit display lines to <n> entries in page. (Default: 20) |
| **-page[=<n>]** | Set page <n> for lines to display. (Default: 1) |
| **<Interface>** | Display VLAN information about this interface. |

## 2.2.95. vpnstats

Alias for **ipsecstats**.

## 2.2.96. zonedefense

Zonedefense.

**Description**

Block/unblock IP addresses/net and ethernet addresses.

**Usage**

```
zonedefense [-save] [-blockip=<ip address>] [-blockenet=<ethernet
          address>] [-eraseip=<ip address>] [-eraseenet=<ethernet
          address>] [-status] [-show]
```

**Options**

| | |
|---|---|
| **-blockenet=<ethernet address>** | Block the specified ethernet address. (Admin only) |
| **-blockip=<ip address>** | Block the specified IP address/net. (Admin only) |
| **-eraseenet=<ethernet address>** | Unblock the specified ethernet address. |
| **-eraseip=<ip address>** | Unblock the specified IP address/net. |
| **-save** | Save the current zonedefense state on all switches. |
| **-show** | Show the current block database. |
| **-status** | Show the current status of the zonedefense state machine. |

# 2.3. Utility

## 2.3.1. geoip

Display GeoIP information.

### Description

Display status of GeoIP database and perform manual lookups.

### Usage

```
geoip
```

Display statistics.

```
geoip -filters [-num=<n>]
```

Display filter information.

```
geoip -status
```

Display statistics.

```
geoip -query <IPAddress>
```

Lookup IP address to GeoIP location.

### Options

| | |
|---|---|
| **-filters** | Display current active Geolocation Filters. |
| **-num=<n>** | List <n> entries. (Default: 20) |
| **-query** | Resolve domain name. |
| **-status** | Display status for GeoIP database. |
| **<IPAddress>** | IP address to resolve. |

## 2.3.2. ping

Ping host.

### Description

Sends one or more ICMP ECHO, TCP SYN or UDP datagrams to the specified IP address of a host. All datagrams are sent preloaded-style (all at once).

The data size `-length` given is the ICMP or UDP data size. 1472 bytes of ICMP data results in a 1500-byte IP datagram (1514 bytes ethernet).

**Usage**

```
ping [<String>] [-srcif=<interface>] [-srcip=<ip address>]
     [-pbr=<table>] [-count=<1...10>] [-length=<2...8192>]
     [-port=<0...65535>] [-srcport=<0...65535>] [-udp] [-tcp]
     [-tos=<0...255>] [-verbose] [-6]
```

**Options**

| | |
|---|---|
| **-6** | Force IPv6. |
| **-count=<1...10>** | Number of packets to send. (Default: 1) |
| **-length=<2...8192>** | Packet size. (Default: 4) |
| **-pbr=<table>** | Route using PBR Table. |
| **-port=<0...65535>** | Destination port of UDP or TCP ping. |
| **-srcif=<interface>** | Pass packet through the rule set, simulating that the packet was received by <srcif>. |
| **-srcip=<ip address>** | Use this source IP. |
| **-srcport=<0...65535>** | Source port of UDP or TCP ping. |
| **-tcp** | Send TCP ping. |
| **-tos=<0...255>** | Type of service. |
| **-udp** | Send UDP ping. |
| **-verbose** | Verbose (more information). |
| **<String>** | IP address or URL of host to ping. |

## 2.3.3. traceroute

Trace route.

**Description**

Print the route packets take to a network host.

**Usage**

```
traceroute <host> [-starthop=<1...255>] [-maxhops=<1...255>]
           [-timeout=<1...60000>] [-count=<1...10>]
           [-size=<0...32768>] [-pbr=<table>] [-srcip=<ip address>]
           [-noresolve] [-nodelay] [-6]
```

Trace using ICMP.

```
traceroute -tcp <host> [-port=<1...65535>] [-starthop=<1...255>]
```

```
            [-maxhops=<1...255>] [-timeout=<1...60000>]
            [-count=<1...10>] [-pbr=<table>] [-srcip=<ip address>]
            [-noresolve] [-nodelay] [-6]
```

Trace using TCP.

```
traceroute -udp <host> [-port=<1...65535>] [-starthop=<1...255>]
            [-maxhops=<1...255>] [-timeout=<1...60000>]
            [-count=<1...10>] [-size=<0...32768>] [-pbr=<table>]
            [-srcip=<ip address>] [-noresolve] [-nodelay] [-6]
```

Trace using UDP.

```
traceroute -stop
```

Stop trace.

**Options**

| | |
|---|---|
| **-6** | Force IPv6 if target is a FQDN. |
| **-count=<1...10>** | Number of queries to send for each hop. (Default: 3) |
| **-maxhops=<1...255>** | Maximum number of hosts to traverse in search of target. (Default: 30) |
| **-nodelay** | Send queries as fast as possible (may look like Denial of Service attack). |
| **-noresolve** | Disable reverse DNS lookup of hosts. |
| **-pbr=<table>** | Route using PBR Table. |
| **-port=<1...65535>** | Destination port. |
| **-size=<0...32768>** | Packet data size. (Default: 32) |
| **-srcip=<ip address>** | Use this source IP. |
| **-starthop=<1...255>** | Initial TTL value. (Default: 1) |
| **-stop** | Stop trace in progress. |
| **-tcp** | Use TCP instead of ICMP. |
| **-timeout=<1...60000>** | How many milliseconds to wait for each reply. (Default: 1000) |
| **-udp** | Use UDP instead of ICMP. |
| **<host>** | IP address or FQDN of host to trace. |

# 2.4. Misc

## 2.4.1. clear

Clears the screen.

### Description

Clears the screen.

### Usage

```
clear
```

## 2.4.2. echo

Print text.

### Description

Print text to the console.

---

**Example 2.18. Hello World**

```
echo Hello World
```

---

### Usage

```
echo [<String>]...
```

### Options

**<String>**                                   Text to print.

## 2.4.3. help

Show help for selected topic.

### Description

The help system contains information about commands and configuration object types.

The fastest way to get help is to simply type **help** followed by the topic that you want help with. A topic can be for example a command name (e.g. **set**) or the name of a configuration object type (e.g. User).

When you don't know the name of what you are looking for you can specify the category of the wanted topic with the `-category` option and use tab-completion to display a list of matching topics.

**Usage**

```
help
```

List commands alphabetically.

```
help <Topic>
```

Display help about selected topic from any category.

```
help -category={COMMANDS | TYPES} [<Topic>]
```

Display help from a specific topic category.

**Options**

**-category={COMMANDS | TYPES}**              Topic category.

**<Topic>**                                                Help topic.

## 2.4.4. history

Dump history to screen.

**Description**

List recently typed commands that have been stored in the command history.

**Usage**

```
history
```

## 2.4.5. logsnoop

Display and filter system log messages.

**Description**

The logsnoop command can be used to display system log events. The source of the log events can be MemLog, real-time or both MemLog followed by real-time logs.

MemLog searching will only be functioning if a LogReceiverMemory object has been configured.

Since the system log rate may be high, displaying real time logs must be done with some caution. For this purpose, it is possible to limit the real time log display rate.

When filtering for log messages to display, there are many parameters that can be filtered on. The most powerful filtering tool is the wildcard matching in which the charcter '*' is interpreted as none/many characters and '?' as any single character.

It should be noted that all log filtering will have a negative effect on system performance.

**Example 2.19. Show log message having 'warning' followed by 'udp' somewhere in the message**

```
:/> logsnoop -on -pattern=*warning*udp*
```

**Example 2.20. Rate limit log flow to five logs per second**

```
:/> logsnoop -on -rate=5
```

**Example 2.21. Show logs from the memlog buffer**

```
:/> logsnoop -on -source=memlog
```

**Example 2.22. Show logs having a source IP value**

```
:/> logsnoop -on -srcip=0.0.0.0/0
```

**Example 2.23. Show logs having a severity of warning or higher**

```
:/> logsnoop -on -severity=warning
```

**Usage**

```
logsnoop -on [-source={MEMLOG | REALTIME | BOTH}]
        [-category=<String>] [-logid=<Integer>] [-event=<String>]
        [-action={NONE | DROP | ALLOW | BLOCK | REJECT |
        <String>}] [-severity={EMERGENCY | ALERT | CRITICAL |
        ERROR | WARNING | NOTICE | INFO | DEBUG}]
        [-starttime=<DateTime>] [-endtime=<DateTime>]
        [-pattern=<String>] [-srcip=<IPAddress>]
        [-destip=<IPAddress>] [-srcport=<0...65535>]
        [-destport=<0...65535>] [-srcif=<Interface>]
```

```
          [-destif=<Interface>] [-ipproto={TCP | UDP | ICMP |
          <String>}] [-rate=<Integer>] [-num=<Integer>]
```

Start log session.

```
logsnoop -off
```

Stop log session.

```
logsnoop
```

Show log snoop status.

**Options**

| | |
|---|---|
| **-action={NONE \| DROP \| ALLOW \| BLOCK \| REJECT \| <String>}** | Log action to filter on. |
| **-category=<String>** | Log category to filter on. |
| **-destif=<Interface>** | Destination interface to filter on. |
| **-destip=<IPAddress>** | Destination IP address or network to filter on. |
| **-destport=<0...65535>** | Destination port to filter on. |
| **-endtime=<DateTime>** | End time of log snooping. Format: year-month-day [HH:MM:SS]. |
| **-event=<String>** | Log event to filter on. |
| **-ipproto={TCP \| UDP \| ICMP \| <String>}** | Protocol to filter on. |
| **-logid=<Integer>** | Numeric log ID to filter on. |
| **-num=<Integer>** | Total log limit, number of logs. |
| **-off** | Stop log session. |
| **-on** | Start log session. |
| **-pattern=<String>** | Free text filter supporting wildcards. |
| **-rate=<Integer>** | Rate limit, logs/sec. Only applicable for real time logs. |
| **-severity={EMERGENCY \| ALERT \| CRITICAL \| ERROR \| WARNING \| NOTICE \| INFO \| DEBUG}** | Log severity to filter on. Equal or higher severity matches. |
| **-source={MEMLOG \| REALTIME \| BOTH}** | Log source. (Default: realtime) |
| **-srcif=<Interface>** | Source interface to filter on. |
| **-srcip=<IPAddress>** | Source IP address or network to filter on. |
| **-srcport=<0...65535>** | Source port to filter on. |
| **-starttime=<DateTime>** | Start time of log snooping. Format: year-month-day [HH:MM:SS]. |

*Note*
*Requires Administrator privileges.*

## 2.4.6. ls

Lists device data accessible by SCP.

### Description

Lists device data which are available through SCP.

---

**Example 2.24. Transfer script files to and from the device**

```
Upload:    scp myscript user@sgw-ip:script/myscript
Download:  scp user@sgw-ip:script/myscript ./myscript
```

---

In addition to the files listed it is possible to upload license, certificates and ssh public key files.

---

**Example 2.25. Upload license data**

```
scp licence.lic user@sgw-ip:license.lic
```

---

Certificates and ssh client key objects are created if they do not exist.

---

**Example 2.26. Upload certificate data**

```
scp certificate.cer user@sgw-ip:certificate/certificate_name
scp certificate.key user@sgw-ip:certificate/certificate_name
```

---

**Example 2.27. Upload ssh public key data**

```
scp sshkey.pub user@sgw-ip:sshclientkey/sshclientkey_name
```

---

### Usage

### Options

| | |
|---|---|
| **-long** | Enable long listing format. |
| **<File>** | File to list. |

## 2.4.7. script

Handle CLI scripts.

### Description

Run, create, show, store or delete script files.

Script files are transferred to and from the device by the SCP protocol. On the device they are stored in the "/script" folder.

---

**Example 2.28. Execute script**

```
"script.sgs":
add IP4Address Name=$1 Address=$2 Comment="$0: \$100".
:/> script -execute -name=script.sgs ip_test 127.0.0.1
is executed as line:
add IP4Address Name=ip_test Address=127.0.0.1 Comment="script.sgs: $100"
```

---

### Usage

```
script -create [[<Category>] <Type> [<Identifier>]] [-name=<Name>]
```

Create configuration script from specified object, class or category.

```
script -execute [-verbose] [-force] [-quiet] -name=<Name>
       [<Parameters>]...
```

Execute script.

```
script -show [-all] [-name=<Name>]
```

Show script in console window.

```
script -store [-all] [-name=<Name>]
```

Store a script to persistent storage.

```
script -remove [-all] [-name=<Name>]
```

Remove script.

```
script
```

List script files.

### Options

**-all**                                  Apply to all scripts.

**-create**                               Create configuration script from specified object, class or category.

**-execute**                              Execute script.

| | |
|---|---|
| **-force** | Force script execution. |
| **-name=<Name>** | Name of script. |
| **-quiet** | Quiet script execution. |
| **-remove** | Remove script. |
| **-show** | Show script in console window. |
| **-store** | Store a script to persistent storage. |
| **-verbose** | Verbose mode. |
| **<Category>** | Category that groups object types. |
| **<Identifier>** | The property that identifies the configuration object. May not be applicable depending on the specified <Type>. |
| **<Parameters>** | List of input arguments. |
| **<Type>** | Type of configuration object to perform operation on. |

**Note**
*Requires Administrator privileges.*

# Chapter 3: Configuration Reference

# 3.1. Access

### Description

Use an access rule to allow or block specific source IP addresses on a specific interface.

### Properties

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the object. |

| | |
|---|---|
| **Action** | Accept, Expect or Drop. (Default: Drop) |
| **Interface** | The interface the packet must arrive on for this rule to be carried out. Exception: the Expect rule. |
| **Network** | The IP span that the sender must belong to for this rule to be carried out. |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

### Note

*If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.2. Address

This is a category that groups the following object types.

# 3.2.1. AddressFolder

### Description

An address folder can be used to group related address objects for better overview.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the network object. (Identifier) |
| **Comments** | Text describing the current object. (Optional) |

## 3.2.1.1. FQDNGroup

### Description

An FQDN Address Group is used for combining several FQDN Address objects for simplified management.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the network object. (Identifier) |
| **Members** | Group members. |
| **Comments** | Text describing the current object. (Optional) |

## 3.2.1.2. FQDNAddress

### Description

Use an FQDN Address item to define a name for a domain name.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the network object. (Identifier) |
| **Address** | FQDN, e.g. "www.example.com" or "*.example.com". |
| **ActiveAddress** | The IP addresses resolved from the name server. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

### 3.2.1.3. EthernetAddress

**Description**

Use an Ethernet Address item to define a symbolic name for an Ethernet MAC address.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the network object. (Identifier) |
| **Address** | Ethernet MAC address, e.g. "12-34-56-78-AB-CD". |
| **Comments** | Text describing the current object. (Optional) |

### 3.2.1.4. EthernetAddressGroup

**Description**

An Ethernet Address Group is used for combining several Ethernet Address objects for simplified management.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the network object. (Identifier) |
| **Members** | Group members. |
| **Comments** | Text describing the current object. (Optional) |

### 3.2.1.5. IP6HAAddress

**Description**

Use an IP6 HA Address item to define a name for a specific IP6 host, network or range for each node in a high availability cluster.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the network object. (Identifier) |
| **Address** | An IP address with one instance for each node in the high availability cluster. |
| **Comments** | Text describing the current object. (Optional) |

### 3.2.1.6. IP6Group

**Description**

An IP6 Address Group is used for combining several IP6 Address objects for simplified management.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the network object. (Identifier) |
| **Members** | Group members. |
| **Comments** | Text describing the current object. (Optional) |

## 3.2.1.7. IP4Address

**Description**

Use an IP4 Address item to define a name for a specific IP4 host, network or range.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the network object. (Identifier) |
| **Address** | IP address, e.g. "172.16.50.8", "192.168.7.0/24" or "172.16.25.10-172.16.25.50". |
| **ActiveAddress** | The dynamically set address used by e.g. DHCP enabled Ethernet interfaces. (Optional) |
| **UserAuthGroups** | Groups and user names that belong to this object. Objects that filter on credentials can only be used as source networks and destinations networks in rules. (Optional) |
| **NoDefinedCredentials** | If this property is enabled the object requires user authentication, but has no credentials (user names or groups) defined. This means that the object only requires that a user is authenticated, but ignores any kind of group membership. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

## 3.2.1.8. IP4Group

**Description**

An IP4 Address Group is used for combining several IP4 Address objects for simplified management.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the network object. (Identifier) |
| **Members** | Group members. |
| **UserAuthGroups** | Groups and user names that belong to this object. Objects that filter on credentials can only be used as source networks and destinations networks in rules. (Optional) |
| **NoDefinedCredentials** | If this property is enabled the object requires user authentication, but has no credentials (user names or groups) defined. This means that the object only requires that a user is authenticated, but ignores any kind of group membership. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

## 3.2.1.9. IP4HAAddress

### Description

Use an IP4 HA Address item to define a name for a specific IP4 host for each node in a high availability cluster.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the network object. (Identifier) |
| **Address** | An IP address with one instance for each node in the high availability cluster. |
| **UserAuthGroups** | Groups and user names that belong to this object. Objects that filter on credentials can only be used as source networks and destinations networks in rules. (Optional) |
| **NoDefinedCredentials** | If this property is enabled the object requires user authentication, but has no credentials (user names or groups) defined. This means that the object only requires that a user is authenticated, but ignores any kind of group membership. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

## 3.2.1.10. IP6Address

### Description

Use an IP6 Address item to define a name for a specific IP6 host, network or range.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the network object. |

116

|  |  |
|---|---|
|  | (Identifier) |
| **Address** | IPv6 address, e.g. "2001:DB8::/32". |
| **ActiveAddress** | The dynamically set address used by e.g. DHCPv6 enabled Ethernet interfaces. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

## 3.2.2. EthernetAddress

The definitions here are the same as in Section 3.2.1.3, "EthernetAddress" .

## 3.2.3. EthernetAddressGroup

The definitions here are the same as in Section 3.2.1.4, "EthernetAddressGroup" .

## 3.2.4. IP4Address

The definitions here are the same as in Section 3.2.1.7, "IP4Address" .

## 3.2.5. IP4Group

The definitions here are the same as in Section 3.2.1.8, "IP4Group" .

## 3.2.6. IP4HAAddress

The definitions here are the same as in Section 3.2.1.9, "IP4HAAddress" .

## 3.2.7. IP6Address

The definitions here are the same as in Section 3.2.1.10, "IP6Address" .

## 3.2.8. IP6Group

The definitions here are the same as in Section 3.2.1.6, "IP6Group" .

## 3.2.9. IP6HAAddress

The definitions here are the same as in Section 3.2.1.5, "IP6HAAddress" .

# 3.3. AdvancedScheduleProfile

**Description**

An advanced schedule profile contains definitions of occurrences used by various policies in the system.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the service. (Identifier) |
| **Comments** | Text describing the current object. (Optional) |

# 3.3.1. AdvancedScheduleOccurrence

**Description**

An advanced schedule occurrence specifies an occurrence that should happen between certain times for days in month/week

**Properties**

| | |
|---|---|
| **StartTime** | Start time in the format HH:MM, for example 13:30. |
| **EndTime** | End time in the format HH:MM, for example 14:15. |
| **Occurrence** | Specify type of occurrence. (Default: Weekly) |
| **Weekly** | Specifies days in week the schedule occurrence should be activated. Monday corresponds to 1 and Sunday 7. (Default: 1-7) |
| **Monthly** | Specifies days in month the schedule occurrence should be activated. The schedule only occurs at days that exists in the month. (Default: 1-31) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.

# 3.4. ALG

This is a category that groups the following object types.

# 3.4.1. ALG_FTP

### Description

Use an FTP Application Layer Gateway to manage FTP traffic through the system.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the ALG. (Identifier) |
| **AllowServerPassive** | Allow server to use passive mode (unsafe for server). (Default: No) |
| **ServerPorts** | Server data ports. (Default: 1024-65535) |
| **AllowClientActive** | Allow client to use active mode (unsafe for client). (Default: No) |
| **ClientPorts** | Client data ports. (Default: 1024-65535) |
| **AllowUnknownCommands** | Allow unknown commands. (Default: No) |
| **AllowSITEEXEC** | Allow SITE EXEC. (Default: No) |
| **MaxLineLength** | Maximum line length in control channel. (Default: 256) |
| **MaxCommandRate** | Maximum number of commands per second. (Default: 20) |
| **Allow8BitStrings** | Allow 8-bit strings in control channel. (Default: Yes) |
| **AllowResumeTransfer** | Allow RESUME even in case of content scanning. (Default: No) |
| **Antivirus** | Disabled, Audit or Protect. (Default: Disabled) |
| **ScanExclude** | List of files to exclude from antivirus scanning. (Optional) |
| **CompressionRatio** | A compression ratio higher than this value will trigger the action in Compression Ratio Action, a value of zero will disable all compression checks. (Default: 20) |
| **CompressionRatioAction** | The action to take when high compression threshold is violated, all actions are logged. (Default: Drop) |
| **AllowEncryptedZip** | Allow encrypted zip files, even though the contents can not be scanned. (Default: No) |
| **MaxArchiveDepth** | The maximum number of archive "layers" that the antivirus engine will extract. (Default: 5) |

| | |
|---|---|
| **ZDEnabled** | Enable ZoneDefense Block. (Default: No) |
| **ZDNetwork** | Hosts within this network will be blocked at switches if a virus is found. |
| **FailModeBehavior** | Standard behaviour on error: Allow or Deny. (Default: Deny) |
| **FileListType** | Specifies if the file list contains files to allow or deny. (Default: Block) |
| **File** | List of file types to allow or deny. (Optional) |
| **VerifyContentMimetype** | Verify that file extentions correspond to the MIME type. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

# 3.4.2. ALG_H323

### Description

Use an H.323 Application Layer Gateway to manage H.323 multimedia traffic.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the ALG. (Identifier) |
| **AllowTCPDataChannels** | Allow TCP data channels (T.120). (Default: Yes) |
| **MaxTCPDataChannels** | Maximum number of TCP data channels per call. (Default: 10) |
| **TranslateAddresses** | Automatic or Specific. (Default: Automatic) |
| **TranslateLogicalChannelAddresses** | Translate logical channel addresses. (Default: Yes) |
| **MaxGKRegLifeTime** | Max Gatekeeper Registration Lifetime. (Default: 1800) |
| **ChannelSetupMode** | Channel connection setup mode. (Default: Optimistic) |
| **Comments** | Text describing the current object. (Optional) |

# 3.4.3. ALG_HTTP

### Description

Use an HTTP Application Layer Gateway to filter HTTP traffic.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the ALG. (Identifier) |

| | |
|---|---|
| **AllowedProtocols** | HTTP and/or HTTPS. (Default: HTTP) |
| **RemoveCookies** | Remove cookies. (Default: No) |
| **RemoveScripts** | Remove Javascript/VBScript. (Default: No) |
| **RemoveApplets** | Remove Java applets. (Default: No) |
| **RemoveActiveX** | Remove ActiveX objects (including Flash). (Default: No) |
| **VerifyUTF8URL** | Verify that URLs does not contain invalid UTF8 encoding. (Default: No) |
| **BlackURLDisplayReason** | Message to show when there is an attempt to access a blacklisted site. (Optional) |
| **HTTPBanners** | HTTP ALG HTML Banners. (Default: Default) |
| **MaxDownloadSize** | The maximum allowed file size in kB. (Optional) |
| **FileListType** | Specifies if the file list contains files to allow or deny. (Default: Block) |
| **File** | List of file types to allow or deny. (Optional) |
| **VerifyContentMimetype** | Verify that file extentions correspond to the MIME type. (Default: No) |
| **Antivirus** | Disabled, Audit or Protect. (Default: Disabled) |
| **ScanExclude** | List of files to exclude from antivirus scanning. (Optional) |
| **CompressionRatio** | A compression ratio higher than this value will trigger the action in Compression Ratio Action, a value of zero will disable all compression checks. (Default: 20) |
| **CompressionRatioAction** | The action to take when high compression threshold is violated, all actions are logged. (Default: Drop) |
| **AllowEncryptedZip** | Allow encrypted zip files, even though the contents can not be scanned. (Default: No) |
| **MaxArchiveDepth** | The maximum number of archive "layers" that the antivirus engine will extract. (Default: 5) |
| **ZDEnabled** | Enable ZoneDefense Block. (Default: No) |
| **ZDNetwork** | Hosts within this network will be blocked at switches if a virus is found. |
| **FailModeBehavior** | Standard behaviour on error: Allow or Deny. (Default: Deny) |
| **AllowFilteringReclassification** | Show reclassification link for blocked sites. (Default: No) |
| **WebContentFilteringMode** | Disabled, Audit or Enable. (Default: Disabled) |
| **FilteringCategories** | Web content categories to block. (Optional) |

*121*

| | |
|---|---|
| **NonManagedAction** | Action to take for content that hasn't been classified. (Default: Allow) |
| **AllowFilteringOverride** | Allow the user to display a blocked site. (Default: No) |
| **OverrideUpdateOnAccess** | Restart the override timer on each new access to disallowed categories. (Default: Yes) |
| **OverrideTimeToLive** | Seconds that all disallowed categories will be allowed for the host that requested the override. (Default: 300) |
| **Comments** | Text describing the current object. (Optional) |

## 3.4.3.1. ALG_HTTP_URL

**Description**

Blacklist URLs to deny access to complete sites, to file types by extension, or to URLs with certain words in them.

**Properties**

| | |
|---|---|
| **Action** | Whitelist or Blacklist. (Default: Blacklist) |
| **URL** | Specifies the URL to blacklist or whitelist. |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.

## 3.4.4. ALG_POP3

**Description**

Use an POP3 Application Layer Gateway to manage POP3 traffic through the system.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the ALG. (Identifier) |
| **BlockUserPass** | Block clients from sending USER and PASS command. (Default: No) |
| **HideUser** | Prevent server from revealing that a user name do not exist. (Default: No) |
| **AllowUnknownCommands** | Allow unknown commands. (Default: No) |

| | |
|---|---|
| **FileListType** | Specifies if the file list contains files to allow or deny. (Default: Block) |
| **File** | List of file types to allow or deny. (Optional) |
| **VerifyContentMimetype** | Verify that file extentions correspond to the MIME type. (Default: No) |
| **Antivirus** | Disabled, Audit or Protect. (Default: Disabled) |
| **ScanExclude** | List of files to exclude from antivirus scanning. (Optional) |
| **CompressionRatio** | A compression ratio higher than this value will trigger the action in Compression Ratio Action, a value of zero will disable all compression checks. (Default: 20) |
| **CompressionRatioAction** | The action to take when high compression threshold is violated, all actions are logged. (Default: Drop) |
| **AllowEncryptedZip** | Allow encrypted zip files, even though the contents can not be scanned. (Default: No) |
| **MaxArchiveDepth** | The maximum number of archive "layers" that the antivirus engine will extract. (Default: 5) |
| **ZDEnabled** | Enable ZoneDefense Block. (Default: No) |
| **ZDNetwork** | Hosts within this network will be blocked at switches if a virus is found. |
| **FailModeBehavior** | Standard behaviour on error: Allow or Deny. (Default: Deny) |
| **Comments** | Text describing the current object. (Optional) |

## 3.4.5. ALG_PPTP

**Description**

Use a PPTP Application Layer Gateway to manage PPTP traffic through the system.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the ALG. (Identifier) |
| **EchoTimeout** | Specifies idle timeout for Echo messages in the PPTP tunnel. (Default: 0) |
| **IdleTimeout** | SPecifies idle timeout for user traffic in the PPTP tunnel. (Default: 0) |
| **Comments** | Text describing the current object. (Optional) |

## 3.4.6. ALG_SIP

### Description

Use a SIP ALG to manage SIP based multimedia sessions.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the ALG. (Identifier) |
| **MaxSessionsPerId** | Maximum number of sessions per SIP URI. (Default: 5) |
| **MaxRegistrationTime** | The maximum allowed time in seconds between registration requests. (Default: 3600) |
| **SipSignalTmout** | Timeout value for last seen SIP message (in seconds). (Default: 43200) |
| **DataChannelTmout** | Timeout value for data channel (in seconds). (Default: 120) |
| **AllowMediaByPass** | Allow clients to exchange media directly when possible. (Default: Yes) |
| **AllowTCPDataChannels** | Allow TCP data channels. (Default: Yes) |
| **MaxTCPDataChannels** | Maximum number of TCP data channels per call. (Default: 5) |
| **Comments** | Text describing the current object. (Optional) |

# 3.4.7. ALG_SMTP

### Description

Use an SMTP Application Layer Gateway to manage SMTP traffic through the system.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the ALG. (Identifier) |
| **VerifySenderEmail** | Check emails for mismatching SMTP command From address and email header From address. (Default: No) |
| **VerifySenderEmailAction** | ...and block them. (Default: Deny) |
| **VerifySenderEmailSpamTag** | Spam Tag that is inserted into the subject. (Default: "*** SPAM *** ") |
| **VerifySenderEmailDomainOnly** | Only check domain names in email From addresses. (Default: No) |
| **MaxEmailPerMinute** | Specifies the maximum amount of emails per minute from the same host. (Optional) |
| **MaxEmailSize** | Specifies the maximum allowed email size in kB. (Optional) |

| | |
|---|---|
| **FileListType** | Specifies if the file list contains files to allow or deny. (Default: Block) |
| **File** | List of file types to allow or deny. (Optional) |
| **VerifyContentMimetype** | Verify that file extentions correspond to the MIME type. (Default: No) |
| **Antivirus** | Disabled, Audit or Protect. (Default: Disabled) |
| **ScanExclude** | List of files to exclude from antivirus scanning. (Optional) |
| **CompressionRatio** | A compression ratio higher than this value will trigger the action in Compression Ratio Action, a value of zero will disable all compression checks. (Default: 20) |
| **CompressionRatioAction** | The action to take when high compression threshold is violated, all actions are logged. (Default: Drop) |
| **AllowEncryptedZip** | Allow encrypted zip files, even though the contents can not be scanned. (Default: No) |
| **MaxArchiveDepth** | The maximum number of archive "layers" that the antivirus engine will extract. (Default: 5) |
| **ZDEnabled** | Enable ZoneDefense Block. (Default: No) |
| **ZDNetwork** | Hosts within this network will be blocked at switches if a virus is found. |
| **FailModeBehavior** | Standard behaviour on error: Allow or Deny. (Default: Deny) |
| **DNSBL** | Disable or Enable DNSBL. (Default: No) |
| **SpamThreshold** | Spam Threshold defines when an email should be considered as Spam. (Default: 10) |
| **DropThreshold** | Drop Threshold defines when an email should be considered malicious and be dropped. (Default: 20) |
| **SpamTag** | Spam Tag that is inserted into the subject for an email considered as Spam or malicious. (Default: "*** SPAM *** ") |
| **ForwardBlockedMail** | Forward blocked mails to DropAddress. (Default: No) |
| **DropAddress** | Email address that emails reaching the drop threshold will be rerouted to. |
| **AppendTXT** | Use TXT records (will only be used if reaching the drop threshold). (Default: No) |
| **CacheSize** | Size of the IP Cache of checked sender IP addresses. (Default: 0) |
| **CacheTimeout** | Timeout in seconds before a cached IP address is removed. (Default: 600) |

*125*

| DNSBlackLists | Specifies the BlackList domain and its weighted value. |
| Comments | Text describing the current object. (Optional) |

## 3.4.7.1. ALG_SMTP_Email

**Description**

Used to whitelist or blacklist an email sender/recipient.

**Properties**

| Type | Specifies if the email address is the sender or the recipient. (Default: Sender) |
| Action | Specifies whether to whitelist (allow) or blacklist (deny) this address. (Default: Blacklist) |
| Email | Specifies the recipient email to blacklist or whitelist. |
| Comments | Text describing the current object. (Optional) |

> **Note**
> If no *Index* is specified when creating an instance of this type, the object will be placed last in the list and the *Index* will be equal to the length of the list.

## 3.4.8. ALG_TFTP

**Description**

Use an TFTP Application Layer Gateway to manage TFTP traffic through the system.

**Properties**

| Name | Specifies a symbolic name for the ALG. (Identifier) |
| AllowedCommands | Specifies allowed commands. (Default: ReadWrite) |
| RemoveOptions | Remove option part from request packet. (Default: No) |
| AllowUnknownOptions | Allow unknown options in request packet. (Default: No) |
| MaxBlocksize | Max value for the blksize option. (Optional) |
| MaxFileTransferSize | Max size for transferred file. (Optional) |
| BlockDirectoryTraversal | Prevent directory traversal (consecutive dots in filenames). (Default: No) |

*126*

**Comments**                                                   Text describing the current object. (Optional)

# 3.4.9. ALG_TLS

### Description

Use a TLS Application Layer Gateway to manage TLS traffic through the system.

**Properties**

**Name**                                                   Specifies a symbolic name for the ALG. (Identifier)

**HostCert**                                               Specifies the host certificate.

**RootCert**                                               Specifies the root certificates. (Optional)

**Comments**                                               Text describing the current object. (Optional)

# 3.5. AntiVirusPolicy

## Description

An Anti-Virus Profile can be used by one or many IP Policies which has its service object configured with a protocol that supports anti-virus scanning (HTTP, FTP, POP3, SMTP, IMAP).

## Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the Profile. (Identifier) |
| **AuditMode** | Anti-Virus audit mode. (Default: No) |
| **ScanExclude** | List of files to exclude from antivirus scanning. (Optional) |
| **CompressionRatio** | A compression ratio higher than this value will trigger the action in Compression Ratio Action, a value of zero will disable all compression checks. (Default: 20) |
| **CompressionRatioAction** | The action to take when high compression threshold is violated, all actions are logged. (Default: Drop) |
| **AllowEncryptedZip** | Allow encrypted zip files, even though the contents can not be scanned. (Default: No) |
| **MaxArchiveDepth** | The maximum number of archive file "layers" that the antivirus engine will extract. (Default: 5) |
| **ZDEnabled** | Enable ZoneDefense Block. (Default: No) |
| **ZDNetwork** | Hosts within this network will be blocked at switches if a virus is found. |
| **FailModeBehavior** | General behavior when anti-virus scanning fails. The data can either be allowed or denied. (Default: Deny) |
| **Comments** | Text describing the current object. (Optional) |

# 3.6. AppControlSettings

### Description

Settings related to the Application Control functionality.

### Properties

**MaxUnclassifiedPackets**

Maximum number of packets in one direction on a connection before the application will be forced to unknown. (Default: 5)

**MaxUnclassifiedBytes**

Maximum number of bytes transfered in one direction on a connection before the application will be forced to unknown. (Default: 7500)

**RestartOnFatalFailure**

Restart the device automatically if a fatal failure occurs that disables Application Control. (Default: No)

> ### Note
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.7. ApplicationRuleSet

**Description**

An Application Rule Set contains a list of Application Rules and some settings and can be used by one or more IP rules/IP Policies to configure Application Control on the traffic matching those IP Rules/IP Policies.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the Profile. (Identifier) |
| **DefaultAction** | Default action if nothing in the rule list matches. (Default: Deny) |
| **UseCustomLimits** | Use custom limits for unclassified traffic in this ruleset instead of the default limits specified in the advanced settings. (Default: No) |
| **MaxUnclassifiedPackets** | Maximum number of packets in one direction on a connection before the application will be forced to unknown. (Default: 5) |
| **MaxUnclassifiedBytes** | Maximum number of bytes transfered in one direction on a connection before the application will be forced to unknown. (Default: 7500) |
| **StrictHTTP** | Handle plain http more strictly to avoid leaking generic http services when only specific http services should be allowed. (Default: Yes) |
| **Comments** | Text describing the current object. (Optional) |

# 3.7.1. ApplicationRule

**Description**

An application rule specifies what action to perform on applications that matches the specified filter criteria.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the Profile. |
| **Action** | Action for matched application. (Default: Allow) |
| **AppFilter** | Application filter. |
| **ApplicationContent** | Extended logging and policy for application attributes. (Default: []) |
| **UserAuthGroups** | Groups and user names that belong to this object. (Optional) |
| **ForwardChain** | Specifies one or more pipes to be used for forward |

| | |
|---|---|
| | traffic. (Optional) |
| **ReturnChain** | Specifies one or more pipes to be used for return traffic. (Optional) |
| **Precedence** | Specifies what precedence should be assigned to the packets before sent into a pipe. (Default: FromPipe) |
| **FixedPrecedence** | Specifies the fixed precedence. |
| **Comments** | Text describing the current object. (Optional) |

### Note

*If no* Index *is specified when creating an instance of this type, the object will be placed last in the list and the* Index *will be equal to the length of the list.*

# 3.8. ARPND

## Description

Use an ARP/Neighbor Discovery entry to publish additional IP addresses and/or MAC addresses on a specified interface.

## Properties

| | |
|---|---|
| **Mode** | Static, Publish or XPublish. (Default: Publish) |
| **Interface** | Indicates the interface to which the ARP entry applies; e.g. the interface the address shall be published on. |
| **IP** | The IP address to be published or statically bound to a hardware address. |
| **MACAddress** | The hardware address associated with the IP address. (Default: 00-00-00-00-00-00) |
| **Comments** | Text describing the current object. (Optional) |

> ***Note***
> *If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.9. ARPNDSettings

**Description**

Advanced ARP/Neighbor Discovery-table settings.

**Properties**

| | |
|---|---|
| **ARPMatchEnetSender** | The Ethernet Sender address matching the hardware address in the ARP data. (Default: DropLog) |
| **ARPQueryNoSenderIP** | If the IP source address of an ARP query (NOT response!) is "0.0.0.0". (Default: DropLog) |
| **ARPSenderIP** | The IP Source address in ARP packets. (Default: Validate) |
| **UnsolicitedARPReplies** | Unsolicited ARP replies. (Default: DropLog) |
| **ARPRequests** | Specifies whether or not the ARP requests should automatically be added to or update the ARP table. (Default: Drop) |
| **ARPChanges** | ARP packets that would cause an entry to be changed. (Default: AcceptLog) |
| **StaticARPChanges** | ARP packets that would cause static entries to be changed. (Default: DropLog) |
| **ARPExpire** | Lifetime of an ARP entry in seconds. (Default: 900) |
| **ARPExpireUnknown** | Lifetime of an "unknown" ARP entry in seconds. (Default: 3) |
| **ARPMulticast** | ARP packets claiming to be multicast addresses; may need to be enabled for some load balancers/redundancy solutions. (Default: DropLog) |
| **ARPBroadcast** | ARP packets claiming to be broadcast addresses; should never need to be enabled. (Default: DropLog) |
| **ARPCacheSize** | Number of ARP entries in cache, total. (Default: 4096) |
| **ARPHashSize** | Number of ARP hash buckets per physical interface. (Default: 512) |
| **ARPHashSizeVLAN** | Number of ARP hash buckets per VLAN interface. (Default: 64) |
| **ARPIPCollision** | Behavior when receiving an ARP request with a sender IP colliding with the one used on the receive interface. (Default: Drop) |
| **ARPLogResolveSuccess** | Specifies whether or not to log when ARP Resolve succeeds. (Default: No) |

**LogResolveFailure**
Specifies whether or not to log failed ARP Resolves. (Default: Yes)

**NDRateLimit**
Rate limit originated ND packets. (Default: 1000)

**MaxAnycastDelayTime**
Randomized time to delay proxied and anycast advertisements. (Default: 100)

**NDMatchEnetSender**
Ignore ND packets with mismatching sender- and options MAC-addresses. (Default: Yes)

**NDValSenderIP**
Validate the IP source address of the ND packet. (Default: Yes)

**NDLogResolveSuccess**
Specifies whether or not to log when ND Resolve succeeds. (Default: No)

**NDChanges**
Action to take when ND packets are received that would modify an existing entry. (Default: FavorOld)

**StaticNDChanges**
Action to take when ND packets are received that would modify a static entry. (Default: DropLog)

**NDValidation**
Action to take when the stateless validation of a ND packet fail. (Default: DropLog)

**NDCacheSize**
Number of cached IP/L2 address tuples. (Per iface). (Default: 1024)

**NDMaxMulticastSolicit**
Number of Neighbor Solicitations before giving up address resolution. (Default: 3)

**NDMaxUnicastSolicit**
Number of Neighbor Solicitations before giving up a zombie during dead peer detection. (Default: 3)

**NDBaseReachableTime**
Multiple of randomized time factor in seconds, resulting in the time before a ND entry becomes a zombie. (Default: 30)

**NDDelayFirstProbeTime**
Time in seconds for a cache entry to go from DELAY to PROBE state unless resolved. (Default: 5)

**NDRetransTimer**
Number of seconds between each Neighbor Solicitation during address resolution and dead peer detection. (Default: 1)

**RAMaxInterval**
Maximum time between sending unsolicited multicast Router Advertisement. (Default: 600s). (Default: 600)

**RAMinInterval**
Minimum time between sending unsolicited multicast Router Advertisement. Will be automatically adjusted if set to less than 3 seconds or greater than .75 * Max RA Interval). (Default: 200)

**RAAutoLifetime**
Auto adjust the Router Lifetime field using the following formula; 3 * Max RA Interval. (Default: Yes)

**RADefaultLifetime**
The value to be placed in the Router Lifetime field of Router Advertisements sent from the SGW, in seconds. (Default: 1800s). (Default: 1800)

| | |
|---|---|
| **RAReachableTime** | The value to be placed in the Reachable Time field in the Router Advertisement messages SGW. The value zero means unspecified. (Default: 0s). (Default: 0) |
| **RARetransTimer** | The value to be placed in the Retrans Timer field in the Router Advertisement messages sent by the SGW. The value zero means unspecified. (Default: 0s). (Default: 0) |
| **RAManagedFlag** | Indicates that addresses are available via DHCPv6. (Default: False). (Default: No) |
| **RAOtherConfigFlag** | Indicates that other configuration information is available via DHCPv6. (Default: False). (Default: No) |
| **RACurHopLimit** | The default value to be placed in the Cur Hop Limit field in the Router Advertisement messages sent by the SGW. The value zero means unspecified. (Default: 64). (Default: 64) |
| **RALinkMTU** | The value to be placed in MTU options sent. A value of zero indicates that no MTU options are sent. (Default: 0). (Default: 0) |
| **RAValidLifetime** | The value to be placed in the Valid Lifetime in the Prefix Information option. The value of 999999999 represents infinity. (Default: 2592000s). (Default: 2592000) |
| **RAPreferredLifetime** | The value to be placed in the Preferred Lifetime in the Prefix Information option. The value of 999999999 represents infinity. (Default: 604800s). (Default: 604800) |
| **RAOnLinkFlag** | Indicates that the advertised prefix can be used for on-link determination. (Default: True). (Default: Yes) |
| **RAAutonomousFlag** | Indicates that the advertised prefix can be used for stateless address configuration. (Default: True). (Default: Yes) |

*Note*

*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.10. AuthAgent

## Description

The Authentication Agent collect user login and logout events on a network domain controller.

## Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the agent. |
| **IPAddress** | The IP address of the agent. |
| **Port** | The listening port of the agent. (Default: 9999) |
| **PSK** | Selects the Pre-shared key to use with this agent. (Default: auth_agent_psk) |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **RoutingTable** | Specifies the routing table the clients host route should be added to. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

> ### *Note*
> *If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.11. AuthenticationSettings

**Description**

Settings related to Authentication and Accounting.

**Properties**

| | |
|---|---|
| **LogoutAccUsersAtShutdown** | Logout authenticated accounting users and send AccountingStop packets prior to shutdown. (Default: Yes) |
| **AllowAuthIfNoAccountingResponse** | Allow an authenticated user to still have access even if no response is received by the Accounting Server. (Default: Yes) |
| **VendorSpecificAttributeAccounting** | Enable sending Vendor-Specific attribute to the RADIUS server at Accounting-Request messages. (Default: No) |
| **VendorSpecificAttributeAuthentication** | Enable sending Vendor-Specific attribute to the RADIUS server at Access-Request messages. (Default: No) |
| **LogALGUser** | Log authenticated user together with URL in ALG log messages. (Default: Yes) |
| **LogConnUser** | Include authenticated user name in CONN logs. (Default: Yes) |
| **MaxRADIUSContexts** | Maximum number of RADIUS communication contexts. (Default: 1024) |
| **BruteForceMode** | Which settings should be used for brute force protection. (Default: Automatic) |
| **BruteForceNumAttempts** | Number of times a user is able to try a password before their account is locked down. (Default: 5) |
| **BruteForceLockdownTime** | How long, in seconds, a user is locked down after failing to login. (Default: 30) |

> **Note**
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.12. AzureVPN

### Description

This type lets you set up an IPsec tunnel to Microsoft Azure (IKEv2 tunnel with AES, SHA-2, DH group 2,14 and forward secrecy). Please note that the DH group 2 is considered insecure and shouldn't be used. Group 2 is however the default and only DH group set by default in Azure. It's recommended that you configure Azure to use the more secure DH group 14.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **LocalNetwork** | The network on "this side" of the IPsec tunnel. The IPsec tunnel will be established between this network and the remote network. |
| **RemoteNetwork** | The network connected to the remote gateway. The IPsec tunnel will be established between the local network and this network. |
| **PSK** | Selects the Pre-shared key to use with this IPsec Tunnel. |
| **RemoteEndpoint** | Specifies the IP address of the remote endpoint. This is the address the firewall will establish the IPsec tunnel to. It also dictates from where inbound IPsec tunnels are allowed. |
| **AutoInterfaceNetworkRoute** | Automatically add a route for this interface using the given remote network. (Default: Yes) |
| **SNMPIndex** | Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0) |
| **MemberOfRoutingTable** | All or Specific. (Default: All) |
| **RoutingTable** | Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

# 3.13. BlacklistWhiteHost

### Description

Hosts and networks added to this whitelist can never be blacklisted by IDP or Threshold Rules.

### Properties

**Addresses**                               Specifies the addresses that will be whitelisted.

**Service**                                 Specifies the service that will be whitelisted.

**Schedule**                                The schedule when the whitelist should be active. (Optional)

**Comments**                                Text describing the current object. (Optional)

> ### *Note*
> *If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.14. BotnetProtection

### Description

Protect both inbound and outbound traffic from undesired communication with command and control servers as well as infected zombie machines. Detected botnet peers are automatically blacklisted for efficient blocking. Specific hosts can be excluded from Botnet Protection using the Whitelist.

### Properties

| | |
|---|---|
| **EnableBotnetBlacklist** | Botnet Protection looks up source and destination IP addresses in the IP reputation database and adds malicious source and destinations to the Blacklist. (Default: No) |
| **ZDEnabled** | Enable ZoneDefense blocking. (Default: No) |
| **ZDNetwork** | Hosts within this range are blocked by ZoneDefense if a zombie machine is detected. |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

> ### Note
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.15. Certificate

**Description**

An X.509 certificate can be used to authenticate IKE peers or HTTPS servers.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the certificate. (Identifier) |
| **Type** | Local, Remote or Request. |
| **CertificateData** | Certificate data. |
| **PrivateKey** | Private key. |
| **CRLChecks** | Specifies whether to check CRLs (Certificate Revocation Lists) when validating certificates. (Default: Enforced) |
| **CRLDistPointList** | Specifies the CRL distribution points to use when validating the certificate itself and any issued certificates. Existing distribution points in the certificates will be overriden. (Optional) |
| **PKAType** | Encryption algorithm of the public key. (Default: Unknown) |
| **IsCA** | Is Certificate Authority. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

# 3.16. COMPortDevice

**Description**

A serial communication port, that is used for accessing the CLI.

**Properties**

| | |
|---|---|
| **Port** | Port. (Identifier) |
| **BitsPerSecond** | Bits per second. (Default: 9600) |
| **DataBits** | Data bits. (Default: 8) |
| **Parity** | Parity. (Default: None) |
| **StopBits** | Stop bits. (Default: 1) |
| **FlowControl** | Flow control. (Default: None) |
| **Comments** | Text describing the current object. (Optional) |

# 3.17. ConfigModePool

**Description**

An IKE Config Mode Pool will dynamically assign the IP address, DNS server, WINS server etc. to the VPN client connecting to this firewall.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the Config Mode Pool. (Identifier) |
| **IPPoolType** | Specifies whether a predefined IP Pool or a static set of IP addresses should be used as IP address source. |
| **IPPool** | Specifies the IP pool to use for assigning IP addresses to VPN clients. |
| **IPPoolAddress** | Specifies the set of IP addresses to use for assigning IP addresses to VPN clients. |
| **IPPoolNetmask** | Specifies the netmask to assign to VPN clients. (Optional) |
| **DNS** | Specifies the IP address of a DNS server that a VPN client should be able to connect to. (Optional) |
| **NBNSIP** | Specifies the IP address of a NBNS/WINS server that a VPN client should be able to connect to. (Optional) |
| **DHCP** | Specifies the IP address of a DHCP that that a VPN client should be able to connect to. (Optional) |
| **Subnets** | Specifies additional subnets behind this firewall. (Optional) |
| **IPv6Prefixes** | Specifies IPv6 prefixes. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

# 3.18. ConnTimeoutSettings

**Description**

Timeout settings for various protocols.

**Properties**

| | |
|---|---|
| **ConnLife_TCP_SYN** | Connection idle lifetime for TCP connections being formed. (Default: 60) |
| **ConnLife_TCP** | Connection idle lifetime for TCP. (Default: 262144) |
| **ConnLife_TCP_FIN** | Connection idle lifetime for TCP connections being closed. (Default: 80) |
| **ConnLife_UDP** | Connection idle lifetime for UDP. (Default: 130) |
| **AllowBothSidesToKeepConnAlive_UDP** | Allow both sides to keep a UDP connection alive. (Default: No) |
| **ConnLife_Ping** | Connection timeout for Ping. (Default: 8) |
| **ConnLife_Other** | Idle lifetime for other protocols. (Default: 130) |
| **ConnLife_IGMP** | Connection idle lifetime for IGMP. (Default: 12) |

> **Note**
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.19. CRLDistPointList

**Description**

A CRL distribution point list specifies one or more locations from where a certificate revocation list (CRL) can be obtained. It can be used to add distribution points to a certificate that does not provide any, or to override existing ones. Listed distribution points will be tried in order of occurance.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the CRL distribution point list. (Identifier) |
| **Comments** | Text describing the current object. (Optional) |

# 3.19.1. CRLDistPoint

**Description**

A CRL distribution point (CDP) specifies a location from where a certificate revocation list (CRL) can be obtained.

**Properties**

| | |
|---|---|
| **URL** | Specifies the URL for the CRL distribution point. For example http://www.example.com/ca.crl. |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> *If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.20. DateTime

**Description**

Set the date, time and time zone information for this system.

**Properties**

| | |
|---|---|
| **Location** | Specifies local time zone. (Default: DLinkHQ) |
| **DSTEnabled** | Enable daylight saving time. (Default: Yes) |
| **TimeSynchronization** | Enable time synchronization. (Default: D-Link) |
| **TimeSyncServerType** | Type of server for time synchronization, UDPTime or SNTP (Simple Network Time Protocol). (Default: SNTP) |
| **TimeSyncServer1** | DNS hostname or IP Address of Timeserver 1. |
| **TimeSyncServer2** | DNS hostname or IP Address of Timeserver 2. (Optional) |
| **TimeSyncServer3** | DNS hostname or IP Address of Timeserver 3. (Optional) |
| **TimeSyncInterval** | Seconds between each resynchronization. (Default: 86400) |
| **TimeSyncMaxAdjust** | Maximum time drift in seconds that a server is allowed to adjust. (Default: 600) |
| **TimeSyncGroupIntervalSize** | Interval according to which server responses will be grouped. (Default: 10) |
| **Comments** | Text describing the current object. (Optional) |

> *Note*
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.21. DefaultInterface

**Description**

A special interface used to represent internal mechanisms in the system as well as an abstract "any" interface.

**Properties**

**Name**

Specifies a symbolic name for the interface. (Identifier)

**SNMPIndex**

Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0)

**Comments**

Text describing the current object. (Optional)

# 3.22. Device

**Description**

Global parameters for this device.

**Properties**

| | |
|---|---|
| **Name** | Name of the device. (Default: Device) |
| **LocalCfgVersion** | Local version number of the configuration. (Default: 1) |
| **NextSNMPIfIndex** | SNMP interface index assigned to the next interface created within the system. (Default: 1) |
| **ConfigUser** | Name of the user who committed the current configuration. (Default: BaseConfiguration) |
| **ConfigSession** | Session type used when the current configuration was committed. (Default: BaseConfiguration) |
| **ConfigIP** | IP address of the user who committed the current configuration. (Optional) |
| **ConfigDate** | Date when the current configuration was committed. (Optional) |
| **OEMID** | OEM identification string. (Default: 0) |
| **HWModel** | System hardware model. (Default: SOFTWARE) |
| **Comments** | Text describing the current object. (Optional) |

> ***Note***
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.23. DHCPRelay

**Description**

Use a DHCP Relay to dynamically alter the routing table according to relayed DHCP leases.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the relay rule. (Identifier) |
| **Action** | Ignore, Relay or BootpFwd. (Default: Ignore) |
| **SourceInterface** | The source interface of the DHCP packet. |
| **TargetDHCPServer** | Specifies the IP of the server to send the relayed DHCP packets to. |
| **TargetDHCPServer2** | Optional secondary server. (Optional) |
| **TargetDHCPServer3** | Optional tertiary server. (Optional) |
| **IPOfferFilter** | Specifies the span of IP addresses that are allowed to be relayed from the DHCP server. (Default: 1) |
| **AddRoute** | Enable dynamic adding of routes as leases are added and removed. (Default: No) |
| **AddRouteLocalIP** | The IP Address specified here will automatically be published on the interfaces where a route is added. (Optional) |
| **AddRouteGatewayIP** | The IP used as gateway to reach hosts on this route. (Optional) |
| **RoutingTable** | Specifies the routing table the clients host route should be added to. (Default: main) |
| **MaxRelaysPerInterface** | Specifies how many relays are allowed per interface, that means, how many DHCP clients are allowed to be relayed through each interface. (Optional) |
| **AgentIP** | Define what IP the relay should use as gateway IP when passing the requests to the DHCP server. (Default: Recv) |
| **AllowNULLOffers** | Accept server responses offering IP address "0.0.0.0" (no IP address offered). (Default: No) |
| **ProxyARPAllInterfaces** | Always select all interfaces, including new ones, for publishing routes needed for the relay via Proxy ARP. (Default: No) |
| **ProxyARPInterfaces** | Specifies the interface/interfaces on which the firewall should publish routes needed for the relay via Proxy ARP. (Optional) |
| **LogEnabled** | Enable logging. (Default: Yes) |

**LogSeverity**                          Specifies with what severity log events will be sent
                                         to the specified log receivers. (Default: Default)

**Comments**                             Text describing the current object. (Optional)

# 3.24. DHCPRelaySettings

**Description**

Advanced DHCP relay settings.

**Properties**

| | |
|---|---|
| **MaxTransactions** | Maximum number of concurrent BOOTP/DHCP transactions. (Default: 32) |
| **TransactionTimeout** | Timeout for each transaction (in seconds). (Default: 10) |
| **MaxPPMPerIface** | Maximum packets per minute that are relayed from clients to the server, per interface. (Default: 500) |
| **MaxHops** | Requests/responses that have traversed more than this many relays will not be relayed. (Default: 5) |
| **MaxLeaseTime** | Maximum lease time (seconds) allowed from the DHCP server (too high times will be lowered silently). (Default: 10000) |
| **MaxConcurrentRelays** | Maximum number of concurrently active DHCP relays. (Default: 256) |
| **AutoSaveRelayPolicy** | Policy for saving the relay list to disk. (Default: ReconfShut) |
| **AutoSaveRelayInterval** | Seconds between auto saving the relay list to disk. (Default: 86400) |

*Note*
*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.25. DHCPServer

**Description**

A DHCP Server determines a set of IP addresses and host configuration parameters to hand out to DHCP clients attached to a given interface.

**Properties**

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the DHCP Server rule. (Identifier) |
| **Interface** | The source interface to listen for DHCP requests on. This can be a single interface or a group of interfaces. |
| **RelayerFilter** | A range, group or network that will allow specific DHCP Relayers access to the DHCP Server. (Default: 0/0) |
| **IPAddressPool** | A range, group or network that the DHCP Server will use as IP address pool to give out DHCP leases from. |
| **Netmask** | Netmask sent to the DHCP Client. (Default: 255) |
| **DefaultGateway** | Specifies what IP should be sent to the client for use as default gateway. If unspecified or if 0.0.0.0 is specified, the IP given to the client will be sent as gateway. (Optional) |
| **Domain** | Domain name used for DNS resolution. (Optional) |
| **LeaseTime** | The time, in seconds, that a DHCP lease should be provided to a host after this the client have to renew the lease. (Default: 86400) |
| **DNS1** | IP of the primary DNS server. (Optional) |
| **DNS2** | IP of the secondary DNS server. (Optional) |
| **NBNS1** | IP of the primary Windows Internet Name Service (WINS) server that is used in Microsoft environments which uses the NetBIOS Name Servers (NBNS) to assign IP addresses to NetBIOS names. (Optional) |
| **NBNS2** | IP of the primary Windows Internet Name Service (WINS) server that is used in Microsoft environments which uses the NetBIOS Name Servers (NBNS) to assign IP addresses to NetBIOS names. (Optional) |
| **LeasesRequireAuth** | Enable distribution of leases only after clients have been authenticated. (Default: No) |
| **NextServer** | IP address of next server in the boot process. |

(Optional)

**LogEnabled**                          Enable logging. (Default: Yes)

**LogSeverity**                         Specifies with what severity log events will be sent
                                        to the specified log receivers. (Default: Default)

**Comments**                            Text describing the current object. (Optional)

## 3.25.1. DHCPServerPoolStaticHost

**Description**

Static DHCP Server host entry

**Properties**

**Host**                                IP Address of the host.

**StaticHostType**                      Identifier for host. (Default: MACAddress)

**MACAddress**                          The hardware address of the host.

**ClientIdentType**                     Type of client identifier specified. (Default: Ascii)

**ClientIdent**                         The client identifier for the host.

**Comments**                            Text describing the current object. (Optional)

> **Note**
> If no `Index` is specified when creating an instance of this type, the object will be placed
> last in the list and the `Index` will be equal to the length of the list.

## 3.25.2. DHCPServerCustomOption

**Description**

Extend the DHCP Server functionality by adding custom options that will be handed out to the
DHCP clients.

**Properties**

**Code**                                The DHCP option code.

**Type**                                What type the option is, i.e. STRING, IP4 and so on.
                                        (Default: UINT8)

**Param**                               The parameter sent with the code, this can be one
                                        parameter or a comma separated list.

**Comments**                            Text describing the current object. (Optional)

**Note**

*If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.26. DHCPServerSettings

**Description**

Advanced DHCP server settings.

**Properties**

**AutoSaveLeasePolicy**            Policy for saving the lease database to disk. (Default: ReconfShut)

**AutoSaveLeaseInterval**          Seconds between auto saving the lease database to disk. (Default: 86400)

***Note***
*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.27. DHCPv6Server

**Description**

A DHCPv6 Server determines a set of IPv6 addresses and host configuration parameters to hand out to DHCPv6 clients attached to a given interface.

**Properties**

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the DHCPv6 Server rule. (Identifier) |
| **Interface** | The source interface to listen for DHCPv6 requests on. This can be a single interface or a group of interfaces. |
| **IPv6AddressPool** | A range, group or network that the DHCP Server will use as IPv6 address pool to give out DHCPv6 leases from. |
| **Domain** | Domain name used for DNS resolution. (Optional) |
| **ValidLeaseTime** | The length of time in seconds that an address remains valid for sending and receiving packets. When expired, the host is not allowed to use the provided address any more and should acquire a new one. (Default: 86400) |
| **PreferredLeaseTime** | The length of time in seconds that an address should be preferred to be used in new communications. When expired, unless renewed, the address becomes deprecated and should no longer be used as a source address in new communications. (Default: 66400) |
| **DNS1** | IPv6 of the primary DNS server. (Optional) |
| **DNS2** | IPv6 of the secondary DNS server. (Optional) |
| **SendUnicastOption** | Enable sending of Unicast option to DHCPv6 client. (Default: No) |
| **ClearUniversalLocalBit** | Clear the universal/local bit in the IPv6 address pool in case of /64 networks. (Default: No) |
| **RapidCommit** | Enable respond with committed address assignments and other resources on Solicit request. (Default: No) |
| **PreferenceConfigured** | Enable Preference option sending in Advertise message. (Default: No) |
| **PreferenceValue** | Preference Option value. (Default: 0) |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent |

to the specified log receivers. (Default: Default)

**Comments**                                   Text describing the current object. (Optional)

# 3.27.1. DHCPv6ServerPoolStaticHost

**Description**

Static DHCPv6 Server host entry

**Properties**

**Host**                                       IPv6 Address of the host.

**MACAddress**                                 The hardware address of the host.

**Comments**                                   Text describing the current object. (Optional)

### *Note*
*If no* Index *is specified when creating an instance of this type, the object will be placed last in the list and the* Index *will be equal to the length of the list.*

# 3.28. DHCPv6ServerSettings

**Description**

Advanced DHCPv6 server settings.

**Properties**

**AutoSaveLeasePolicy**                Policy for saving the lease database to disk. (Default: ReconfShut)

**AutoSaveLeaseInterval**              Seconds between auto saving the lease database to disk. (Default: 86400)

> ***Note***
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.29. DiagnosticsSettings

**Description**

Control how anonymous usage statistics are automatically shared with D-Link to improve the quality of the product and the services. Sensitive information e.g. VPN keys or certificates are not shared. All communication is encrypted and no information is shared with 3rd parties.

**Properties**

**EnableDiagnostics**

Allow anonymous diagnostics reports to be sent to D-Link. (Default: Yes)

**IncludeUsageStatistics**

Include usage statistics e.g. CPU load, connection count and memory usage to manufacturer. The information will improve the quality of future products and releases. (Default: Yes)

**SendExceptionReports**

Send exception reports automatically to the manufacturer. The reports will help us to identify critical issues and to provide a correction quicker. (Default: Yes)

**Note**
*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.30. DNS

**Description**

Configure the DNS (Domain Name System) client settings.

**Properties**

| | |
|---|---|
| **DNSServer1** | IP of the primary IPv4 DNS Server. (Optional) |
| **DNSServer2** | IP of the secondary IPv4 DNS Server. (Optional) |
| **DNSServer3** | IP of the tertiary IPv4 DNS Server. (Optional) |
| **IP6DNSServer1** | IP of the primary IPv6 DNS Server. (Optional) |
| **IP6DNSServer2** | IP of the secondary IPv6 DNS Server. (Optional) |
| **IP6DNSServer3** | IP of the tertiary IPv6 DNS Server. (Optional) |
| **MinTTL** | Overrides lower TTLs received from the DNS server when used in DNS cache. (Default: 1) |
| **MinCacheTime** | Determines the minimum amount of time an IP address remains in the cache. (Default: 86400) |
| **Comments** | Text describing the current object. (Optional) |

> ***Note***
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

*160*

# 3.31. DNSProfile

**Description**

A DNS Profile can be used by one or many IP Policies which has its service object configured with DNS as protocol.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the Profile. (Identifier) |
| **MaxUDPQueryLength** | Maximum payload size in DNS queries over UDP. (Default: 4096) |
| **MaxUDPResponseLength** | Maximum payload size in DNS responses over UDP. (Default: 4096) |
| **MaxTCPQueryLength** | Maximum message size in DNS queries over TCP. (Default: 4096) |
| **MaxTCPResponseLength** | Maximum message size in DNS responses over TCP. (Default: 4096) |
| **LogDNSLookups** | Log resolved addresses. (Default: Yes) |
| **PopulateDNSCache** | Populate resovled addresses to DNS-cache. (Default: Yes) |
| **RecursionDesiredFlag** | Policy for handling the Recursion Desired flag in DNS messages. (Default: Allow) |
| **MaxQuestionEntries** | Maximum number of question entries. (Default: 1) |
| **AllowAllClasses** | All DNS Record classes are allowed in DNS queries and responses. (Default: No) |
| **AllowedClasses** | List of allowed DNS Record classes in DNS queries and reponses. (Default: IN) |
| **AllowAllTypes** | All DNS Record types are allowed in DNS queries and responses. (Default: Yes) |
| **AllowedTypes** | List of allowed DNS Record types in DNS queries and responses. |
| **ScrambleQueryID** | Mitigation against cache poisoning. Scrambles message IDs in queries sent over UDP, and de-scrambles them before delivering the reply. (Default: Yes) |
| **Comments** | Text describing the current object. (Optional) |

# 3.32. DoSProtection

### Description

DoS Protection protects servers behind the firewall from Denial of Service attacks. Detected DoS sources are automatically blacklisted for efficient blocking. Specific hosts can be excluded from DoS protection using the Whitelist. DoS Protection can also block traffic from configured geographical regions.

### Properties

| | |
|---|---|
| **EnableRegionBlacklist** | Enables filtering by regions. (Default: No) |
| **Regions** | Specifies matching regions for this filter. (Optional) |
| **EnableDoSBlacklist** | DoS Protection looks up source IP addresses in the IP reputation database and adds malicious sources to the Blacklist. (Default: No) |
| **Interfaces** | Interfaces to protect from attacks. Normally the interfaces towards the Internet. (Optional) |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.33. DynamicRoutingRule

**Description**

A Dynamic Routing Policy rule creates a filter to catch statically configured or OSPF learned routes. The matched routes can be controlled by the action rules to be either exported to OSPF processes or to be added to one or more routing tables.

**Properties**

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the rule. (Optional) |
| **From** | OSPF or Routing table. (Default: OSPF) |
| **OSPFProcess** | Specifies from which OSPF process the route should be imported from into either a routing table or another OSPF process. |
| **RoutingTable** | Specifies from which routing table a route should be imported into the OSPF AS or copied into another routing table. |
| **DestinationInterface** | The interface that the policy has to match. (Optional) |
| **DestinationNetworkExactly** | Specifies if the route needs to match a specific network exactly. (Optional) |
| **DestinationNetworkIn** | Specifies if the route just needs to be within a specific network. (Optional) |
| **NextHop** | The next hop (router) on the route that this policy has to match. (Optional) |
| **MetricRange** | Specifies an interval that the metric of the routes needs to be within. (Optional) |
| **RouterID** | Specifies if the policy should filter on router ID. (Optional) |
| **OSPFRouteType** | Specifies if the policy should filter on OSPF router type. (Optional) |
| **OSPFTagRange** | Specifies an interval that the tag of the routers need to be within. (Optional) |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

> ***Note***
> *If no* `Index` *is specified when creating an instance of this type, the object will be placed*

*last in the list and the `Index` will be equal to the length of the list.*

# 3.33.1. DynamicRoutingRuleExportOSPF

### Description

An OSPF action is used to manipulate and export new or changed routes to an OSPF Router Process.

### Properties

| | |
|---|---|
| **ExportToProcess** | Specifies to which OSPF Process the route change should be exported. |
| **SetTag** | Specifies a tag for this route. This tag can be used in other routers for filtering. (Optional) |
| **SetRouteType** | The external route type. (Optional) |
| **OffsetMetric** | Increases the metric of the imported route by this value. (Optional) |
| **LimitMetricRange** | Limits the metrics for these routes to a minimum and maximum value, if a route has a higher or lower value then specified it will be set to the specified value. (Optional) |
| **SetForward** | IP to route over. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

### Note
*If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.33.2. DynamicRoutingRuleAddRoute

### Description

A routing action is used to manipulate and insert new or changed routes to one or more local routing tables.

### Properties

| | |
|---|---|
| **Destination** | Specifies to which routing table the route changes to the OSPF Process should be exported. |
| **OverrideStatic** | Allow override of static routes. (Default: No) |
| **OverwriteDefault** | Allow overwrite of default route. (Default: No) |

**OffsetMetric**                     Increases the metric by this value. (Optional)

**OffsetMetricType2**                Increases the for Type2 routers metric by this value.
                                     (Optional)

**LimitMetricRange**                 Limits the metrics for these routes to a minimum
                                     and maximum value, if a route has a higher or
                                     lower value then specified it will be set to the
                                     specified value. (Optional)

**ProxyARPAllInterfaces**            Always select all interfaces, including new ones, for
                                     publishing routes via Proxy ARP. (Default: No)

**ProxyARPInterfaces**               Specifies the interfaces on which the firewall
                                     should publish routes via Proxy ARP. (Optional)

**Comments**                         Text describing the current object. (Optional)

> ***Note***
> *If no `Index` is specified when creating an instance of this type, the object will be placed
> last in the list and the `Index` will be equal to the length of the list.*

# 3.34. DynDnsClientCjbNet

**Description**

Configure the parameters used to connect to the Cjb.net Dynamic DNS service.

**Properties**

| | |
|---|---|
| **Username** | Username. |
| **Password** | The password for the specified username. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

***Note***
*If no Index is specified when creating an instance of this type, the object will be placed last in the list and the Index will be equal to the length of the list.*

# 3.35. DynDnsClientDLink

**Description**

Configure the parameters used to connect to the D-Link DynDNS service.

**Properties**

| | |
|---|---|
| **DNSName** | The DNS name excluding the .dlinkddns.com suffix. |
| **Username** | Username. |
| **Password** | The password for the specified username. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

***Note***
*If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.36. DynDnsClientDLinkChina

**Description**

Configure the parameters used to connect to the D-Link DynDNS service (China only).

**Properties**

| | |
|---|---|
| **DNSName** | The DNS name excluding the .dlinkddns.com suffix. |
| **Username** | Username. |
| **Password** | The password for the specified username. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

> ***Note***
> *If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.37. DynDnsClientDyndnsOrg

**Description**

Configure the parameters used to connect to the dyn.com Dynamic DNS service.

**Properties**

| | |
|---|---|
| **DNSName** | The DNS name excluding the .dyndns.org suffix. |
| **Username** | Username. |
| **Password** | The password for the specified username. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

***Note***
*If no Index is specified when creating an instance of this type, the object will be placed last in the list and the Index will be equal to the length of the list.*

# 3.38. DynDnsClientDynsCx

**Description**

Configure the parameters used to connect to the dyns.cx Dynamic DNS service.

**Properties**

| | |
|---|---|
| **DNSName** | The DNS name excluding the .dyns.cx suffix. |
| **Username** | Username. |
| **Password** | The password for the specified username. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

***Note***
*If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.39. DynDnsClientPeanutHull

**Description**

Configure the parameters used to connect to the Peanut Hull Dynamic DNS service.

**Properties**

| | |
|---|---|
| **DNSNames** | Specifies the DNS names separated by ";". |
| **Username** | Username. |
| **Password** | The password for the specified username. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> If no *Index* is specified when creating an instance of this type, the object will be placed last in the list and the *Index* will be equal to the length of the list.

# 3.40. EmailControlProfile

**Description**

An E-mail Control Profile can be used by one or many IP Policies which has its service object configured with a protocol that supports e-mail scanning (IMAP, POP3, SMTP).

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the Profile. (Identifier) |
| **AntiSpam** | Anti-Spam protects against unsolicited bulk email. (Default: No) |
| **TagThreshold** | An email is tagged if the total score of all anti-spam mechanisms exceeds this threshold. (Default: 10) |
| **RejectThreshold** | An email is rejected if the total score of all anti-spam mechanisms exceeds this threshold. Applies to SMTP only. (Default: 20) |
| **TagSubject** | Prefix email subject with a custom text string if the Tag Threshold is exceeded. (Default: Yes) |
| **SubjectTag** | Custom text string to tag subject with. (Default: "*** SPAM *** ") |
| **TagHeader** | Suffix email header with informative X-Spam header fields. (Default: Yes) |
| **DomainVerification** | Use DNS to verify reply-to domains in emails. If a domain appears to be forged, the configured score value is added to the total score for that email. (Default: Yes) |
| **DomainVerificationScore** | Specify a score value for Domain Verification. (Default: 10) |
| **LinkProtection** | Neutralize undesirable web links in emails. If one or more undesirable links are found, the configured score value is added to the total score for that email. (Default: Yes) |
| **LinkProtectionScore** | Specify a score value for Link Protection. (Default: 10) |
| **LinkProtectionCategories** | Specify undesirable link categories. (Optional; Default: ADULT_CONTENT,BOTNETS,CHILD_ABUSE_MATERIAL,CRIME_TERROR |
| **LinkProtectionNonManagedAction** | Action to take for content that has not been classified. (Default: Allow) |
| **DNSBL** | A DNS Blacklist is a 3rd party database of IP addresses that have been used to send spam. As the name implies, the DNS protocol is used to perform queries. Up to 10 DNS Blacklists may be configured. (Default: No) |

| | |
|---|---|
| **DNSBL1** | IP address blacklisting using an external database. If the sender's IP address is blacklisted, the configured score value is added to the total score for that email. (Default: No) |
| **DNSBL2** | IP address blacklisting using an external database. If the sender's IP address is blacklisted, the configured score value is added to the total score for that email. (Default: No) |
| **DNSBL3** | IP address blacklisting using an external database. If the sender's IP address is blacklisted, the configured score value is added to the total score for that email. (Default: No) |
| **DNSBL4** | IP address blacklisting using an external database. If the sender's IP address is blacklisted, the configured score value is added to the total score for that email. (Default: No) |
| **DNSBL5** | IP address blacklisting using an external database. If the sender's IP address is blacklisted, the configured score value is added to the total score for that email. (Default: No) |
| **DNSBL6** | IP address blacklisting using an external database. If the sender's IP address is blacklisted, the configured score value is added to the total score for that email. (Default: No) |
| **DNSBL7** | IP address blacklisting using an external database. If the sender's IP address is blacklisted, the configured score value is added to the total score for that email. (Default: No) |
| **DNSBL8** | IP address blacklisting using an external database. If the sender's IP address is blacklisted, the configured score value is added to the total score for that email. (Default: No) |
| **DNSBL9** | IP address blacklisting using an external database. If the sender's IP address is blacklisted, the configured score value is added to the total score for that email. (Default: No) |
| **DNSBL10** | IP address blacklisting using an external database. If the sender's IP address is blacklisted, the configured score value is added to the total score for that email. (Default: No) |
| **DNSBL1Name** | Specify the DNS name of a DNS Blacklist. |
| **DNSBL2Name** | Specify the DNS name of a DNS Blacklist. |
| **DNSBL3Name** | Specify the DNS name of a DNS Blacklist. |
| **DNSBL4Name** | Specify the DNS name of a DNS Blacklist. |
| **DNSBL5Name** | Specify the DNS name of a DNS Blacklist. |
| **DNSBL6Name** | Specify the DNS name of a DNS Blacklist. |

| | |
|---|---|
| **DNSBL7Name** | Specify the DNS name of a DNS Blacklist. |
| **DNSBL8Name** | Specify the DNS name of a DNS Blacklist. |
| **DNSBL9Name** | Specify the DNS name of a DNS Blacklist. |
| **DNSBL10Name** | Specify the DNS name of a DNS Blacklist. |
| **DNSBL1Score** | Specify a score value for DNS Blacklist 1. (Default: 10) |
| **DNSBL2Score** | Specify a score value for DNS Blacklist 2. (Default: 10) |
| **DNSBL3Score** | Specify a score value for DNS Blacklist 3. (Default: 10) |
| **DNSBL4Score** | Specify a score value for DNS Blacklist 4. (Default: 10) |
| **DNSBL5Score** | Specify a score value for DNS Blacklist 5. (Default: 10) |
| **DNSBL6Score** | Specify a score value for DNS Blacklist 6. (Default: 10) |
| **DNSBL7Score** | Specify a score value for DNS Blacklist 7. (Default: 10) |
| **DNSBL8Score** | Specify a score value for DNS Blacklist 8. (Default: 10) |
| **DNSBL9Score** | Specify a score value for DNS Blacklist 9. (Default: 10) |
| **DNSBL10Score** | Specify a score value for DNS Blacklist 10. (Default: 10) |
| **BlacklistTag** | For IMAP and POP3, custom text string to tag subject of blacklisted emails. For SMTP this has no effect; blacklisted messages are rejected instead. (Default: "*** BLACK LISTED *** ") |
| **IMAP_HideUser** | Prevent server from revealing that a user name does not exist. (Default: No) |
| **IMAP_BlockPlainAuth** | Block plain text authentication. (Default: No) |
| **IMAP_AllowSTARTTLS** | Allow clients to use the STARTTLS command. Note that this allows encrypted transactions to take place, which circumvents any enabled security mechanisms. (Default: No) |
| **POP3_HideUser** | Prevent server from revealing that a user name does not exist. (Default: No) |
| **POP3_AllowUnknownCommands** | Allow unknown commands. (Default: No) |
| **POP3_BlockUserPass** | Block clients from sending USER and PASS command. (Default: No) |
| **POP3_AllowSTARTTLS** | Allow clients to use the STARTTLS command. Note |

|  | that this allows encrypted transactions to take place, which circumvents any enabled security mechanisms. (Default: No) |
| --- | --- |
| **SMTP_MaxEmailPerMinute** | Specifies the maximum amount of emails per minute from the same host. (Optional) |
| **SMTP_MaxEmailSize** | Specifies the maximum allowed email size in kB. (Optional) |
| **SMTP_AllowSTARTTLS** | Allow clients to use the STARTTLS command. Note that this allows encrypted transactions to take place, which circumvents any enabled security mechanisms. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

# 3.40.1. EmailFilter

**Description**

Add an email filter to whitelist or blacklist an email source and/or destination combination. A whitelisted message will bypass all other anti-spam mechanisms. A blacklisted message is treated as spam.

**Properties**

| **Action** | A blacklisted message is treated as spam. A whitelisted message will bypass all other anti-spam mechanisms. (Default: Blacklist) |
| --- | --- |
| **SrcType** | Source can either be an IP address or an email address from which the email was sent. (Default: Email) |
| **SrcEmail** | Specify sender email address. Wildcards can be used. Supported wildcards are *(multi character match) and ?(single character match). |
| **SrcIP** | Specify the IP address of the sender. (Optional) |
| **DestEmail** | Specify email address of the receiver. Wildcards can be used. Supported wildcards are *(multi character match) and ?(single character match). (Default: *) |
| **Comments** | Text describing the current object. (Optional) |

***Note***

*If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.41. Ethernet

**Description**

An Ethernet interface represents a logical endpoint for Ethernet traffic.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **EthernetDevice** | Hardware settings for the Ethernet interface. |
| **VLanQoSInherit** | Set whether VLANs using the interface should inherit the IP QoS bits. (Default: No) |
| **ReceiveMulticastTraffic** | Sets the multicast receive mode of the interface. (Default: Auto) |
| **LACPPortPriority** | Port priority value to be sent in LACP messages. (Default: 1) |
| **IP** | The IP address of the interface. |
| **Network** | The network of the interface. |
| **DefaultGateway** | The default gateway of the interface. (Optional) |
| **Broadcast** | The broadcast address of the connected network. (Optional) |
| **EnableIPv6** | Enable processing of IPv6 traffic on this interface. (Default: No) |
| **IPv6IP** | The IP address of the interface. |
| **IPv6Network** | The network of the interface. |
| **IPv6DefaultGateway** | The default gateway of the interface. (Optional) |
| **RouterDiscovery** | Uses Router information (ND RA) from local network to auto-configure Network and Default Gateway addresses. (Default: No) |
| **AutoIPv6IP** | Automatically configures IP Address using Network Address and EUI-64. (Default: No) |
| **DHCPv6Enabled** | Enable DHCPv6 client on this interface. (Default: No) |
| **PrivateIP** | The private IP address of this high availability node. (Optional) |
| **PrivateIP6** | The private IP6 address of this high availability node. (Default: localhost6) |
| **NOCHB** | This will disable sending Cluster Heartbeats from this interface (used by HA to detect if a node is online and working). (Optional) |

**MTU**  Specifies the size (in bytes) of the largest packet that can be passed onward. Must be 1294 or larger when IPv6 is enabled. (Default: 1500)

**Metric**  Specifies the metric for the auto-created route. (Default: 100)

**DHCPEnabled**  Enable DHCP client on this interface. (Default: No)

**DHCPHostName**  Optional DHCP Host Name. Leave blank to use default name. (Optional)

**AutoSwitchRoute**  Allows traffic to be forwarded transparently across all interfaces with Transparent Mode enabled that belong to the same routing table. (Default: No)

**DHCPPassthrough**  Allow DHCP to pass through transparently. (Default: No)

**NonIPPassthrough**  Allow non-IP protocols to pass through transparently. (Default: No)

**BroadcastFwd**  By default, this traffic is dropped. (Default: No)

**AutoInterfaceNetworkRoute**  Automatically add a route for this interface using the given network. (Default: Yes)

**AutoDefaultGatewayRoute**  Automatically add a default route for this interface using the given default gateway. (Default: Yes)

**DHCPDNS1**  IP of the primary DNS server. (Optional)

**DHCPDNS2**  IP of the secondary DNS server. (Optional)

**DHCPv6DNS1**  IP of the primary IPv6 DNS server. (Optional)

**DHCPv6DNS2**  IP of the secondary IPv6 DNS server. (Optional)

**EnableRouterAdvertisement**  Enable Router Advertisement for this interface. (Default: No)

**SNMPIndex**  Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0)

**MemberOfRoutingTable**  All or Specific. (Default: All)

**RoutingTable**  Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main)

**Comments**  Text describing the current object. (Optional)

*177*

# 3.42. EthernetDevice

**Description**

Hardware settings for an Ethernet interface.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the device. (Identifier) |
| **EthernetDriver** | The Ethernet PCI driver that should be used by the interface. |
| **PCIBus** | PCI bus number where the Ethernet adapter is installed. |
| **PCISlot** | PCI slot number used by the Ethernet adapter. |
| **PCIPort** | Some Ethernet adapters have multiple ports that share the same bus and slot number. This parameter specifies what port to be used. |
| **Media** | Specifies if the link speed should be auto-negotiated or locked to a static speed. (Default: Auto) |
| **Duplex** | Specifies if the duplex should be auto-negotiated or locked to full or half duplex. (Default: Auto) |
| **MACAddress** | The hardware address for the interface. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

# 3.43. EthernetSettings

**Description**

Settings for Ethernet interface.

**Properties**

**DHCP_MinimumLeaseTime**
Minimum lease time (seconds) accepted from the DHCP server. (Default: 60)

**DHCP_ValidateBcast**
Require that the assigned broadcast address is the highest address in the assigned network. (Default: Yes)

**DHCP_AllowGlobalBcast**
Allow DHCP server to assign 255.255.255.255 as broadcast (Non-standard). (Default: No)

**DHCP_UseLinkLocalIP**
Use a 169.254.*.* IP while waiting for a lease (instead of 0.0.0.0). (Default: No)

**DHCP_DisableArpOnOffer**
Disable arp resolve on offers (normally used to verify that an IP is not occupied). (Default: No)

**Ringsize_e1000_rx**
Size of e1000 receive ring (per interface). (Default: 128)

**Ringsize_e1000_tx**
Size of e1000 send ring (per interface). (Default: 256)

**Ringsize_r8169_rx**
Size of r8169 receive ring (per interface). (Default: 256)

**Ringsize_r8169_tx**
Size of r8169 send ring (per interface). (Default: 256)

**IfaceMon_e1000**
Enable interface monitor for e1000 interfaces. (Default: Yes)

**IfaceMon_BelowCPULoad**
Temporarily disable interface monitor if CPU load goes above this percentage. (Default: 80)

**IfaceMon_BelowIfaceLoad**
Temporarily disable interface monitor on an interface if network load on the interface goes above this percentage. (Default: 70)

**IfaceMon_MinInterval**
Minimum interval between two resets of the same interface. (Default: 30)

**IfaceMon_RxErrorPerc**
At what percentage of errors to received packets to declare a problem. (Default: 20)

**IfaceMon_TxErrorPerc**
At what percentage of errors to sent packets to declare a problem. (Default: 7)

**IfaceMon_ErrorTime**
How long a problem must persist before an interface is reset. (Default: 10)

**Note**
*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.44. EventReceiverSNMP2c

### Description

A SNMP2c event receiver is used to receive SNMP events from the system.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the log receiver. (Identifier) |
| **Community** | A shared string between the firewall and the server, identifying them as part of the same group. (Default: public) |
| **RepeatCount** | Controls how many times an event is sent before being ignored. A value of 0 means events will never be ignored. (Default: 0) |
| **SNMP2cIfTraps** | This enables generation of SNMPv2c traps for interface up/down events. (Default: No) |
| **IPAddress** | Destination IP address. |
| **Port** | Destination port. (Default: 162) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Optional; Default: Emergency,Alert,Critical,Error,Warning,Notice,Info) |
| **RoutingTable** | Specifies the routing table the clients host route should be added to. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

# 3.44.1. LogReceiverMessageException

### Description

A log message exception is used to override the severity filter in the log receiver.

**Properties**

| | |
|---|---|
| **LogCategory** | The Category of the log message. |
| **LogID** | The ID number of the log message, a empty value selects all messages of this category. (Optional) |
| **LogType** | EXCLUDE or INCLUDE. (Default: EXCLUDE) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

**Note**

*If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.45. EventReceiverSNMPv3

**Description**

A SNMPv3 event receiver is used to receive SNMP events from the system.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the log receiver. (Identifier) |
| **Snmp3SecurityLevel** | Enabled SNMPv3 security level. (Default: noAuthNoPriv) |
| **Username** | The user on whose behalf the message is being exchanged. |
| **Password** | The password used for both authentication and encryption. |
| **RepeatCount** | Controls how many times an event is sent before being ignored. A value of 0 means events will never be ignored. (Default: 0) |
| **SNMP3IfTraps** | This enables generation of SNMPv3 traps for interface up/down events. (Default: No) |
| **IPAddress** | Destination IP address. |
| **Port** | Destination port. (Default: 162) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Optional; Default: Emergency,Alert,Critical,Error,Warning,Notice,Info) |
| **RoutingTable** | Specifies the routing table the clients host route should be added to. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

## 3.45.1. LogReceiverMessageException

The definitions here are the same as in Section 3.44.1, "LogReceiverMessageException" .

# 3.46. FileControlPolicy

## Description

A File Control Profile can be used by one or many IP Policies which has its service object configured with a protocol that supports file control scanning (HTTP, FTP, POP3, SMTP, IMAP).

## Properties

**Name**
Specifies a symbolic name for the Profile. (Identifier)

**FileListType**
Specifies if the file list contains files to allow or deny. (Default: Block)

**File**
List of file types to allow or deny. (Optional)

**VerifyContentMimetype**
Verify that file extentions correspond to the MIME type. (Default: No)

**Comments**
Text describing the current object. (Optional)

# 3.47. FragSettings

### Description

Settings related to fragmented packets.

### Properties

| | |
|---|---|
| **PseudoReass_MaxConcurrent** | Maximum number of concurrent fragment reassemblies. Set to 0 to drop all fragments. (Default: 1024) |
| **IllegalFrags** | Illegaly constructed fragments; partial overlaps, bad sizes, etc. (Default: DropLog) |
| **DuplicateFragData** | On receipt of duplicate fragments, verify matching data... (Default: Check8) |
| **FragReassemblyFail** | Failed packet reassembly attempts - due to timeouts or packet losses. (Default: LogSuspectSubseq) |
| **DroppedFrags** | Fragments of packets dropped due to rule base. (Default: LogSuspect) |
| **DuplicateFrags** | Duplicate fragments received. (Default: LogSuspect) |
| **FragmentedICMP** | Fragmented ICMP messages other than Ping; normally invalid. (Default: DropLog) |
| **MinimumFragLength** | Minimum allowed length of non-last fragments. (Default: 8) |
| **ReassTimeout** | Timeout of a reassembly, since previous received fragment. (Default: 65) |
| **ReassTimeLimit** | Maximum lifetime of a reassembly, since first received fragment. (Default: 90) |
| **ReassDoneLinger** | How long to remember a completed reassembly (watching for old dups). (Default: 20) |
| **ReassIllegalLinger** | How long to remember an illegal reassembly (watching for more fragments). (Default: 60) |
| **IP6IllegalFrags** | Illegaly constructed fragments; partial overlaps, bad sizes, etc. (Default: DropLog) |
| **IP6DuplicateFragData** | On receipt of duplicate fragments, verify matching data... (Default: Check8) |
| **IP6FragReassemblyFail** | Failed packet reassembly attempts - due to timeouts or packet losses. (Default: LogSuspectSubseq) |
| **IP6DroppedFrags** | Fragments of packets dropped due to rule base. (Default: LogSuspect) |
| **IP6DuplicateFrags** | Duplicate fragments received. (Default: |

LogSuspect)

**IP6RejectBadFragLength**     Send Parameter Problem error upon reception of fragments with bad data length. (Default: No)

**IP6IgnoreStubFrags**     Ignore fragments with M flag cleared and fragment offset zero. (Default: No)

**IP6MinimumFragLength**     Minimum allowed length of non-last fragments. (Default: 8)

**IP6ReassTimeout**     Timeout of a reassembly, since previous received fragment. (Default: 65)

**IP6ReassTimeLimit**     Maximum lifetime of a reassembly, since first received fragment. (Default: 90)

**IP6ReassDoneLinger**     How long to remember a completed reassembly (watching for old dups). (Default: 20)

**IP6ReassIllegalLinger**     How long to remember an illegal reassembly (watching for more fragments). (Default: 60)

**IP6SendErrorOnTimeout**     Send ICMPv6 error when a fragment reassembly time out. (Default: No)

---

***Note***
*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

---

# 3.48. GeolocationFilter

**Description**

The Geolocation Filter allows the system to filter IP addresses based on region.

**Properties**

**Name**                                    Specifies a symbolic name for the rule. (Identifier)

**Regions**                                 Specifies matching regions for this filter. (Optional)

**MatchPrivate**                            Specify if filter should match private networks (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16, fd00::/8). (Default: No)

**MatchUnknown**                            Specify if filter should match unclassified networks. (Default: No)

**Comments**                                Text describing the current object. (Optional)

# 3.49. GotoRule

**Description**

A goto rule specifies what IP rule set to match IP rules in for traffic that matches the specified filter criteria.

**Properties**

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the rule. (Optional) |
| **Action** | Goto Action. (Default: Goto) |
| **RuleSet** | Where to redirect rule lookup. |
| **SourceInterface** | Specifies the name of the receiving interface to be compared to the received packet. |
| **DestinationInterface** | Specifies the destination interface to be compared to the received packet. |
| **SourceNetwork** | Specifies the sender span of IP addresses to be compared to the received packet. |
| **DestinationNetwork** | Specifies the span of IP addresses to be compared to the destination IP of the received packet. |
| **Service** | Specifies a service that will be used as a filter parameter when matching traffic with this rule. |
| **Schedule** | By adding a schedule to a rule, the firewall will only allow that rule to trigger at those designated times. (Optional) |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> *If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.50. GRETunnel

### Description

A GRE interface is a Generic Routing Encapsulation (no encryption, no authentication, only encapsulation) tunnel over an existing IP network.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **IP** | Specifies the IP address of the GRE interface. |
| **Network** | Specifies the network address of the GRE interface. |
| **RemoteEndpoint** | Specifies the IP address of the remote endpoint. |
| **EncapsulationChecksum** | Add an extra level of checksum above the one provided by the IPv4 layer. (Default: No) |
| **OriginatorIPType** | Specifies what IP address to use as source IP in e.g. NAT. (Default: LocalInterface) |
| **OriginatorIP** | Manually specified originator IP address to use as source IP in e.g. NAT. |
| **Metric** | Specifies the metric for the auto-created route. (Default: 90) |
| **AutoInterfaceNetworkRoute** | Automatically add a route for this interface using the given remote network. (Default: Yes) |
| **OutgoingRoutingTable** | The outer PBR Table to use. (Default: main) |
| **UseSessionKey** | Specify whether or not to use a session key. (Default: No) |
| **SessionKey** | Session key. (Default: 0) |
| **SNMPIndex** | Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0) |
| **MemberOfRoutingTable** | All or Specific. (Default: All) |
| **RoutingTable** | Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

# 3.51. HighAvailability

### Description

Configure the High Availability cluster parameters for this system.

### Properties

| | |
|---|---|
| **Enabled** | Enable high availability. (Default: No) |
| **ClusterID** | A (locally) unique cluster ID to use in identifying this group of HA firewalls. (Default: 0) |
| **SyncIface** | Specifies the interface used for state synchronization. |
| **NodeID** | Master or Slave. (Default: Master) |
| **HASyncBufSize** | How much sync data, in KB, to buffer while waiting for acknowledgments from the cluster peer. (Default: 4096) |
| **HASyncMaxPktBurst** | The maximum number of state sync packets to send in a burst. (Default: 100) |
| **HAInitialSilence** | The number of seconds to stay silent on startup or after reconfiguration. (Default: 5) |
| **UseUniqueSharedMac** | Use a unique shared mac address for each interface. (Default: Yes) |
| **HADeactivateBeforeReconf** | Deactivate(hand over) before Reconfiguration if Active. (Default: Yes) |
| **ReconfFailoverTime** | Number of non-responsive seconds before failover at HA reconf (0=immediate failover). (Default: 0) |
| **HAFailoverTime** | Number of milliseconds before failover when active HA node becomes non-responsive. (Default: 750) |

### Note

*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.52. HTTPALGBanners

### Description

HTTP banner files specifies the look and feel of HTTP ALG restriction web pages.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the HTTP Banner Files. (Identifier) |
| **CompressionForbidden** | HTML for the CompressionForbidden.html web page. |
| **ContentForbidden** | HTML for the ContentForbidden.html web page. |
| **URLForbidden** | HTML for the URLForbidden.html web page. |
| **RestrictedSiteNotice** | HTML for the RestrictedSiteNotice.html web page. |
| **ReclassifyURL** | HTML for the ReclassifyURL.html web page. |
| **Comments** | Text describing the current object. (Optional) |

# 3.53. HTTPAuthBanners

**Description**

HTTP banner files specifies the look and feel of HTML authentication web pages.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the HTTP Banner Files. (Identifier) |
| **FormLogin** | HTML for the FormLogin.html web page. |
| **LoginSuccess** | HTML for the LoginSuccess.html web page. |
| **LoginFailure** | HTML for the LoginFailure.html web page. |
| **LoginAlreadyDone** | HTML for the LoginAlreadyDone.html web page. |
| **LoginChallenge** | HTML for the LoginChallenge.html web page. |
| **LoginChallengeTimeout** | HTML for the LoginChallenge.html Timeout' web page. |
| **LogoutSuccess** | HTML for the LogoutSuccess.html web page. |
| **LogoutSuccessBasicAuth** | HTML for the LogoutSuccessBasicAuth.html web page. |
| **LogoutFailure** | HTML for the LogoutFailure.html web page. |
| **FileNotFound** | HTML for the FileNotFound.html web page. |
| **Comments** | Text describing the current object. (Optional) |

# 3.54. HTTPPoster

### Description

Use the HTTP poster for dynamic DNS or automatic logon to services using web-based authentication.

### Properties

| | |
|---|---|
| **URL** | The URL that will be posted when the firewall is loaded. |
| **RepostDelay** | Delay in seconds until the URL is refetched. (Default: 1200) |
| **AlwaysRepost** | Repost on each reconfiguration. (Default: No) |
| **PostValues** | HTTP POST the values. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

> ***Note***
> *If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.55. HWM

### Description

Hardware Monitoring allows monitoring of hardware sensors.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. |
| **Type** | Type of monitoring. |
| **Sensor** | Sensor index. |
| **MinLimit** | Lower limit. (Optional) |
| **MaxLimit** | Upper limit. (Optional) |
| **EnableMonitoring** | Enable/disable monitoring. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

> ### *Note*
> *If no* Index *is specified when creating an instance of this type, the object will be placed last in the list and the* Index *will be equal to the length of the list.*

# 3.56. HWMSettings

**Description**

General settings for Hardware Monitoring

**Properties**

| | |
|---|---|
| **EnableSensors** | Enable/disable all HWM functionality. (Default: No) |
| **SensorPollInterval** | Sensor polling interval. (Default: 500) |
| **MemoryPollInterval** | Memory polling interval in minutes. (Default: 15) |
| **MemoryUsePercent** | Should mem monitor use percentage as unit for monitoring, else it is megabyte. (Default: Yes) |
| **MemoryLogRepetition** | Should a log message be sent for each poll result that is in the Alert, Critical or Warning level, or should a log message only be sent when a new level is reached. (Default: No) |
| **MemoryAlertLevel** | Alert log message if free memory is below this value, disable by using 0. (Default: 0) |
| **MemoryCriticalLevel** | Critical log message if free memory is below this value, disable by using 0. (Default: 0) |
| **MemoryWarningLevel** | Warning log message if free memory is below this value, disable by using 0. (Default: 0) |

> **Note**
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.57. ICMPSettings

**Description**

Settings related to the ICMP protocol.

**Properties**

**ICMPSendPerSecLimit**    Maximum number of ICMP responses that will be sent each second. (Default: 500)

**SilentlyDropStateICMPErrors**    Silently drop ICMP errors regarding statefully tracked open connections. (Default: Yes)

**ICMP6MaxOptND**    Total number of options allowed per ICMP6 ND header. (Default: 32)

**ICMP6NDOnMaxOptND**    Validate the number of options per extension header when it goes beyond ICMP6MaxOptND. (Default: DropLog)

> *Note*
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.58. IDList

**Description**

An ID list contains IDs, which are used within the authentication process when establishing an IPsec tunnel.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the ID list. (Identifier) |
| **Comments** | Text describing the current object. (Optional) |

# 3.58.1. ID

**Description**

An ID is used to define parameters that are matched against the subject field in an X.509 certificate when establishing an IPsec tunnel.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. (Identifier) |
| **Type** | IP, DNS, E-Mail or Distinguished name. |
| **IP** | IP address. |
| **Hostname** | Host name. |
| **CommonName** | Common name of the owner of the certificate. (Optional) |
| **OrganizationName** | Organization name of the owner of the certificate. (Optional) |
| **OrganizationalUnit** | Organizational unit of the owner of the certificate. (Optional) |
| **Country** | Specifies the country. (Optional) |
| **LocalityName** | Locality. (Optional) |
| **EMailAddress** | E-mail address. (Optional) |
| **DNTuples** | Use the most common DN types, or add tuples as a comma separated list of types. E.g. 'DNTuples={SN;12345}, {S;Smith}' for serial number and surname. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

# 3.59. IDPRule

**Description**

An IDP Rule defines a filter for matching specific network traffic. When the filter criterion is met, the IDP Rule Actions are evaluated and possible actions taken.

**Properties**

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the rule. (Optional) |
| **SourceInterface** | Specifies the name of the receiving interface to be compared to the received packet. |
| **SourceNetwork** | Specifies the sender span of IP addresses to be compared to the received packet. |
| **DestinationInterface** | Specifies the destination interface to be compared to the received packet. |
| **DestinationNetwork** | Specifies the span of IP addresses to be compared to the destination IP of the received packet. |
| **Service** | Specifies a service that will be used as a filter parameter when matching traffic with this rule. |
| **Schedule** | By adding a schedule to a rule, the firewall will only allow that rule to trigger at those designated times. (Optional) |
| **InsertionEvasion** | Protect against insertion/evastion attacks. (Default: Yes) |
| **URIIllegalUTF8** | Specifies what action to take if invalid UTF-8 characters are seen in a HTTP URI. (Default: Log) |
| **URIIllegalHex** | Specifies what action to take when invalid hexencoding (%xx) is seen in a HTTP URI. (Default: DropLog) |
| **URIDoubleEncode** | Specifies what action to take when seeing double encoded characters in a HTTP URI. (Default: Ignore) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.

## 3.59.1. IDPRuleAction

**Description**

An IDP Rule Action specifies what signatures to search for in the network traffic, and what action to take if those signatures are found.

**Properties**

| | |
|---|---|
| **Action** | Specifies what action to take if the given signature is found. (Default: Audit) |
| **Signatures** | Specifies what signature(s) to search for in the network traffic. (Optional) |
| **ZoneDefense** | Activate ZoneDefense. (Default: No) |
| **BlackList** | Activate BlackList. (Default: No) |
| **BlackListTimeToBlock** | The number of seconds that the dynamic black list should remain. (Optional) |
| **BlackListBlockOnlyService** | Only block the service that triggered the blacklisting. (Default: No) |
| **BlackListIgnoreEstablished** | Do not drop existing connection. (Default: No) |
| **PipeLimit** | Specifies the bandwidth limit in kbps for hosts triggered by this action. |
| **PipeNetwork** | Traffic shaping will only apply to hosts that are within this network. (Default: 0/0) |
| **PipeNewConnections** | Enable piping of new connections from and to the same host. (Default: No) |
| **PipeTimeWindow** | Throttling of new connections to and from the triggering host will stop after the configured amount of time. (Default: 10) |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> *If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.60. IGMPRule

**Description**

An IGMP rule specifies how to handle inbound IGMP reports and outbound IGMP queries.

**Properties**

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the rule. (Optional) |
| **Type** | The type of IGMP messages the rule applies to. (Default: Report) |
| **Action** | Drop, Snoop, Proxy or PIM. (Default: Drop) |
| **SourceInterface** | Specifies the name of the receiving interface to be compared to the received packet. |
| **SourceNetwork** | Specifies the sender span of IP addresses to be compared to the received packet. |
| **DestinationInterface** | Specifies the destination interface to be compared to the received packet. (Default: core) |
| **MulticastGroup** | Specifies the multicast group to be compared to the received packet. |
| **MulticastSource** | Specifies the multicast source to be compared to the received packet. |
| **RelayInterface** | Specifies the interface via which to relay IGMP messages. |
| **TranslateMGroup** | Translate the multicast group for packets matching this rule. (Default: No) |
| **GrpAllToOne** | Rewrite all multicast groups to a single IP. (Default: No) |
| **NewGrpIP** | Translate the multicast group to this address. |
| **TranslateMSource** | Translate the multicast source for packets matching this rule. (Default: No) |
| **SrcAllToOne** | Rewrite all multicast sources to a single IP. (Default: No) |
| **NewSrcIP** | Translate the multicast source to this address. |
| **Filter** | Pass IGMP data not matching this rule to the next rule. (Default: Yes) |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

**Note**

*If no* Index *is specified when creating an instance of this type, the object will be placed last in the list and the* Index *will be equal to the length of the list.*

# 3.61. IGMPSetting

**Description**

IGMP parameters can be tuned for one, or a group of interfaces in order to match the characteristics of a network.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. (Identifier) |
| **Interface** | The interfaces that these settings should apply to. |
| **RobustnessVariable** | IGMP is robust to (Robustness Variable - 1) packet losses. (Default: 2) |
| **MaxRequestsPerSecond** | Maximum number of IGMP requests to process each second and interface. (Default: 100) |
| **RouterVersion** | Multiple IGMP querying routers on a network must use the same IGMP version. (Default: IGMPv3) |
| **LowestCompatibleVersion** | The lowest IGMP version to allow on incoming requests. (Default: IGMPv1) |
| **QueryInterval** | The interval between general queries sent by the firewall. (Default: 125000) |
| **QueryResponseInterval** | The maximum time until a host (client) has to send an answer to a query. (Default: 10000) |
| **LastMemberQueryInterval** | The maximum time until a host (client) has to send an answer to a group and group-and-source specific query. (Default: 10000) |
| **LastMemberQueryCount** | The number of group and group-and-source specific queries sent until the firewall decides there are no more subscribers to a specific multicast group. (Default: 2) |
| **StartupQueryInterval** | The general query interval to use during the startup phase. (Default: 30000) |
| **StartupQueryCount** | The number of startup queries to send during the startup phase. (Default: 2) |
| **UnsolicatedReportInterval** | The time between repetitions of a host's initial membership reports to a group. (Default: 1000) |
| **ReactToOwnQueries** | Should the system respond to Member Report Queries orginating from itself. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

# 3.62. IKEAlgorithms

### Description

Configure algorithms which are used in the IKE phase of an IPsec session.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. (Identifier) |
| **DESEnabled** | Enable DES encryption algorithm. (Default: No) |
| **DES3Enabled** | Enable 3DES encryption algorithm. (Default: No) |
| **AESEnabled** | Enable AES encryption algorithm. (Default: No) |
| **BlowfishEnabled** | Enable Blowfish encryption algorithm. (Default: No) |
| **TwofishEnabled** | Enable Twofish encryption algorithm. (Default: No) |
| **CAST128Enabled** | Enable CAST128 encryption algorithm. (Default: No) |
| **BlowfishMinKeySize** | Specifies the minimum Blowfish key size in bits. (Default: 128) |
| **BlowfishKeySize** | Specifies the Blowfish preferred key size in bits. (Default: 128) |
| **BlowfishMaxKeySize** | Specifies the maximum Blowfish key size in bits. (Default: 448) |
| **TwofishMinKeySize** | Specifies the minimum Twofish key size in bits. (Default: 128) |
| **TwofishKeySize** | Specifies the Twofish preferred key size in bits. (Default: 128) |
| **TwofishMaxKeySize** | Specifies the maximum Twofish key size in bits. (Default: 256) |
| **AESMinKeySize** | Specifies the minimum AES key size in bits. (Default: 128) |
| **AESKeySize** | Specifies the preferred AES key size in bits. (Default: 128) |
| **AESMaxKeySize** | Specifies the maximum AES key size in bits. (Default: 256) |
| **MD5Enabled** | Enable MD5 integrity algorithm. (Default: No) |
| **SHA1Enabled** | Enable SHA1 integrity algorithm. (Default: No) |
| **SHA256Enabled** | Enable SHA256 integrity algorithm. (Default: No) |
| **SHA512Enabled** | Enable SHA512 integrity algorithm. (Default: No) |

**XCBCEnabled**                                                          Enable AES-XCBC integrity algorithm. (Default: No)

**Comments**                                                              Text describing the current object. (Optional)

# 3.63. InterfaceGroup

**Description**

Use an interface group to combine several interfaces for a simplified security policy.

**Properties**

**Name**                          Specifies a symbolic name for the interface.
                                  (Identifier)

**Equivalent**                    Specifies if the interfaces should be considered
                                  security equivalent, that means that if enabled the
                                  interface group can be used as a destination
                                  interface in rules where connections might need to
                                  be moved between the two interfaces. (Default:
                                  No)

**Members**                       Specifies the interfaces that are included in the
                                  interface group.

**Comments**                      Text describing the current object. (Optional)

# 3.64. IP6in4Tunnel

### Description

A 6in4 tunnel (no encryption, no authentication, only encapsulation) allows tunneling of IPv6 packets over an existing IPv4 network.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **IP** | Specifies the IPv6 address of the 6in4 tunnel interface. |
| **Network** | Specifies the remote IPv6 network of the 6in4 interface. |
| **RemoteEndpoint** | Specifies the IPv4 address of the remote endpoint. |
| **OriginatorIPType** | Specifies what IPv4 address to use as source IP for the encapsulated IPv6 packets. (Default: LocalInterface) |
| **OriginatorIP** | Manually specified IPv4 address to use as source IP for the encapsulated IPv6 packets. |
| **Metric** | Specifies the metric for the auto-created route. (Default: 90) |
| **AutoInterfaceNetworkRoute** | Automatically add a route for this interface using the given remote network. (Default: Yes) |
| **MTU** | Specify the Maximum Transmission Unit for IPv6 packets entering this tunnel. (Default: 1280) |
| **OutgoingRoutingTable** | The outer PBR Table to use when communicating with the remote endpoint. (Default: main) |
| **SNMPIndex** | Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0) |
| **MemberOfRoutingTable** | All or Specific. (Default: All) |
| **RoutingTable** | Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

# 3.65. IPPolicy

### Description

An IP Policy specifies what action to perform on network traffic that matches the specified filter criteria.

### Properties

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the policy. |
| **Action** | Allow or Deny. (Default: Allow) |
| **Reject** | Drop the packet and respond with an ICMP error or TCP reset. (Default: No) |
| **SourceAddressTranslation** | Action to take on source address. (Default: Auto) |
| **NATSourceAddressAction** | Specify method to determine which sender address to use. (Default: OutgoingInterfaceIP) |
| **SATSourceAddressAction** | Specify method to determine which sender address to use. |
| **SourceNewIP** | Specifies which sender address will be used. |
| **SourceBaseIP** | Specifies base address for sender address. |
| **SourceNATPool** | Specifies NAT Pool to fetch sender address to be used. |
| **SourcePortAction** | Specify method to determine which port action to use. (Default: None) |
| **SourceNewSinglePort** | Translate to this port. (Optional) |
| **SourceBasePort** | Transpose using this port as base. (Optional) |
| **DestAddressTranslation** | Action to take on destination address. (Default: None) |
| **DestAddressAction** | Specify method to determine which destination address to use. |
| **DestNewIP** | Specifies which destination address will be used. |
| **DestBaseIP** | Specifies base address for destination address. |
| **DestPortAction** | Specify method to determine which port action to use. (Default: None) |
| **DestNewSinglePort** | Translate to this port. (Optional) |
| **DestBasePort** | Transpose using this port as base. (Optional) |
| **AntiVirus** | Anti-Virus scanning. (Default: No) |
| **AV_Mode** | Anti-Virus mode. (Default: UsePolicy) |

| | |
|---|---|
| **AV_Policy** | Selects preconfigured Anti-Virus Profile. |
| **AV_AuditMode** | Anti-Virus audit mode. (Default: No) |
| **AV_ScanExclude** | List of files to exclude from antivirus scanning. (Optional) |
| **AV_CompressionRatio** | A compression ratio higher than this value will trigger the action in Compression Ratio Action, a value of zero will disable all compression checks. (Default: 20) |
| **AV_CompressionRatioAction** | The action to take when high compression threshold is violated, all actions are logged. (Default: Drop) |
| **AV_AllowEncryptedZip** | Allow encrypted zip files, even though the contents can not be scanned. (Default: No) |
| **AV_MaxArchiveDepth** | The maximum number of archive file "layers" that the antivirus engine will extract. (Default: 5) |
| **AV_ZDEnabled** | Enable ZoneDefense Block. (Default: No) |
| **AV_ZDNetwork** | Hosts within this network will be blocked at switches if a virus is found. |
| **AV_FailModeBehavior** | General behavior when anti-virus scanning fails. The data can either be allowed or denied. (Default: Deny) |
| **WebControl** | Web Control. (Default: No) |
| **Web_Policy** | Selects preconfigured Web Profile. |
| **FileControl** | File Control. (Default: No) |
| **FC_Mode** | File Control mode. (Default: UsePolicy) |
| **FC_Policy** | Selects preconfigured File Control Profile. |
| **FC_ListType** | Specifies if the file list contains files to allow or deny. (Default: Block) |
| **FC_FileExtension** | List of file types to allow or deny. (Optional) |
| **FC_VerifyContentMimetype** | Verify that file extentions correspond to the MIME type. (Default: No) |
| **AppControl** | Application Control. (Default: No) |
| **AC_Mode** | Application Control mode. (Default: UsePolicy) |
| **AC_RuleSet** | Selects preconfigured Application Rule. |
| **AC_AppAction** | Allow or Deny selected applications. (Default: Allow) |
| **AC_Applications** | List of applications to match. |
| **EmailControl** | Email Control. (Default: No) |
| **EC_Policy** | Selects preconfigured Email Control Profile. |

| | |
|---|---|
| **VoIP** | Voice over IP. (Default: No) |
| **VoIP_Policy** | Selects preconfigured VoIP Profile. |
| **DNS** | DNS. (Default: No) |
| **DNS_Policy** | Selects preconfigured DNS Profile. |
| **FTPControl** | Enables FTP protocol specific settings. (Default: No) |
| **FTPAllowServerPassive** | Allow server to use passive mode (unsafe for server). (Default: Yes) |
| **FTPServerPorts** | Server data ports. (Default: 1024-65535) |
| **FTPAllowClientActive** | Allow client to use active mode (unsafe for client). (Default: Yes) |
| **FTPClientPorts** | Client data ports. (Default: 1024-65535) |
| **FTPAllowUnknownCommands** | Allow unknown commands. (Default: No) |
| **FTPAllowSITEEXEC** | Allow SITE EXEC. (Default: No) |
| **FTPMaxLineLength** | Maximum line length in control channel. (Default: 256) |
| **FTPMaxCommandRate** | Maximum number of commands per second. (Default: 20) |
| **FTPAllow8BitStrings** | Allow 8-bit strings in control channel. (Default: Yes) |
| **FTPAllowResumeTransfer** | Allow RESUME even in case of content scanning. (Default: No) |
| **TFTPControl** | Enables TFTP protocol specific settings. (Default: No) |
| **TFTPAllowedCommands** | Specifies allowed commands. (Default: ReadWrite) |
| **TFTPRemoveOptions** | Remove option part from request packet. (Default: No) |
| **TFTPAllowUnknownOptions** | Allow unknown options in request packet. (Default: No) |
| **TFTPMaxBlocksize** | Max value for the blksize option. (Optional) |
| **TFTPMaxFileTransferSize** | Max size for transferred file. (Optional) |
| **TFTPBlockDirectoryTraversal** | Prevent directory traversal (consecutive dots in filenames). (Default: No) |
| **PPTPControl** | Enables PPTP protocol specific settings. (Default: No) |
| **PPTPEchoTimeout** | Specifies idle timeout for Echo messages in the PPTP tunnel. (Default: 0) |
| **PPTPIdleTimeout** | SPecifies idle timeout for user traffic in the PPTP tunnel. (Default: 0) |
| **TLSControl** | Enables TLS protocol specific settings. (Default: No) |

*209*

**TLSHostCert**                          Specifies the host certificate.

**TLSRootCert**                          Specifies the root certificates. (Optional)

**HTTPInspection**                       Enables HTTP protocol validation and logging of URLs. (Default: No)

**HTTPAllowUnknownProtocols**            Allow non-HTTP protocols to pass through without inspection. (Default: No)

**SourceInterface**                      Specifies the name of the receiving interface to be compared to the received packet.

**DestinationInterface**                 Specifies the destination interface to be compared to the received packet.

**SourceNetwork**                        Specifies the sender span of IP addresses to be compared to the received packet.

**DestinationNetwork**                   Specifies the span of IP addresses to be compared to the destination IP of the received packet.

**SourceGeoFilter**                      Specifies the region filter to be compared against the sender Geolocation of the received packet. (Optional)

**DestinationGeoFilter**                 Specifies the region filter to be compared against the destination Geolocation of the received packet. (Optional)

**Service**                              Specifies a service that will be used as a filter parameter when matching traffic with this rule.

**Schedule**                             By adding a schedule to a rule, the firewall will only allow that rule to trigger at those designated times. (Optional)

**LogEnabled**                           Enable logging. (Default: Yes)

**LogSeverity**                          Specifies with what severity log events will be sent to the specified log receivers. (Default: Default)

**Comments**                             Text describing the current object. (Optional)

---

**Note**

*If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.66. IPPool

### Description

An IP Pool is a dynamic object which consists of IP leases that are fetched from a DHCP Server. The IP Pool is used as an address source by subsystems that may need to distribute addresses, e.g. by IPsec in Configuration mode.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the IP Pool. (Identifier) |
| **DHCPServerType** | Should server address be specified or should broadcast on a interface be used. (Default: Interface) |
| **ServerIP** | DHCP Server Address. |
| **ServerFilter** | Specifies which DHCP server that leases should be accepted from. (Optional) |
| **Interface** | Specifies the interface which has the DHCP server that leases are accepted from. |
| **IPFilter** | Specifies which IP addresses that are accepted from the DHCP server. (Optional) |
| **RoutingTable** | The routing table to use in communication with the DHCP server. (Default: main) |
| **ReceiveInterface** | Which interface to use when communicating with the DHCP server. (Optional) |
| **PrefetchLeases** | Specifies the number of leases an IP Pool will keep prefetched. (Default: 3) |
| **MaxFree** | Maximum number of free address that the IP pool will keep, others will be returned back to DCHP server. (Optional) |
| **MaxClients** | Maximum number clients that the IP pool is allowed to contain. (Optional) |
| **MacRangeStart** | Specifies the lower boundary of MAC addresses that DCHP Clients will use in communication with a server. (Optional) |
| **MacRangeEnd** | Specifies the upper boundary of MAC addresses that DCHP Clients will use in communication with a server. (Optional) |
| **SenderIP** | The local IP that should be used when communication with the DHCP server. (Optional) |
| **AscendingFreeList** | Enabling this will result in the IPs being fetched in a predictable manner from the free list. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

# 3.67. IPRule

## Description

An IP rule specifies what action to perform on network traffic that matches the specified filter criteria.

## Properties

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the rule. (Optional) |
| **Action** | Reject, Drop, FwdFast, Allow, NAT, SAT or SLB_SAT. |
| **SourceInterface** | Specifies the name of the receiving interface to be compared to the received packet. |
| **DestinationInterface** | Specifies the destination interface to be compared to the received packet. |
| **SourceNetwork** | Specifies the sender span of IP addresses to be compared to the received packet. |
| **DestinationNetwork** | Specifies the span of IP addresses to be compared to the destination IP of the received packet. |
| **Service** | Specifies a service that will be used as a filter parameter when matching traffic with this rule. |
| **Schedule** | By adding a schedule to a rule, the firewall will only allow that rule to trigger at those designated times. (Optional) |
| **NATAction** | Specify sender address or Use interface address. (Default: UseInterfaceAddress) |
| **NATSenderAddress** | Specifies which sender address will be used. |
| **NATPool** | Specifies the NATPool object to use. |
| **SATTranslate** | Specifies whether to translate source IP or destination IP. (Default: DestinationIP) |
| **SATTranslateToIP** | Translate to this IP address. |
| **SATTranslateToPort** | Translate to this port. (Optional) |
| **SATAllToOne** | Rewrite all destination IPs to a single IP. (Default: No) |
| **SLBAddresses** | The IP addresses of the servers in the server farm. |
| **SLBStickiness** | Specifies stickiness mode. (Default: None) |
| **SLBIdleTimeOut** | New connections that arrive within the idle timeout are assigned to the same real server as previous connections from that address. The timeout is refreshed after each new connection. (Default: 30) |

**SLBMaxSlots**                          Specifies maximum number of slots for IP and network stickiness. (Default: 2048)

**SLBNetSize**                           Specifies network size for network stickiness. (Default: 24)

**SLBNewPort**                           Rewrite destination port to this port. (Optional)

**SLBMonitorRoutingTable**               Routing table used for server monitoring. (Default: main)

**SLBMonitorPing**                       Enable monitoring using ICMP Ping packets. (Default: No)

**SLBPingPollingInterval**               Delay in milliseconds between each ping interval. (Default: 5000)

**SLBPingSamples**                       Specifies the number of attempts to use for statistical calculations. (Default: 10)

**SLBPingMaxPollFails**                  Specifies the maximum number of failed ping attempts until host is considered to be unreachable. (Default: 2)

**SLBPingMaxAverageLatency**             Specifies the max average latency for the sample attempts. (Default: 800)

**SLBMonitorTCP**                        Enable monitoring using TCP handshakes. (Default: No)

**SLBTCPPorts**                          Specifies the ports that will be monitored.

**SLBTCPPollingInterval**                Delay in milliseconds between each TCP handshake. (Default: 10000)

**SLBTCPSamples**                        Specifies the number of attempts to use for statistical calculations. (Default: 10)

**SLBTCPMaxPollFails**                   Specifies the maximum number of failed TCP attempts until host is considered to be unreachable. (Default: 2)

**SLBTCPMaxAverageLatency**              Specifies the max average latency for the sample attempts. (Default: 800)

**SLBMonitorHTTP**                       Enable monitoring using HTTP requests. (Default: No)

**SLBHTTPPorts**                         Specifies the ports that will be monitored. (Default: 80)

**SLBHTTPPollingInterval**               Delay in milliseconds between each monitor interval. (Default: 10000)

**SLBHTTPSamples**                       Specifies the number of attempts to use for statistical calculations. (Default: 10)

**SLBHTTPMaxPollFails**                  Specifies the maximum number of failed HTTP attempts until host is considered to be unreachable. (Default: 2)

**SLBHTTPMaxAverageLatency**             Specifies the max average latency for the sample

*213*

|  | attempts. (Default: 800) |
| --- | --- |
| **SLBHTTPURLType** | Defines how the request URL should be interpreted. (Default: FQDN) |
| **SLBHTTPRequestURL** | Specifies the HTTP URL to monitor. |
| **SLBHTTPExpectedResponse** | Expected HTTP response. (Optional) |
| **SLBMonitorReset** | Reset active connections when monitor fail. Uses additional resources to track all connections. (Default: No) |
| **SLBDistribution** | Specifies the algorithm used for the load distribution tasks. (Default: RoundRobin) |
| **SLBWindowTime** | Specifies the window time used for counting the number of seconds back in time to summarize the number of new connections for connection-rate algorithm. (Default: 10) |
| **SLBServerId** | Identifier used when uploading server state. |
| **RequireIGMP** | Multicast traffic must have been requested using IGMP before it is forwarded. (Default: Yes) |
| **MultiplexArgument** | Specifies how the traffic should be forwarded and translated. |
| **MultiplexAllToOne** | Rewrite all destination IPs to a single IP. (Default: No) |
| **AppControl** | Application Control. (Default: No) |
| **AC_Mode** | Application Control mode. (Default: UsePolicy) |
| **AC_RuleSet** | Selects preconfigured Application Rule. |
| **AC_AppAction** | Allow or Deny selected applications. (Default: Allow) |
| **AC_Applications** | List of applications to match. |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

> ### Note
> *If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.68. IPRuleFolder

## Description

An IP Rule Folder can be used to group IP Rules into logical groups for better overview and simplified management.

**Properties**

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies the name of the folder. |
| **Comments** | Text describing the current object. (Optional) |

> ***Note***
> *If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

## 3.68.1. IPPolicy

The definitions here are the same as in Section 3.65, "IPPolicy" .

## 3.68.2. SLBPolicy

### Description

Server Load Balancing using Static Address Translation. Allows distribution of client requests over a number of servers.

**Properties**

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the policy. |
| **SLBAddresses** | The IP addresses of the servers in the server farm. |
| **SLBStickiness** | Specifies stickiness mode. (Default: None) |
| **SLBIdleTimeOut** | New connections that arrive within the idle timeout are assigned to the same real server as previous connections from that address. The timeout is refreshed after each new connection. (Default: 30) |
| **SLBMaxSlots** | Specifies maximum number of slots for IP and network stickiness. (Default: 2048) |
| **SLBNetSize** | Specifies network size for network stickiness. (Default: 24) |

| | |
|---|---|
| **SLBNewPort** | Rewrite destination port to this port. (Optional) |
| **SLBMonitorRoutingTable** | Routing table used for server monitoring. (Default: main) |
| **SLBMonitorPing** | Enable monitoring using ICMP Ping packets. (Default: No) |
| **SLBPingPollingInterval** | Delay in milliseconds between each ping interval. (Default: 5000) |
| **SLBPingSamples** | Specifies the number of attempts to use for statistical calculations. (Default: 10) |
| **SLBPingMaxPollFails** | Specifies the maximum number of failed ping attempts until host is considered to be unreachable. (Default: 2) |
| **SLBPingMaxAverageLatency** | Specifies the max average latency for the sample attempts. (Default: 800) |
| **SLBMonitorTCP** | Enable monitoring using TCP handshakes. (Default: No) |
| **SLBTCPPorts** | Specifies the ports that will be monitored. |
| **SLBTCPPollingInterval** | Delay in milliseconds between each TCP handshake. (Default: 10000) |
| **SLBTCPSamples** | Specifies the number of attempts to use for statistical calculations. (Default: 10) |
| **SLBTCPMaxPollFails** | Specifies the maximum number of failed TCP attempts until host is considered to be unreachable. (Default: 2) |
| **SLBTCPMaxAverageLatency** | Specifies the max average latency for the sample attempts. (Default: 800) |
| **SLBMonitorHTTP** | Enable monitoring using HTTP requests. (Default: No) |
| **SLBHTTPPorts** | Specifies the ports that will be monitored. (Default: 80) |
| **SLBHTTPPollingInterval** | Delay in milliseconds between each monitor interval. (Default: 10000) |
| **SLBHTTPSamples** | Specifies the number of attempts to use for statistical calculations. (Default: 10) |
| **SLBHTTPMaxPollFails** | Specifies the maximum number of failed HTTP attempts until host is considered to be unreachable. (Default: 2) |
| **SLBHTTPMaxAverageLatency** | Specifies the max average latency for the sample attempts. (Default: 800) |
| **SLBHTTPURLType** | Defines how the request URL should be interpreted. (Default: FQDN) |
| **SLBHTTPRequestURL** | Specifies the HTTP URL to monitor. |

| | |
|---|---|
| **SLBHTTPExpectedResponse** | Expected HTTP response. (Optional) |
| **SLBMonitorReset** | Reset active connections when monitor fail. Uses additional resources to track all connections. (Default: No) |
| **SLBDistribution** | Specifies the algorithm used for the load distribution tasks. (Default: RoundRobin) |
| **SLBWindowTime** | Specifies the window time used for counting the number of seconds back in time to summarize the number of new connections for connection-rate algorithm. (Default: 10) |
| **SLBServerId** | Server identifier (max 32 characters) used when uploading server state to firewall. |
| **SourceInterface** | Specifies the name of the receiving interface to be compared to the received packet. |
| **DestinationInterface** | Specifies the destination interface to be compared to the received packet. |
| **SourceNetwork** | Specifies the sender span of IP addresses to be compared to the received packet. |
| **DestinationNetwork** | Specifies the span of IP addresses to be compared to the destination IP of the received packet. |
| **SourceGeoFilter** | Specifies the region filter to be compared against the sender Geolocation of the received packet. (Optional) |
| **DestinationGeoFilter** | Specifies the region filter to be compared against the destination Geolocation of the received packet. (Optional) |
| **Service** | Specifies a service that will be used as a filter parameter when matching traffic with this rule. |
| **Schedule** | By adding a schedule to a rule, the firewall will only allow that rule to trigger at those designated times. (Optional) |
| **SourceAddressTranslation** | Action to take on source address. (Default: Auto) |
| **NATSourceAddressAction** | Specify method to determine which sender address to use. (Default: OutgoingInterfaceIP) |
| **SATSourceAddressAction** | Specify method to determine which sender address to use. |
| **SourceNewIP** | Specifies which sender address will be used. |
| **SourceBaseIP** | Specifies base address for sender address. |
| **SourceNATPool** | Specifies NAT Pool to fetch sender address to be used. |
| **SourcePortAction** | Specify method to determine which port action to use. (Default: None) |

*217*

| | |
|---|---|
| **SourceNewSinglePort** | Translate to this port. (Optional) |
| **SourceBasePort** | Transpose using this port as base. (Optional) |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

> ### Note
> If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.

# 3.68.3. MulticastPolicy

### Description

Multiplex Static Address Translation. The Multicast rule is used to achieve duplication and forwarding of packets through more than one interface.

### Properties

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the policy. |
| **RequireIGMP** | Multicast traffic must have been requested using IGMP before it is forwarded. (Default: Yes) |
| **MultiplexArgument** | Specifies how the traffic should be forwarded and translated. |
| **MultiplexAllToOne** | Rewrite all destination IPs to a single IP. (Default: No) |
| **SourceInterface** | Specifies the name of the receiving interface to be compared to the received packet. |
| **DestinationInterface** | Specifies the destination interface to be compared to the received packet. |
| **SourceNetwork** | Specifies the sender span of IP addresses to be compared to the received packet. |
| **DestinationNetwork** | Specifies the span of IP addresses to be compared to the destination IP of the received packet. |
| **SourceGeoFilter** | Specifies the region filter to be compared against the sender Geolocation of the received packet. (Optional) |
| **DestinationGeoFilter** | Specifies the region filter to be compared against the destination Geolocation of the received packet. |

(Optional)

**Service**                          Specifies a service that will be used as a filter parameter when matching traffic with this rule.

**Schedule**                         By adding a schedule to a rule, the firewall will only allow that rule to trigger at those designated times. (Optional)

**SourceAddressTranslation**         Action to take on source address. (Default: Auto)

**NATSourceAddressAction**           Specify method to determine which sender address to use. (Default: OutgoingInterfaceIP)

**SATSourceAddressAction**           Specify method to determine which sender address to use.

**SourceNewIP**                      Specifies which sender address will be used.

**SourceBaseIP**                     Specifies base address for sender address.

**SourceNATPool**                    Specifies NAT Pool to fetch sender address to be used.

**SourcePortAction**                 Specify method to determine which port action to use. (Default: None)

**SourceNewSinglePort**              Translate to this port. (Optional)

**SourceBasePort**                   Transpose using this port as base. (Optional)

**LogEnabled**                       Enable logging. (Default: Yes)

**LogSeverity**                      Specifies with what severity log events will be sent to the specified log receivers. (Default: Default)

**Comments**                         Text describing the current object. (Optional)

---

*Note*
*If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

---

# 3.68.4. StatelessPolicy

**Description**

No state is kept between packets which means it is less secure and slower than stateful forwarding.

**Properties**

**Index**                            The index of the object, starting at 1. (Identifier)

**Name**                             Specifies a symbolic name for the policy.

**Action**                           Allow or Deny. (Default: Allow)

| | |
|---|---|
| **Reject** | Drop the packet and respond with an ICMP error or TCP reset. (Default: No) |
| **SourceAddressTranslation** | Action to take on source address. (Default: None) |
| **SATSourceAddressAction** | Specify method to determine which sender address to use. |
| **SourceNewIP** | Specifies which sender address will be used. |
| **SourceBaseIP** | Specifies base address for sender address. |
| **SourcePortAction** | Specify method to determine which port action to use. (Default: None) |
| **SourceNewSinglePort** | Translate to this port. (Optional) |
| **SourceBasePort** | Transpose using this port as base. (Optional) |
| **DestAddressTranslation** | Action to take on destination address. (Default: None) |
| **DestAddressAction** | Specify method to determine which destination address to use. |
| **DestNewIP** | Specifies which destination address will be used. |
| **DestBaseIP** | Specifies base address for destination address. |
| **DestPortAction** | Specify method to determine which port action to use. (Default: None) |
| **DestNewSinglePort** | Translate to this port. (Optional) |
| **DestBasePort** | Transpose using this port as base. (Optional) |
| **SourceInterface** | Specifies the name of the receiving interface to be compared to the received packet. |
| **DestinationInterface** | Specifies the destination interface to be compared to the received packet. |
| **SourceNetwork** | Specifies the sender span of IP addresses to be compared to the received packet. |
| **DestinationNetwork** | Specifies the span of IP addresses to be compared to the destination IP of the received packet. |
| **SourceGeoFilter** | Specifies the region filter to be compared against the sender Geolocation of the received packet. (Optional) |
| **DestinationGeoFilter** | Specifies the region filter to be compared against the destination Geolocation of the received packet. (Optional) |
| **Service** | Specifies a service that will be used as a filter parameter when matching traffic with this rule. |
| **Schedule** | By adding a schedule to a rule, the firewall will only allow that rule to trigger at those designated times. (Optional) |

| | |
|---|---|
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.

## 3.68.5. GotoRule

The definitions here are the same as in Section 3.49, "GotoRule" .

## 3.68.6. ReturnRule

### Description

A return rule makes the IP rule scan resume from the goto rule that led to the current IP rule set. If there was no goto rule leading to the current IP rule set the connection is dropped and rule scanning stops.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the rule. (Optional) |
| **Action** | Return Action. (Default: Return) |
| **SourceInterface** | Specifies the name of the receiving interface to be compared to the received packet. |
| **DestinationInterface** | Specifies the destination interface to be compared to the received packet. |
| **SourceNetwork** | Specifies the sender span of IP addresses to be compared to the received packet. |
| **DestinationNetwork** | Specifies the span of IP addresses to be compared to the destination IP of the received packet. |
| **Service** | Specifies a service that will be used as a filter parameter when matching traffic with this rule. |
| **Schedule** | By adding a schedule to a rule, the firewall will only allow that rule to trigger at those designated times. (Optional) |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.

## 3.68.7. IPRule

The definitions here are the same as in Section 3.67, "IPRule" .

# 3.69. IPRuleSet

**Description**

An IP Rule Set is a self-contained set of IP Rules. Default action is Drop.

**Properties**

| | |
|---|---|
| **Name** | A name to uniquely identify this IPRuleSet. (Identifier) |
| **Comments** | Text describing the current object. (Optional) |

## 3.69.1. IPPolicy

The definitions here are the same as in Section 3.65, "IPPolicy" .

## 3.69.2. SLBPolicy

The definitions here are the same as in Section 3.68.2, "SLBPolicy" .

## 3.69.3. MulticastPolicy

The definitions here are the same as in Section 3.68.3, "MulticastPolicy" .

## 3.69.4. StatelessPolicy

The definitions here are the same as in Section 3.68.4, "StatelessPolicy" .

## 3.69.5. GotoRule

The definitions here are the same as in Section 3.49, "GotoRule" .

## 3.69.6. ReturnRule

The definitions here are the same as in Section 3.68.6, "ReturnRule" .

## 3.69.7. IPRuleFolder

The definitions here are the same as in Section 3.68, "IPRuleFolder" .

## 3.69.8. IPRule

The definitions here are the same as in Section 3.67, "IPRule" .

# 3.70. IPsecAlgorithms

## Description

Configure algorithms which are used in the IPsec phase of an IPsec session.

## Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. (Identifier) |
| **NULLEnabled** | Enable plaintext. (Default: No) |
| **DESEnabled** | Enable DES encryption algorithm. (Default: No) |
| **DES3Enabled** | Enable 3DES encryption algorithm. (Default: No) |
| **AESEnabled** | Enable AES encryption algorithm. (Default: No) |
| **BlowfishEnabled** | Enable Blowfish encryption algorithm. (Default: No) |
| **TwofishEnabled** | Enable Twofish encryption algorithm. (Default: No) |
| **CAST128Enabled** | Enable CAST128 encryption algorithm. (Default: No) |
| **BlowfishMinKeySize** | Specifies the minimum Blowfish key size in bits. (Default: 128) |
| **BlowfishKeySize** | Specifies the Blowfish preferred key size in bits. (Default: 128) |
| **BlowfishMaxKeySize** | Specifies the maximum Blowfish key size in bits. (Default: 448) |
| **TwofishMinKeySize** | Specifies the minimum Twofish key size in bits. (Default: 128) |
| **TwofishKeySize** | Specifies the Twofish preferred key size in bits. (Default: 128) |
| **TwofishMaxKeySize** | Specifies the maximum Twofish key size in bits. (Default: 256) |
| **AESMinKeySize** | Specifies the minimum AES key size in bits. (Default: 128) |
| **AESKeySize** | Specifies the preferred AES key size in bits. (Default: 128) |
| **AESMaxKeySize** | Specifies the maximum AES key size in bits. (Default: 256) |
| **MD5Enabled** | Enable MD5 integrity algorithm. (Default: No) |
| **SHA1Enabled** | Enable SHA1 integrity algorithm. (Default: No) |
| **SHA256Enabled** | Enable SHA256 integrity algorithm. (Default: No) |

**SHA512Enabled**                          Enable SHA512 integrity algorithm. (Default: No)

**XCBCEnabled**                            Enable AES-XCBC integrity algorithm. (Default: No)

**Comments**                               Text describing the current object. (Optional)

# 3.71. IPsecTunnel

### Description

An IPsec tunnel item is used to define IPsec endpoint and will appear as a logical interface in the system.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **LocalNetwork** | The network on "this side" of the IPsec tunnel. The IPsec tunnel will be established between this network and the remote network. |
| **RemoteNetwork** | The network connected to the remote gateway. The IPsec tunnel will be established between the local network and this network. |
| **RemoteEndpoint** | Specifies the IP address of the remote endpoint. This is the address the firewall will establish the IPsec tunnel to. It also dictates from where inbound IPsec tunnels are allowed. (Optional) |
| **ConfigMode** | Use config mode to assign unique IP addresses to connecting roaming clients or act as a client. (Default: Off) |
| **IKEConfigModePool** | Selects IKE Config Mode Pool to use for the tunnel. (Optional) |
| **IP** | Netobject that will be assigned an IP when the tunnel is established. Can be used to NAT traffic into the interface. (Optional) |
| **DNS** | Netobject that will be assigned an IP of DNS when the tunnel is established. (Set only if a DNS is assigned by the remote endpoint.). (Optional) |
| **IKEAlgorithms** | Specifies the IKE Proposal list used with the tunnel. (Default: High) |
| **IPsecAlgorithms** | Specifies the IPsec Proposal list used with the tunnel. (Default: High) |
| **IKELifeTimeSeconds** | The lifetime of the IKE connection in seconds. Whenever it expires, a new phase-1 exchange will be performed. (Default: 28800) |
| **IPsecLifeTimeSeconds** | The lifetime of the IPsec connection in seconds. Whenever it's exceeded, a re-key will be initiated, providing new IPsec encryption and authentication session keys. (Default: 3600) |
| **IPsecLifeTimeKilobytes** | The lifetime of the IPsec connection in kilobytes. (Default: 0) |
| **EncapsulationMode** | Specifies if the IPsec tunnel should use Tunnel or |

| | |
|---|---|
| | Transport mode. (Default: Tunnel) |
| **AuthMethod** | Certificate or Pre-shared key. (Default: PSK) |
| **PSK** | Selects the Pre-shared key to use with this IPsec Tunnel. |
| **LocalID** | Specifies the local identity of the tunnel. (Optional) |
| **RemoteID** | Identities authorized to setup a tunnel. If not set, all authenticated peers will be authorized. (Optional) |
| **EnforceLocalID** | Enable if local identity must match any identity proposed by the IKE peer. (Default: No) |
| **GatewayCertificate** | Selects the certificate the firewall uses to authenticate itself to the other IPsec peer. |
| **RootCertificates** | Selects one or more root certificates to use with this IPsec Tunnel. |
| **XAuth** | Required for inbound or Pass to peer gateway. (Default: Off) |
| **XAuthUsername** | Specifies the username to pass to the remote gateway vie IKE XAuth. |
| **XAuthPassword** | Specifies the password to pass to the remote gateway vie IKE XAuth. |
| **AddRouteToRemoteNet** | Dynamically add route to the remote networks when a tunnel is established. (Default: No) |
| **PlaintextMTU** | Specifies the size in bytes at which to fragment plaintext packets (rather than fragmenting IPsec). (Default: 1420) |
| **OriginatorIPType** | Specifies what IP address to use as source IP in e.g. NAT. (Default: LocalInterface) |
| **OriginatorIP** | Manually specified originator IP address to use as source IP in e.g. NAT. |
| **OriginatorHAIP** | Manually specified private originator IP address for use in HA. (Optional) |
| **TunnelMonitor** | Monitor a host inside the tunnel and renegotiatie the tunnel if the host stops answering on ICMP pings. (Default: No) |
| **MonitoredIP** | IP address of the host being monitored with ICMP pings. Source address will be the OriginatorIP configured for the tunnel interface. |
| **MaxLoss** | Specifies how many consecutive ICMP pings must be lost before the tunnel is renegotiated. (Default: 10) |
| **IKEMode** | (IKEv1 only) Specifies which IKE mode to use: main or aggressive. (Default: Main) |
| **IKEVersion** | Specifies the IKE version to use for the tunnel. |

(Default: 2)

| | |
|---|---|
| **DHGroup** | Specifies the Diffie-Hellman group to use when doing key exchanges in IKE. (Default: 2) |
| **PFSDHGroup** | Specifies which Diffie-Hellman group to use with PFS. (Default: None,1,2,5) |
| **SetupSAPer** | Setup security association per network, host or port. (Default: Net) |
| **DeadPeerDetection** | Enable Dead Peer Detection. (Default: Yes) |
| **DeadPeerDetectionInterval** | Specifies the interval between IKE Dead Peer Detecion messages sent if no IKE or ESP message has been received from peer since the last IKE packet sent. (Default: 30) |
| **NATTraversal** | Enable or disable NAT traversal. (Default: OnIfNeeded) |
| **AutoEstablish** | Negotiate tunnel directly after reconfigureation. (Default: No) |
| **Metric** | Specifies the metric for the auto-created route. (Default: 90) |
| **AutoInterfaceNetworkRoute** | Automatically add a route for this interface using the given remote network. (Default: Yes) |
| **IKEIPsecPerIKELimit** | Specifies the maximum number of IPsec SAs one IKE SA is allowed to create. (Default: 0) |
| **IKEMaxIPsecPerIKELimitViolations** | Specifies how many times the IPsec per IKE SA limit can be exceeded before action is taken and the IKE is removed. (Default: 0) |
| **IKEDSField** | Specifies the value of the Differentiated Services Field of the IP header in IKE packets. (Default: 0) |
| **IPsecDSField** | Specifies the value of the Differentiated Services Field of the outer IP header of IPsec packets in tunnel mode. If unspecified, the value of the inner IP header will be used instead. (Optional) |
| **LocalEndpoint** | Specifies on which local address this tunnel should accept incoming IKE/IPsec traffic. (Optional) |
| **IncomingInterfaceFilter** | Specifies which interface this tunnel should use for IKE/IPsec traffic. (Default: any) |
| **OutgoingRoutingTable** | Specifies which routing table this tunnel should use for IKE/IPsec traffic. (Default: main) |
| **RequestEAPID** | Send an EAP identity request to client. This allows the client to use different identities for the IKE and EAP negotiation. (Default: Yes) |
| **EAP** | Use EAP to authenticate either the firewall itself or the connecting peer. (Default: Off) |
| **EAPUsername** | Specifies the username to pass to the remote |

gateway via EAP.

**EAPPassword**                     Specifies the password to pass to the remote gateway via EAP.

**SNMPIndex**                       Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0)

**MemberOfRoutingTable**            All or Specific. (Default: All)

**RoutingTable**                    Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main)

**ProxyARPAllInterfaces**           Always select all interfaces, including new ones, for publishing routes via Proxy ARP. (Default: No)

**ProxyARPInterfaces**              Specifies the interfaces on which the firewall should publish routes via Proxy ARP. (Optional)

**Comments**                        Text describing the current object. (Optional)

# 3.72. IPsecTunnelSettings

**Description**

Settings for the IPsec tunnel interfaces used for establishing IPsec VPN connections to and from this system.

**Properties**

| | |
|---|---|
| **IPsecMaxTunnels** | Amount of IPsec tunnels allowed (0 = automatic). (Default: 0) |
| **IPsecMaxRules** | Amount of IPsec rules allowed (0 = automatic). (Default: 0) |
| **IKESendInitialContact** | Send 'initial contact' messages. (Default: Yes) |
| **IKESendCRLs** | Send CRLs in the IKE exchange. (Default: Yes) |
| **IKECRLValidityTime** | Maximum number of seconds a CRL is considered valid (0=obey the 'next update' field in the CRL). (Default: 86400) |
| **IKEMaxCAPath** | Maximum number of CA certificates in a certificate path. (Default: 15) |
| **IPsecCertCacheMaxCerts** | Maximum number of entries in the certificate cache. (Default: 1024) |
| **IPsecBeforeRules** | Pass IKE & IPsec (ESP/AH) traffic sent to the firewall directly to the IPsec engine without consulting the ruleset. (Default: Yes) |
| **IPsecHardwareAcceleration** | IPsec hardware acceleration. (Default: Coprocessor) |
| **IPsecDisablePKAccel** | Disable hardware acceleration for public-key operations. (Default: No) |
| **AESNIEnable** | Enable AES-NI acceleration for processors that support it. (Default: Yes) |
| **IPsecXCBCFallbackToRFC3664** | Enable fallback to XCBC RFC3664 if XCBC RFC4344 fails when using IKEv2. (Default: Yes) |
| **IPsecDeleteSAOnIPValidationFailure** | Enable tunnel deletion when decrypted source IP address doesn't match the remote net. (Default: No) |
| **IPsecSAKeepTime** | Number of seconds a SA will linger after a delete. (Default: 3) |
| **IPsecForceRequireCookie** | Force requirement of cookies. Used for test purposes only! (Default: No) |
| **IPsecDisableCallingStationID** | Disable calling station ID and called station ID in RADIUS messages. (Default: No) |
| **IpsecUseClientCfgModeAttributes** | Use client requested subnet attributes for config mode. (Default: No) |

**IPsecAllowIKEPortChange**              Allow port change to 4500 in IKE negotiation even
                                         when no NAT is detected. (Default: No)

**IPsecLogKeyMaterial**                  Enable logging of IPsec key material. (Default: No)

**IPsecESPDetectNATChange**              Use inbound ESP packets to detect that NAT
                                         mappings have changed. (Default: Yes)

> ### *Note*
> *This object type does not have an identifier and is identified by the name of the type
> only. There can only be one instance of this type.*

# 3.73. IPSettings

### Description

Settings related to the IP protocol.

### Properties

**IP6LogOnForwardHopLimit0**
Log any attempts of forwarding IPv6 packets with HopLimit=0 destined for outside the firewall; this should never happen! (Default: DropLog)

**IP6AnycastSrc**
Drop Log packets with anycast source address. (Default: DropLog)

**HopLimitMin**
The minimum IP Hop-Limit value accepted on receipt. (Default: 3)

**HopLimitOnLow**
What action to take on too low unicast Hop-Limit values. (Default: DropLog)

**HopLimitMinMulticast**
The minimum IP multicast Hop-Limit value accepted on receipt. (Default: 1)

**HopLimitOnLowMulticast**
What action to take on too low multicast Hop-Limit values. (Default: DropLog)

**DefaultHopLimit**
The default IP Hop-Limit of packets originated by the firewall (32-255). (Default: 255)

**IP6Fl**
Validate IPV6 Flow label header field. (Default: Ignore)

**IP6TC**
Validate IPV6 Traffic class header field. (Default: Ignore)

**IP6MaxExtHdr**
Maximum allowed size of all IP6 extension headers. (Default: 256)

**IP6OnMaxExtHdr**
Validate the extension header length when it goes beyond IP6MaxExtHdr. (Default: DropLog)

**RejectUnorderedExtHdr**
Send an ICMPv6 error when encountering extension headers out of order. (Default: No)

**IP6MaxOptHdr**
Total number of options allowed per IP6 extension header. (Default: 8)

**IP6OnMaxOptHdr**
Validate the number of options per extension header when it goes beyond IP6MaxOptHdr. (Default: DropLog)

**IP6ValidateSyntax**
Validate ipv6 syntax violation. (Default: ValidateLogBad)

**IP6OPT_PADN**
Validate when ipv6 padn option data fields are non-zero. (Default: StripLog)

**IP6OPT_JUMBO**
Validate jumbogram packets. (Default: ValidateLog)

| | |
|---|---|
| **IP6OPT_RA** | Validate Router Alert packets. (Default: Ignore) |
| **IP6OPT_HA** | Validate Home Address option packets. (Default: Ignore) |
| **IP6OPT_OTH** | Validate unknown option types. (Default: RFC2460Log) |
| **IP6_RH0** | Validate routing header type 0 option. (Default: RFC5095NoSupportLog) |
| **IP6_RH2** | Validate routing header type 2 option. (Default: RFC2460NoSupportLog) |
| **IP6_RHOther** | Validate routing header other than type 0 or 2 option. (Default: RFC2460NoSupportLog) |
| **IP6OnLocalUnrecognizedHdr** | How to handle packets destined to the SGW with unrecognized IPV6 headers. (Default: DropLog) |
| **LogCheckSumErrors** | Log IP packets with bad checksums. (Default: Yes) |
| **LogNonIPv4IPv6** | Log occurrences of non-IPv4/IPv6 packets. (Default: Yes) |
| **LogReceivedTTL0** | Log received packets with TTL=0; this should never happen! (Default: Yes) |
| **LogOnForwardTTL0** | Log any attempts of forwarding IPv4 packets with TTL=0 destined for outside the firewall; this should never happen! (Default: DropLog) |
| **Log0000Src** | Log invalid 0.0.0.0 source address. (Default: Drop) |
| **Block0Net** | Block 0.* source addresses. (Default: DropLog) |
| **Block127Net** | Block 127.* source addresses. (Default: DropLog) |
| **BlockMulticastSrc** | Block multicast source addresses (224.0.0.0--239.255.255.255). (Default: DropLog) |
| **TTLMin** | The minimum IP Time-To-Live value accepted on receipt. (Default: 3) |
| **TTLOnLow** | What action to take on too low unicast TTL values. (Default: DropLog) |
| **TTLMinMulticast** | The minimum IP multicast Time-To-Live value accepted on receipt. (Default: 3) |
| **TTLOnLowMulticast** | What action to take on too low multicast TTL values. (Default: DropLog) |
| **DefaultTTL** | The default IP Time-To-Live of packets originated by the firewall (32-255). (Default: 255) |
| **LayerSizeConsistency** | TCP/UDP/ICMP/etc layer data and header sizes matching lower layer size information. (Default: ValidateLogBad) |
| **SecuRemoteUDPEncapCompat** | Allow IP data to contain eight bytes more than the UDP total length field specifies -- Checkpoint |

|  | SecuRemote violates NAT-T drafts. (Default: No) |
|---|---|
| **IPOptionSizes** | Validity of IP header option sizes. (Default: ValidateLogBad) |
| **IPOPT_SR** | How to handle IP packets with contained source or return routes. (Default: DropLog) |
| **IPOPT_TS** | How to handle IP packets with contained Timestamps. (Default: DropLog) |
| **IPOPT_RTRALT** | How to handle IP packets with contained route alert. (Default: ValidateLogBad) |
| **IPOPT_OTHER** | How to handle IP options not specified above. (Default: DropLog) |
| **DirectedBroadcasts** | How to handle directed broadcasts being passed from one interface to another. (Default: DropLog) |
| **TransparentBroadcastNAT** | How to handle Broadcast packets matching a NAT rule in Transparent mode. (Default: DropLog) |
| **IPRF** | How to handle the IP Reserved Flag, if set; it should never be. (Default: DropLog) |
| **StripDFOnSmall** | Strip the "DontFragment" flag for packets of this size or smaller. (Default: 65535) |
| **MulticastIPEnetOnMismatch** | What action to take when ethernet and IP multicast addresses do not match. (Default: DropLog) |

> ### Note
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.74. L2TPClient

### Description

A PPTP/L2TP client interface is a PPP (Point-to-Point Protocol) tunnel over an existing IP network. Its IP address and DNS servers are dynamically assigned.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **IP** | The host name to store the assigned IP address in, if this network object exists and have a value other then 0.0.0.0 the PPTP/L2TP client will try to get that one from the PPTP/L2TP server as preferred IP. (Optional) |
| **Network** | The network from which traffic should be routed into the tunnel. |
| **RemoteEndpoint** | The IP address of the L2TP/PPTP server. |
| **TunnelProtocol** | Specifies if PPTP or L2TP should be used for this tunnel. (Default: PPTP) |
| **OriginatorIPType** | Specifies what IP address to use as source IP in e.g. NAT. (Default: LocalInterface) |
| **OriginatorIP** | Manually specified originator IP address to use as source IP in e.g. NAT. |
| **DNS1** | IP of the primary DNS server. (Optional) |
| **DNS2** | IP of the secondary DNS server. (Optional) |
| **Username** | Specifies the username to use for this PPTP/L2TP interface. |
| **Password** | The password to use for this PPTP/L2TP interface. |
| **PPPAuthNoAuth** | Allow no authentication for this tunnel. (Default: No) |
| **PPPAuthPAP** | Use PAP authentication protocol for this tunnel. User name and password are sent in plaintext. (Default: Yes) |
| **PPPAuthCHAP** | Use CHAP authentication protocol for this tunnel. (Default: Yes) |
| **PPPAuthMSCHAP** | Use MS-CHAP authentication protocol for this tunnel. (Default: Yes) |
| **PPPAuthMSCHAPv2** | Use MS-CHAP v2 authentication protocol for this tunnel. (Default: Yes) |
| **MPPENone** | Allow authentication without Microsoft Point-to-Point Encryption (MPPE). (Default: Yes) |

**MPPERC440**                            Use an RC4 40 bit MPPE session key with MS-CHAP or MS-CHAP v2 authentication protocol. (Default: Yes)

**MPPERC456**                            Use an RC4 56 bit MPPE session key with MS-CHAP or MS-CHAP v2 authentication protocol. (Default: Yes)

**MPPERC4128**                           Use an RC4 128 bit MPPE session key with MS-CHAP or MS-CHAP v2 authentication protocol. (Default: Yes)

**DialOnDemand**                         Enable Dial-on-demand which means that the L2TP/PPTP tunnel will not be setup until traffic is sent on the interface. (Default: No)

**ActivitySensing**                      Specifies if the dial-on-demand should trigger on inbound or outbound traffic or both. (Default: BiDirectional)

**IdleTimeout**                          Idle timeout in seconds for dial-on-demand. (Default: 3600)

**Metric**                               Specifies the metric for the auto-created route. (Default: 90)

**MTU**                                  Specifies the size (in bytes) of the largest packet that can be passed onward. (Default: 1456)

**AutoInterfaceNetworkRoute**            Automatically add a route for this interface using the given remote network. (Default: Yes)

**MPPEAllowStateful**                    Allow usage of Stateful MPPE (less secure, use only for compatibility). (Default: No)

**IPsecInterface**                       Use this IPsec interface to encypt the traffic to the L2TP server. (L2TP/IPsec). (Optional)

**SNMPIndex**                            Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0)

**MemberOfRoutingTable**                 All or Specific. (Default: All)

**RoutingTable**                         Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main)

**Comments**                             Text describing the current object. (Optional)

# 3.75. L2TPServer

### Description

A PPTP/L2TP server interface terminates PPP (Point to Point Protocol) tunnels set up over existing IP networks.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **IP** | The IP address of the PPTP/L2TP server interface. |
| **TunnelProtocol** | Specifies if PPTP or L2TP should be used for this tunnel. (Default: PPTP) |
| **Interface** | The interface that the PPTP/L2TP Server should be listening on. |
| **ServerIP** | Specifies the IP that the PPTP/L2TP server should listen on, this can be an IP of a interface, or for example an ARP published IP. |
| **UseUserAuth** | Enable the use of user authentication rules on this server. (Default: Yes) |
| **MPPENone** | Allow no authentication for this tunnel. (Default: Yes) |
| **MPPERC440** | Use an RC4 40 bit MPPE session key with MS-CHAP or MS-CHAP v2 authentication protocol. (Default: Yes) |
| **MPPERC456** | Use an RC4 56 bit MPPE session key with MS-CHAP or MS-CHAP v2 authentication protocol. (Default: Yes) |
| **MPPERC4128** | Use an RC4 128 bit MPPE session key with MS-CHAP or MS-CHAP v2 authentication protocol. (Default: Yes) |
| **IPPool** | A range, group or network that the PPTP/L2TP server will use as IP address pool to give out IP addresses to the clients from. |
| **DNS1** | IP of the primary DNS server. (Optional) |
| **DNS2** | IP of the secondary DNS server. (Optional) |
| **NBNS1** | IP of the primary Windows Internet Name Service (WINS) server that is used in Microsoft environments which uses the NetBIOS Name Servers (NBNS) to assign IP addresses to NetBIOS names. (Optional) |
| **NBNS2** | IP of the primary Windows Internet Name Service (WINS) server that is used in Microsoft environments which uses the NetBIOS Name |

|  |  |
|---|---|
|  | Servers (NBNS) to assign IP addresses to NetBIOS names. (Optional) |
| **AllowedRoutes** | Restricts networks for which routes may automatically be added. (Default: all-nets) |
| **MPPEAllowStateful** | Allow usage of Stateful MPPE (less secure, use only for compatibility). (Default: No) |
| **SNMPIndex** | Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0) |
| **MemberOfRoutingTable** | All or Specific. (Default: All) |
| **RoutingTable** | Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main) |
| **ProxyARPAllInterfaces** | Always select all interfaces, including new ones, for publishing routes via Proxy ARP. (Default: No) |
| **ProxyARPInterfaces** | Specifies the interfaces on which the firewall should publish routes via Proxy ARP. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

# 3.76. L2TPServerSettings

**Description**

PPTP/L2TP server settings.

**Properties**

**L2TPBeforeRules**                     Pass L2TP connections sent to the firewall directly
                                        to the L2TP engine without consulting the ruleset.
                                        (Default: Yes)

**PPTPBeforeRules**                     Pass PPTP connections sent to the firewall directly
                                        to the PPTP engine without consulting the ruleset.
                                        (Default: Yes)

> ***Note***
> *This object type does not have an identifier and is identified by the name of the type
> only. There can only be one instance of this type.*

# 3.77. L2TPv3Client

### Description

A L2TPv3 client interface terminates L2 (Ethernet and VLAN) tunnels set up over existing IP networks.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **IP** | The IP address of the L2TPv3 Client interface. |
| **LocalNetwork** | The network on "this side" of the L2TPv3 tunnel. |
| **PseudowireType** | Specifies if L2TPv3 should tunnel Ethernet or IEEE 802.1Q (VLAN) tagged Ethernet frames. (Default: Ethernet) |
| **Protocol** | Specifies if L2TPv3 should tunnel over IP or UDP. (Default: UDP) |
| **RemoteEndpoint** | The IP address of the L2TPv3 server. |
| **OriginatorIPType** | Specifies what IP address to use as source IP in e.g. NAT. (Default: LocalInterface) |
| **OriginatorIP** | Manually specified originator IP address to use as source IP in e.g. NAT. |
| **IPsecInterface** | Use this IPsec interface to encypt the traffic to the L2TPv3 server. (L2TP/IPsec). (Optional) |
| **AutoRouteMetric** | Specifies the metric for the auto-created route used by the L2TPv3 Client. (Default: 100) |
| **HostName** | The host name for this L2TPv3 Client. (Used in the Host Name AVP). (Optional) |
| **RouterID** | Router ID. (Used in the Router ID AVP). (Optional) |
| **DHCPPassthrough** | Allow DHCP to pass through transparently. (Default: No) |
| **NonIPPassthrough** | Allow non-IP protocols to pass through transparently. (Default: No) |
| **SNMPIndex** | Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0) |
| **MemberOfRoutingTable** | All or Specific. (Default: All) |
| **RoutingTable** | Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main) |
| **ProxyARPAllInterfaces** | Always select all interfaces, including new ones, for |

| | |
|---|---|
| | publishing routes via Proxy ARP. (Default: No) |
| **ProxyARPInterfaces** | Specifies the interfaces on which the firewall should publish routes via Proxy ARP. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

# 3.78. L2TPv3Server

## Description

A L2TPv3 server interface terminates L2 (Ethernet and VLAN) tunnels set up over existing IP networks.

## Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **IP** | The IP address of the L2TPv3 Server interface. |
| **LocalNetwork** | The network on "this side" of the L2TPv3 tunnel. |
| **Protocol** | Specifies if L2TPv3 should tunnel over IP or UDP. (Default: UDP) |
| **Interface** | The interface that the L2TPv3 Server should be listening on. |
| **ServerIP** | Specifies the IP that the L2TPv3 Server should listen on, this can be an IP of a interface, or for example an ARP published IP. |
| **AutoRouteMetric** | Specifies the metric for the auto-created route used by the L2TPv3 Server. (Default: 100) |
| **HostName** | The host name for this L2TPv3 Server. (Used in the Host Name AVP). (Optional) |
| **RouterID** | Router ID. (Used in the Router ID AVP). (Optional) |
| **DHCPPassthrough** | Allow DHCP to pass through transparently. (Default: No) |
| **NonIPPassthrough** | Allow non-IP protocols to pass through transparently. (Default: No) |
| **SNMPIndex** | Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0) |
| **MemberOfRoutingTable** | All or Specific. (Default: All) |
| **RoutingTable** | Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main) |
| **ProxyARPAllInterfaces** | Always select all interfaces, including new ones, for publishing routes via Proxy ARP. (Default: No) |
| **ProxyARPInterfaces** | Specifies the interfaces on which the firewall should publish routes via Proxy ARP. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

# 3.79. LANtoLANVPN

## Description

This type lets you setup an IPsec tunnel between gateways in an easy way with algorithms that are known to be secure. (IKEv2 tunnel with AES-128 and SHA-256. DH group 14 (2048-bits) and forward secrecy.)

## Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **LocalNetwork** | The network on "this side" of the IPsec tunnel. The IPsec tunnel will be established between this network and the remote network. |
| **RemoteNetwork** | The network connected to the remote gateway. The IPsec tunnel will be established between the local network and this network. |
| **AuthMethod** | Certificate or Pre-shared key. (Default: PSK) |
| **PSK** | Selects the Pre-shared key to use with this IPsec Tunnel. |
| **GatewayCertificate** | Selects the certificate the firewall uses to authenticate itself to the other IPsec peer. |
| **RootCertificates** | Selects one or more root certificates to use with this IPsec Tunnel. |
| **RemoteEndpoint** | Specifies the IP address of the remote endpoint. This is the address the firewall will establish the IPsec tunnel to. It also dictates from where inbound IPsec tunnels are allowed. (Optional) |
| **AutoInterfaceNetworkRoute** | Automatically add a route for this interface using the given remote network. (Default: Yes) |
| **SNMPIndex** | Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0) |
| **MemberOfRoutingTable** | All or Specific. (Default: All) |
| **RoutingTable** | Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

# 3.80. LDAPDatabase

**Description**

External LDAP server used to verify user names and passwords.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the server. (Identifier) |
| **IP** | The IP address of the server. |
| **Port** | The TCP port of the server. (Default: 389) |
| **SourceIPSelection** | Which IP should be used as a source IP. (Default: Automatic) |
| **SourceIP** | The IP address to be used as source IP. |
| **Timeout** | The timeout, in milliseconds, used when processing requests. (Default: 5) |
| **NameAttr** | Specifies a name attribute in LDAP database. (Default: uid) |
| **PassAttr** | Specifies a password attribute in LDAP database. (Optional) |
| **GroupsAttr** | Specifies the group membership attribute used in the LDAP database. (Default: memberOf) |
| **GetGroups** | Retrieve group membership for users. (Default: Yes) |
| **DomainName** | The domain name of the server. (Optional) |
| **CombinedUsername** | Combine Name Attribute with given username, Optional Attribute and Base Object in LDAP bind request. (Default: No) |
| **OptionalAttribute** | Optional attribute to be used in bind request together with given username and Base Object. (Optional) |
| **BaseObject** | Specifies a base object to search. (Optional) |
| **UserName** | Specifies a user name. (Optional) |
| **Password** | Specifies a user password. (Optional) |
| **Type** | Add domain name to username. (Default: 0) |
| **RoutingTable** | Specifies the routing table the clients host route should be added to. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

# 3.81. LDAPServer

## Description

An LDAP server is used as a central repository of certificates and CRLs that the firewall can download when necessary.

### Properties

**Host**                            Specifies the IP address or hostname of the LDAP server.

**Username**                        Specifies the username to use when accessing the LDAP server. (Optional)

**Password**                        Specifies the password to use when accessing the LDAP server. (Optional)

**Port**                            Specifies the LDAP service port number. (Default: 389)

**Comments**                        Text describing the current object. (Optional)

> ***Note***
> *If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.82. LengthLimSettings

**Description**

Length limitations for various protocols.

**Properties**

| | |
|---|---|
| **MaxTCPLen** | TCP; Sometimes has to be increased if tunneling protocols are used. (Default: 1480) |
| **MaxUDPLen** | UDP; Many interactive applications use large UDP packets, may otherwise be decreased to 1480. (Default: 60000) |
| **MaxICMPLen** | ICMP; May be decreased to 1480 if desired. (Default: 10000) |
| **MaxICMPv6Len** | ICMPv6; May be decreased to 1280 if desired. (Default: 10000) |
| **MaxGRELen** | Encapsulated (tunneled transport), used by PPTP. (Default: 2000) |
| **MaxESPLen** | IPsec ESP; Encrypted communication. (Default: 2000) |
| **MaxAHLen** | IPsec AH; Authenticated communication. (Default: 2000) |
| **MaxSKIPLen** | SKIP; Simple Key management for IP, VPN protocol. (Default: 2000) |
| **MaxOSPFLen** | OSPF; Open Shortest Path First, routing protocol. (Default: 1480) |
| **MaxIPIPLen** | IPIP/FWZ; Encapsulated (tunneled) transport, used by VPN-1. (Default: 2000) |
| **MaxIPCompLen** | IPsec IPComp; Compressed communication. (Default: 2000) |
| **MaxL2TPLen** | L2TP; Layer 2 Tunneling Protocol. (Default: 2000) |
| **MaxOtherSubIPLen** | Others; sometimes has to be increased if unknown tunneling protocols are used. (Default: 1480) |
| **LogOversizedPackets** | Log occurrences of oversized packets. (Default: Yes) |

> **Note**
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.83. LinkAggregation

### Description

A Link Aggregation interface combines multiple Ethernet interfaces into a single logical endpoint.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **Members** | A set of Ethernet interfaces to aggregate. (Optional) |
| **DistributionAlgorithm** | Specifies how outgoing traffic will be distributed among the active links. (Default: Combination) |
| **Mode** | Specifies the method used to aggregate links. (Default: Static) |
| **LACPActivity** | Specifies if the system should actively attempt to initiate LACP negotiations or wait for a partner system to do so. (Default: Active) |
| **LACPTimeout** | Specifies how soon the system will reselect active links if a link is broken. (Default: Long) |
| **LACPSystemPriority** | System priority value to be sent in LACP messages. (Default: 1) |
| **MACAddress** | The hardware address for the interface. (Optional) |
| **IP** | The IP address of the interface. |
| **Network** | The network of the interface. |
| **DefaultGateway** | The default gateway of the interface. (Optional) |
| **Broadcast** | The broadcast address of the connected network. (Optional) |
| **EnableIPv6** | Enable processing of IPv6 traffic on this interface. (Default: No) |
| **IPv6IP** | The IP address of the interface. |
| **IPv6Network** | The network of the interface. |
| **IPv6DefaultGateway** | The default gateway of the interface. (Optional) |
| **RouterDiscovery** | Uses Router information (ND RA) from local network to auto-configure Network and Default Gateway addresses. (Default: No) |
| **AutoIPv6IP** | Automatically configures IP Address using Network Address and EUI-64. (Default: No) |
| **DHCPv6Enabled** | Enable DHCPv6 client on this interface. (Default: No) |

| | |
|---|---|
| **PrivateIP** | The private IP address of this high availability node. (Optional) |
| **PrivateIP6** | The private IP6 address of this high availability node. (Default: localhost6) |
| **NOCHB** | This will disable sending Cluster Heartbeats from this interface (used by HA to detect if a node is online and working). (Optional) |
| **MTU** | Specifies the size (in bytes) of the largest packet that can be passed onward. Must be 1294 or larger when IPv6 is enabled. (Default: 1500) |
| **Metric** | Specifies the metric for the auto-created route. (Default: 100) |
| **DHCPEnabled** | Enable DHCP client on this interface. (Default: No) |
| **DHCPHostName** | Optional DHCP Host Name. Leave blank to use default name. (Optional) |
| **AutoSwitchRoute** | Allows traffic to be forwarded transparently across all interfaces with Transparent Mode enabled that belong to the same routing table. (Default: No) |
| **DHCPPassthrough** | Allow DHCP to pass through transparently. (Default: No) |
| **NonIPPassthrough** | Allow non-IP protocols to pass through transparently. (Default: No) |
| **BroadcastFwd** | By default, this traffic is dropped. (Default: No) |
| **AutoInterfaceNetworkRoute** | Automatically add a route for this interface using the given network. (Default: Yes) |
| **AutoDefaultGatewayRoute** | Automatically add a default route for this interface using the given default gateway. (Default: Yes) |
| **DHCPDNS1** | IP of the primary DNS server. (Optional) |
| **DHCPDNS2** | IP of the secondary DNS server. (Optional) |
| **DHCPv6DNS1** | IP of the primary IPv6 DNS server. (Optional) |
| **DHCPv6DNS2** | IP of the secondary IPv6 DNS server. (Optional) |
| **EnableRouterAdvertisement** | Enable Router Advertisement for this interface. (Default: No) |
| **SNMPIndex** | Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0) |
| **MemberOfRoutingTable** | All or Specific. (Default: All) |
| **RoutingTable** | Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main) |

**Comments**                                    Text describing the current object. (Optional)

# 3.84. LinkMonitor

**Description**

The Link Monitor allows the system to monitor one or more hosts and take action if they are unreachable.

**Properties**

| | |
|---|---|
| **Action** | Specifies what action the system should take. |
| **Addresses** | Specifies the addresses that should be monitored. |
| **MaxLoss** | A single host is considered unreachable if this number of consecutive ping responses to that host are not replied to. (Default: 7) |
| **PingInterval** | Milliseconds between each monitor attempt. (Default: 250) |
| **InitGracePeriod** | Do not allow triggering of the link monitor for this number of seconds after the last reconfiguration. (Default: 45) |
| **RoutingTable** | Routing table used for link monitoring. (Default: main) |
| **UseSharedIP** | Use the shared IP of an HA cluster instead of the private IP of the node. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.

# 3.85. LocalReassSettings

**Description**

Parameters use for local fragment reassembly.

**Properties**

**LocalReass_MaxConcurrent**          Maximum number of concurrent local reassemblies. (Default: 256)

**LocalReass_MaxSize**          Maximum size of a locally reassembled packet. (Default: 10000)

**LocalReass_NumLarge**          Number of large (>2K) local reassembly buffers (of the above size). (Default: 32)

> **Note**
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.86. LocalUserDatabase

**Description**

A local user database contains user accounts used for authentication purposes.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. (Identifier) |
| **Comments** | Text describing the current object. (Optional) |

# 3.86.1. User

**Description**

User credentials may be used in User Authentication Rules, which in turn are used in e.g. PPP, IPsec XAuth, Web Authentication, etc

**Properties**

| | |
|---|---|
| **Name** | Specifies the username to add into the user database. (Identifier) |
| **Password** | The password for this user. |
| **HasStrongPassword** | Specifies if the user's password is strong. (Default: No) |
| **Groups** | Specifies the user groups that this user is a member of, e.g. Administrators. (Optional) |
| **IPPool** | If the user is logging in over PPTP/L2TP or IPsec it will be assigned this static IP. (Optional) |
| **AutoAddRouteNet** | PPTP/L2TP networks behind the user. (Optional) |
| **AutoAddRouteMetric** | Metric for the network. (Optional) |
| **SSHKeys** | Public keys used to log in via SSH. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

# 3.87. LogReceiverMemory

**Description**

A memory log receiver is used to receive and keep log events in system RAM.

**Properties**

**Name**                                Specifies a symbolic name for the log receiver. (Identifier)

**LogSeverity**                         Specifies with what severity log events will be sent to the specified log receivers. (Optional; Default: Emergency,Alert,Critical,Error,Warning,Notice,Info)

**Comments**                            Text describing the current object. (Optional)

# 3.87.1. LogReceiverMessageException

The definitions here are the same as in Section 3.44.1, "LogReceiverMessageException" .

# 3.88. LogReceiverSMTP

**Description**

Mail Alerting is used for sending important events via email.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the log receiver. (Identifier) |
| **IPAddress** | IP address or DNS name of an SMTP server that accepts emails for the given address(es). |
| **Port** | TCP port of the SMTP server. Changing it to 465 will NOT make the connection encrypted - it will simply not work. (Default: 25) |
| **Recipient** | Who to send email to. To send to multiple recipients, configure an alias (aka mailing list) on the server and send to that. |
| **Sender** | The sender email address to use for log event emails. |
| **Identity** | Customizes how the system identifies itself to the SMTP server when issuing the EHLO command. Preferrably, this should be the DNS name of the sending interface, as the server may be configured to require it. By default, the numeric IP address of the sending interface is used. (Optional) |
| **XMailer** | Specifies a custom X-Mailer email header string. The X-Mailer header field is typically used to identify the name and version number of the software that generated the email. (Optional) |
| **Subject** | The email Subject to use for log event emails. |
| **Activation** | Select how events trigger an alert. (Default: SingleEvent) |
| **EventCountThreshold** | How many events are required to trigger the alert?. (Default: 10) |
| **EventCountPeriod** | How far back in time to look when counting events. Events that were included in a previous email are not counted again. (Default: 60) |
| **KeepCollectingPeriod** | To provide context in the alert email, more events can be collected for a short time and included in the email. Set to 0 to not collect and send as soon as possible. (Default: 1) |
| **MinTimeBetweenEmail** | Emails will never be sent more often than this. Additional alerts will be sent in the next email. (Default: 30) |
| **SendReportEmails** | Periodically send report emails containing events |

|  |  |
|---|---|
|  | that did not trigger the rate threshold. The report will always be sent, even if nothing occured. (Default: No) |
| **ReportEmailInterval** | How often to send report emails. (Default: 24) |
| **ReportEmailSubject** | The email Subject to use for report emails. |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Optional; Default: Emergency,Alert,Critical,Error,Warning,Notice,Info) |
| **RoutingTable** | Specifies the routing table the clients host route should be added to. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

## 3.88.1. LogReceiverMessageException

The definitions here are the same as in Section 3.44.1, "LogReceiverMessageException" .

# 3.89. LogReceiverSyslog

**Description**

A Syslog receiver is used to receive log events from the system in the standard Syslog format.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the log receiver. (Identifier) |
| **IPAddress** | Specifies the IP address of the log receiver. |
| **Port** | Specifies the port number of the log service. (Default: 514) |
| **Facility** | Specifies what facility is used when logging. (Default: local0) |
| **Hostname** | Specifies a unique hostname. If not configured, the IP address of the sending interface will be sent as hostname. (Optional) |
| **RFC5424** | Send Syslog messages according to RFC5424. (Default: No) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Optional; Default: Emergency,Alert,Critical,Error,Warning,Notice,Info) |
| **RoutingTable** | Specifies the routing table the clients host route should be added to. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

# 3.89.1. LogReceiverMessageException

The definitions here are the same as in Section 3.44.1, "LogReceiverMessageException" .

# 3.90. LogSettings

**Description**

Advanced log settings.

**Properties**

**LogSendPerSecLimit**                        Limits how many log packets the firewall may send
                                              out per second. (Default: 2000)

> ### *Note*
> *This object type does not have an identifier and is identified by the name of the type
> only. There can only be one instance of this type.*

# 3.91. LoopbackInterface

## Description

Loopback interfaces will take all packets sent through them and pass them back up a different interface as newly received packets.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **LoopTo** | Loopback interface. (Optional) |
| **IP** | Interface address. |
| **Network** | The network of the interface. |
| **Broadcast** | The broadcast address of the connected network. (Optional) |
| **Metric** | Specifies the metric for the auto-created route. (Default: 100) |
| **AutoInterfaceNetworkRoute** | Automatically add a route for this virtual LAN interface using the given network. (Default: Yes) |
| **EnableIPv6** | Enable processing of IPv6 traffic on this interface. (Default: No) |
| **IPv6IP** | IPv6 Interface address. |
| **IPv6Network** | The network of the interface. |
| **SNMPIndex** | Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0) |
| **MemberOfRoutingTable** | All or Specific. (Default: All) |
| **RoutingTable** | Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

# 3.92. MiscSettings

**Description**

Miscellaneous Settings

**Properties**

| | |
|---|---|
| **UDPSrcPort0** | How to treat UDP packets with source port 0. (Default: DropLog) |
| **Port0** | How to treat TCP/UDP packets with destination port 0 and TCP packets with source port 0. (Default: DropLog) |
| **HighBuffers_Dynamic** | Allocate the HighBuffers value dynamically. (Default: Yes) |
| **HighBuffers** | Number of packet buffers to allocate in addition to the ~200 initial buffers. (Default: 1024) |
| **LocalUndelivered** | How to treat (allowed) packets to the firewall that do not match open ports (snmp, scp, netcon, etc). (Default: DropLog) |
| **WCFPerfLog** | Enables periodical logging of Web Contentent Filtering resolving performance. (Default: Disabled) |
| **AllowIPRules** | Allow using IPRules in addition to IPPolicies. (Default: Yes) |
| **EnablePollOffload** | Enable interface poll offloading. (Default: Yes) |
| **AVCache_Lifetime** | Number of minutes that an anti-virus cache entry remains in the cache (0=cache disabled). (Default: 20) |
| **EnforceStrongPasswords** | When enabled, newly created or modified user passwords must comply to predefined complexity rules. Passwords created before enabling this setting will NOT be verified. (Default: Yes) |
| **HTTPPipeliningMaxReq** | Maximum number of pipelined requests in HTTP. (Default: 64) |

**Note**

*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.93. MulticastPolicy

The definitions here are the same as in Section 3.68.3, "MulticastPolicy" .

# 3.94. MulticastSettings

**Description**

Advanced Multicast Settings.

**Properties**

| | |
|---|---|
| **AutoAddMulticastCoreRoute** | Auto generate core route for "224.0.0.1-239.255.255.255". (Default: Yes) |
| **IGMPBeforeRules** | Allows IGMP traffic to enter the firewall by default. (Default: Yes) |
| **IGMPMaxGlobalRequestsPerSecond** | Maximum number of requests per second. (Default: 1000) |
| **IGMPMaxRequestsPerSecond** | Maximum number of requests per interface per second. (Default: 100) |
| **IGMPReactToOwnQueries** | The firewall should always respond with Member Reports, even to Queries originating from itself. (Default: No) |
| **IGMPRobustnessVariable** | IGMP is robust to 'value' - 1 packet losses. (Default: 2) |
| **IGMPQueryInterval** | The interval (ms) between general queries sent by the firewall. (Default: 125000) |
| **IGMPQueryResponseInterval** | The maximum time (ms) until a host/client has to send an answer to a query. (Default: 10000) |
| **IGMPStartupQueryInterval** | The general query interval (ms) to use during the startup phase (default: 1/4 of the 'IGMP Query Interval' parameter. (Default: 30000) |
| **IGMPStartupQueryCount** | The number of startup queries to send during the startup phase. (Default: 2) |
| **IGMPLastMemberQueryInterval** | The maximum time (ms) until a host/client has to send an answer to a group and group-and-source specific query. (Default: 5000) |
| **IGMPUnsolicatedReportInterval** | The time between repetitions (ms) of an initial membership report. (Default: 1000) |
| **IGMPRouterVersion** | Multiple IGMP querying routers on a network must use the same IGMP version. (Default: IGMPv3) |
| **IGMPLowestCompatibleVersion** | Lowest IGMP compatibility mode. (Default: IGMPv1) |

> ***Note***
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.95. NATPool

**Description**

A NAT Pool is used for NATing multiple concurrent connections to using different source IP addresses.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the NAT Pool. (Identifier) |
| **Type** | Specifies how NAT'ed connections are assigned a NAT IP address. (Default: stateful) |
| **IPSource** | Specify which IP Address source to use. (Default: IPRange) |
| **IPPool** | Specifies the IP Pool used for retrieving IP addresses for NAT translation. |
| **IPPoolIPs** | The number of IP addresses to get from the IP Pool. |
| **IPRange** | Specifies the range of IP addresses used for NAT translation. |
| **StateKeepAlive** | The number of seconds that stateful NAT state will be kept in absence of new connections. (Default: 120) |
| **MaxStates** | Maximum number of statefully tracked NATPool states. (Default: 16384) |
| **ProxyARPAllInterfaces** | Always select all interfaces, including new ones, for publishing routes needed for receiving traffic on NATPool addresses. (Default: No) |
| **ProxyARPInterfaces** | Specifies the interface/interfaces on which the firewall should publish routes needed for the relay via Proxy ARP. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

# 3.96. OSPFProcess

### Description

An OSPF Router Process defines a group of routers exchanging routing information via the Open Shortest Path First routing protocol.

### Properties

**Name**

Specifies a symbolic name for the OSPF process. (Identifier)

**RouterID**

Specifies the IP address that is used to identify the router. If no router ID is configured, it will be computed automatically based on the highest IP address of any interface participating in the OSPF process. (Optional)

**PrivRouterID**

The private router ID of this high availability node. (Optional)

**RFC1583**

Enable this if the firewall will be used in a environment that consists of routers that only support RFC 1583. (Default: No)

**SPFHoldTime**

Specifies the minimum time, in seconds, between two SPF calculations. (Default: 10)

**SPFDelayTime**

Specifies the delay time, in seconds, between when OSPF receives a topology change and when it starts a SPF calculation. (Default: 5)

**LSAGroupPacing**

This specifies the time in seconds at which interval the OSPF LSAs are collected into a group and refreshed. (Default: 10)

**RoutesHoldtime**

This specifies the time in seconds that the routing table will be kept unchanged after a reconfiguration of OSPF entries or a HA failover. (Default: 45)

**RefBandwidthValue**

Set the reference bandwidth that is used when calculating the default interface cost for routes. (Default: 1)

**RefBandwidthUnit**

Sets the reference bandwidth unit. (Default: Gbps)

**MemoryMaxUsage**

Maximum amount in bytes of RAM that the OSPF process is allowed to use. The default is one percent of installed RAM. Specifying 0 indicates that the OSPF process is allowed to use all available RAM. (Optional)

**DebugPacket**

Enables or disabled logging of general packet parsing events and also specifies the details of the log. (Default: Off)

**DebugHello**

Enables or disabled logging of hello packets and also specifies the details of the log. (Default: Off)

**DebugDDesc**
Enables or disabled logging of database description packets and also specifies the details of the log. (Default: Off)

**DebugExchange**
Enables or disabled logging of exchange packets and also specifies the details of the log. (Default: Off)

**DebugLSA**
Enables or disabled logging of LSA events and also specifies the details of the log. (Default: Off)

**DebugSPF**
Enables or disabled logging of SPF calculation events and also specifies the details of the log. (Default: Off)

**DebugRoute**
Enables or disabled logging of routing table manipulation events and also specifies the details of the log. (Default: Off)

**AuthType**
Specifies the authentication type for the OSPF protocol exchanges. (Default: None)

**AuthPassphrase**
Specifies the passphrase used for authentication. (Optional)

**AuthMD5ID**
Specifies the MD5 key ID used for MD5 digest authentication.

**AuthMD5Key**
A 128-bit key used to produce the MD5 digest. (Optional)

**LogEnabled**
Enable logging. (Default: Yes)

**LogSeverity**
Specifies with what severity log events will be sent to the specified log receivers. (Default: Default)

**Comments**
Text describing the current object. (Optional)

## 3.96.1. OSPFArea

### Description

An OSPF area is a sub-domain within the OSPF process which collects OSPF interfaces, neighbors, aggregates and virtual links.

**Properties**

**Name**
Specifies a symbolic name for the area. (Identifier)

**AreaID**
Specifies the area id, if 0.0.0.0 is specified this is the backbone area.

**Stub**
Enable to make the router automatically advertises a default route so that routers in the stub area can reach destinations outside the area. (Default: No)

**StubSummarize**
Become a default router for stub area (Summarize). (Default: Yes)

| | |
|---|---|
| **StubMetric** | Route metric for stub area. (Optional) |
| **FilterExternal** | Specifies the network addresses allowed to be imported into this area from external routing sources. (Optional) |
| **FilterInterArea** | Specifies the network addresses allowed to be imported from other routers inside the area. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

## 3.96.1.1. OSPFInterface

### Description

Select and define the properties of an interface that should be made a member of the Router Process.

**Properties**

| | |
|---|---|
| **Interface** | Specifies which interface in the firewall will be used for this OSPF interface. (Identifier) |
| **Type** | Auto, Broadcast, Point-to-point or Point-to-multipoint. (Default: Auto) |
| **Network** | Specifies the network related to the configured OSPF interface. (Optional) |
| **MetricType** | Metric value or Bandwidth. (Default: MetricValue) |
| **Metric** | Specifies the routing metric for this OSPF interface. (Default: 10) |
| **BandwidthValue** | Specifies the bandwidth for this OSPF interface. |
| **BandwidthUnit** | Specifies the bandwidth unit. (Default: Mbps) |
| **UseDefaultAuth** | Use the authentication configuration specified in the OSPF process. (Default: Yes) |
| **AuthType** | Specifies the authentication type for the OSPF protocol exchanges. (Default: None) |
| **AuthPassphrase** | Specifies the passphrase used for authentication. (Optional) |
| **AuthMD5ID** | Specifies the MD5 key ID used for MD5 digest authentication. |
| **AuthMD5Key** | A 128-bit key used to produce the MD5 digest. (Optional) |
| **HelloInterval** | Specifies the number of seconds between HELLO packets sent from the interface. (Default: 10) |
| **RtrDeadInterval** | If no HELLO packets are received from a neighbor within this interval (in seconds), that neighbor |

| | |
|---|---|
| | router will be declared to be down. (Default: 40) |
| **RxmtInterval** | Specifies the number of seconds between retransmissions of LSAs to neighbors on this interface. (Default: 5) |
| **RtrPrio** | Specifies the router priority, a higher number increases this routers chance of becoming DR or BDR, if 0 is specified this router will not be eligible in the DR/BDR election. (Default: 1) |
| **InfTransDelay** | Specifies the estimated transmit delay for the interface in seconds. This value represents the maximum time it takes to forward a LSA packet trough the router. (Default: 1) |
| **WaitInterval** | Specifies the number of seconds between the time when the interface brought up and the election of the DR and BDR. This value should be higher than the hello interval. (Default: 40) |
| **Passive** | Enable to make it possible to include networks into the OSPF routing process, without running OSPF on the interface connected to that network. (Default: No) |
| **IgnoreMTU** | Enable to allow OSPF MTU mismatches. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

# 3.96.1.2. OSPFNeighbor

## Description

For point-to-point and point-to-multipoint networks, specify the IP addresses of directly connected routers.

## Properties

| | |
|---|---|
| **Interface** | Specifies the OSPF interface of the neighbor. |
| **IPAddress** | IP Address of the neighbor. |
| **Metric** | Specifies the metric of the neighbor. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

### Note
*If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.96.1.3. OSPFAggregate

## Description

An aggregate is used to replace any number of smaller networks belonging to the local (intra) area with one contiguous network which may then be advertised or hidden.

**Properties**

**Network**                        The aggregate network used to combine several small routes.

**Advertise**                      Advertise the aggregate. (Default: Yes)

**Comments**                       Text describing the current object. (Optional)

> ### *Note*
> *If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.96.1.4. OSPFVLink

## Description

An area that does not have a direct connection to the backbone must have at least one area border router with a virtual link to a backbone router, or to another router with a link to the backbone.

**Properties**

**Name**                           Specifies a symbolic name for the virtual link. (Identifier)

**RouterID**                       The ID of the router on the other side of the virtual link.

**UseDefaultAuth**                 Use the authentication configuration specified in the OSPF process. (Default: Yes)

**AuthType**                       Specifies the authentication type for the OSPF protocol exchanges. (Default: None)

**AuthPassphrase**                 Specifies the passphrase used for authentication. (Optional)

**AuthMD5ID**                      Specifies the MD5 key ID used for MD5 digest authentication.

**AuthMD5Key**                     A 128-bit key used to produce the MD5 digest. (Optional)

**Comments**                       Text describing the current object. (Optional)

# 3.97. Pipe

## Description

A pipe defines basic traffic shaping parameters. The pipe rules then determines which traffic goes through which pipes.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the pipe. (Identifier) |
| **LimitKbpsTotal** | Total bandwidth limit for this pipe in kilobits per second. (Optional) |
| **LimitPPSTotal** | Total packet per second limit for this pipe. (Optional) |
| **LimitKbps0** | Specifies the bandwidth limit in kbps for precedence 0 (the lowest precedence). (Optional) |
| **LimitPPS0** | Specifies the packet per second limit for precedence 0 (the lowest precedence). (Optional) |
| **LimitKbps1** | Specifies the bandwidth limit in kbps for precedence 1. (Optional) |
| **LimitPPS1** | Specifies the packet per second limit for precedence 1. (Optional) |
| **LimitKbps2** | Specifies the bandwidth limit in kbps for precedence 2. (Optional) |
| **LimitPPS2** | Specifies the packet per second limit for precedence 2. (Optional) |
| **LimitKbps3** | Specifies the bandwidth limit in kbps for precedence 3. (Optional) |
| **LimitPPS3** | Specifies the packet per second limit for precedence 3. (Optional) |
| **LimitKbps4** | Specifies the bandwidth limit in kbps for precedence 4. (Optional) |
| **LimitPPS4** | Specifies the packet per second limit for precedence 4. (Optional) |
| **LimitKbps5** | Specifies the bandwidth limit in kbps for precedence 5. (Optional) |
| **LimitPPS5** | Specifies the packet per second limit for precedence 5. (Optional) |
| **LimitKbps6** | Specifies the bandwidth limit in kbps for precedence 6. (Optional) |
| **LimitPPS6** | Specifies the packet per second limit for precedence 6. (Optional) |
| **LimitKbps7** | Specifies the bandwidth limit in kbps for |

| | precedence 7 (the highest precedence). (Optional) |
|---|---|
| **LimitPPS7** | Specifies the packet per second limit for precedence 7 (the highest precedence). (Optional) |
| **UserLimitKbpsTotal** | Total bandwidth limit per group in the pipe in kilobits per second. (Optional) |
| **UserLimitPPSTotal** | Total throughput limit per group in the pipe in packets per second. (Optional) |
| **UserLimitKbps0** | Specifies the bandwidth limit per group in kbps for precedence 0 (the lowest precedence). (Optional) |
| **UserLimitPPS0** | Specifies the throughput limit per group in PPS for precedence 0 (the lowest precedence). (Optional) |
| **UserLimitKbps1** | Specifies the bandwidth limit per group in kbps for precedence 1. (Optional) |
| **UserLimitPPS1** | Specifies the throughput limit per group in PPS for precedence 1. (Optional) |
| **UserLimitKbps2** | Specifies the bandwidth limit per group in kbps for precedence 2. (Optional) |
| **UserLimitPPS2** | Specifies the throughput limit per group in PPS for precedence 2. (Optional) |
| **UserLimitKbps3** | Specifies the bandwidth limit per group in kbps for precedence 3. (Optional) |
| **UserLimitPPS3** | Specifies the throughput limit per group in PPS for precedence 3. (Optional) |
| **UserLimitKbps4** | Specifies the bandwidth limit per group in kbps for precedence 4. (Optional) |
| **UserLimitPPS4** | Specifies the throughput limit per group in PPS for precedence 4. (Optional) |
| **UserLimitKbps5** | Specifies the bandwidth limit per group in kbps for precedence 5. (Optional) |
| **UserLimitPPS5** | Specifies the throughput limit per group in PPS for precedence 5. (Optional) |
| **UserLimitKbps6** | Specifies the bandwidth limit per group in kbps for precedence 6. (Optional) |
| **UserLimitPPS6** | Specifies the throughput limit per group in PPS for precedence 6. (Optional) |
| **UserLimitKbps7** | Specifies the bandwidth limit per group in kbps for precedence 7 (the highest precedence). (Optional) |
| **UserLimitPPS7** | Specifies the throughput limit per group in PPS for precedence 7 (the highest precedence). (Optional) |
| **Grouping** | Grouping enables per-port/IP/network static bandwidth limits as well as dynamic balancing between groups. (Default: None) |

**GroupingNetworkSize**    If users are grouped according to source or destination network, the size of the network has to be specified by this setting. (Default: 0)

**Dynamic**    Enable dynamic balancing of groups. (Default: No)

**PrecedenceMin**    Specifies the lowest allowed precedence for traffic in this pipe. If a packet with a lower precedence enters, its precedence is raised to this value. (Default: 0)

**PrecedenceDefault**    Specifies the default precedence for the pipe. If a packet enters this pipe without a set precedence, it gets assigned this value. Should be higher than or equal to the minimum precedence. (Default: 0)

**PrecedenceMax**    Specifies the highest allowed precedence for traffic in this pipe. If a packet with a higher precedence enters, its precedence is lowered to this value. Should be higher than or equal to the default precedence. (Default: 7)

**Comments**    Text describing the current object. (Optional)

# 3.98. PipeRule

### Description

A Pipe Rule determines traffic shaping policy - which Pipes to use - for one or more types of traffic with the same granularity as the standard ruleset.

### Properties

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the object. (Optional) |
| **SourceInterface** | Specifies the name of the receiving interface to be compared to the received packet. |
| **SourceNetwork** | Specifies the sender span of IP addresses to be compared to the received packet. |
| **DestinationInterface** | Specifies the destination interface to be compared to the received packet. |
| **DestinationNetwork** | Specifies the span of IP addresses to be compared to the destination IP of the received packet. |
| **Service** | Specifies a service that will be used as a filter parameter when matching traffic with this rule. |
| **Schedule** | By adding a schedule to a rule, the firewall will only allow that rule to trigger at those designated times. (Optional) |
| **ForwardChain** | Specifies one or more pipes to be used for forward traffic. (Optional) |
| **ReturnChain** | Specifies one or more pipes to be used for return traffic. (Optional) |
| **Precedence** | Specifies what precedence should be assigned to the packets before sent into a pipe. (Default: FromPipe) |
| **FixedPrecedence** | Specifies the fixed precedence. |
| **Comments** | Text describing the current object. (Optional) |

> ### *Note*
> *If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.99. PPPoETunnel

**Description**

A PPPoE interface is a PPP (point-to-point protocol) tunnel over an existing physical Ethernet interface. Its IP address is dynamically assigned.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **EthernetInterface** | The physical Ethernet interface that connects to the PPPoE server network. |
| **IP** | The host name to store the assigned IP address in. |
| **Network** | The network from which traffic should be routed into the tunnel. |
| **DNS1** | IP of the primary DNS server. (Optional) |
| **DNS2** | IP of the secondary DNS server. (Optional) |
| **Username** | Specifies the username to use for this PPPoE tunnel. |
| **Password** | The password to use for this PPPoE tunnel. |
| **ServiceName** | Specifies the PPPoE server service name used to distinguish between two or more PPPoE servers attached to the same network. (Optional) |
| **PPPAuthNoAuth** | Allow no authentication for this tunnel. (Default: No) |
| **PPPAuthPAP** | Use PAP authentication protocol for this tunnel. User name and password are sent in plaintext. (Default: Yes) |
| **PPPAuthCHAP** | Use CHAP authentication protocol for this tunnel. (Default: Yes) |
| **PPPAuthMSCHAP** | Use MS-CHAP authentication protocol for this tunnel. (Default: Yes) |
| **PPPAuthMSCHAPv2** | Use MS-CHAP v2 authentication protocol for this tunnel. (Default: Yes) |
| **DialOnDemand** | Enable Dial-on-demand which means that the PPPoE tunnel will not be setup until traffic is sent on the interface. (Default: No) |
| **ActivitySensing** | Specifies if the dial-on-demand should trigger on inbound or outbound traffic or both. (Default: BiDirectional) |
| **IdleTimeout** | Idle timeout in seconds for dial-on-demand. (Default: 3600) |

**Metric**                              Specifies the metric for the auto-created route. (Default: 90)

**AutoInterfaceNetworkRoute**           Automatically add a route for this interface using the given remote network. (Default: Yes)

**Schedule**                            The schedule defines when the PPPoE tunnel should be active. (Optional)

**ForceUnnumbered**                     Force the PPPoE tunnel to be unnumbered. (Default: No)

**SpecifyManually**                     Make it possible to manually specify IP Address object. (Default: No)

**MTU**                                 Specifies the size (in bytes) of the largest packet that can be passed onward. (Default: 1492)

**SNMPIndex**                           Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0)

**MemberOfRoutingTable**                All or Specific. (Default: All)

**RoutingTable**                        Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main)

**Comments**                            Text describing the current object. (Optional)

# 3.100. PPPSettings

**Description**

Settings related to the PPP protocol.

**Properties**

**InitialResendTime**                    Initial time in milliseconds to wait before sending a
                                         new configuration request if no server response is
                                         received. (Default: 200)

> ***Note***
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.101. PSK

### Description

PSK (Pre-Shared Key) authentication is based on a shared secret that is known only by the parties involved.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the pre-shared key. (Identifier) |
| **Type** | Specifies the type of the shared key. |
| **PSKAscii** | Specifies the PSK as a passphrase. |
| **PSKHex** | Specifies the PSK as a hexadecimal key. |
| **Comments** | Text describing the current object. (Optional) |

# 3.102. RadiusAccounting

**Description**

External RADIUS server used to collect user statistics.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the server. (Identifier) |
| **IPAddress** | The IP address of the server. |
| **Port** | The UDP port of the server. (Default: 1813) |
| **RetryTimeout** | The retry timeout, in seconds, used when trying to contact the RADIUS accounting server. If no response has been given after for example 2 seconds, the firewall will try again by sending a new AccountingRequest packet. (Default: 2) |
| **SharedSecret** | The shared secret phrase for the Authenticator generation. |
| **SourceIPSelection** | Which IP should be used as a source IP. (Default: Automatic) |
| **SourceIP** | The IP address to be used as source IP. |
| **RoutingTable** | Specifies the routing table the clients host route should be added to. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

# 3.103. RadiusRelay

**Description**

RADIUS relay for intercepting packets from a user endpoint and sending packets to a remote RADIUS server.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the relayer. (Identifier) |
| **SourceInterface** | Specifies the name of the receive interface for RADIUS relay requests. |
| **ClientIPFilter** | Specifies the network that the AP belongs to. |
| **ListeningIP** | Specifies the local IP address on which the system receives Access Point requests. This parameter is optional and will use IP of source interface, if not set. (Optional) |
| **ListeningPort** | Specifies the listening port on which the system receives Access Point requests. (Default: 1812) |
| **RemoteServerIP** | Specifies the IP address of the remote RADIUS server. |
| **RemoteServerPort** | Specifies the port of the remote RADIUS server. (Default: 1812) |
| **SendingIP** | Specifies the local IP address from which the system sends requests to the remote Remote RADIUS server. This parameter is optional and will use IP of routed destination interface, if not set. (Optional) |
| **IdleTimeout** | A successfully authenticated user will be logged out automatically after this many seconds, if no traffic has been received from the user's IP address. (Default: 1800) |
| **SessionTimeout** | A successfully authenticated user will be logged out automatically after this many seconds, even if traffic has been received from the user's IP address. (Default: 0) |
| **UseServerTimeouts** | Use timeouts received from the authentication server. If no values are received, the manually specified values will be used. (Default: No) |
| **DHCPServer** | Specifies the DHCP server rule that is responsible for distributing leases for authenticated users. |
| **OverrideUserDataInterface** | Optionally specify the source interface for the authenticated user data. If not specified, the configured RADIUS Relay source interface will be used. (Optional) |

**LogEnabled**                    Enable logging. (Default: Yes)

**LogSeverity**                   Specifies with what severity log events will be sent to the specified log receivers. (Default: Default)

**RoutingTable**                  Specifies the routing table the clients host route should be added to. (Default: main)

**Comments**                      Text describing the current object. (Optional)

# 3.104. RadiusServer

**Description**

External RADIUS server used to verify user names and passwords.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the server. (Identifier) |
| **IPAddress** | The IP address of the server. |
| **Port** | The UDP port of the server. (Default: 1812) |
| **RetryTimeout** | The retry timeout, in seconds, used when trying to contact the RADIUS server. If no response has been given after for example 2 seconds, the firewall will try again by sending a new Access-Request packet. (Default: 2) |
| **SharedSecret** | The shared secret phrase for the Authenticator generation. |
| **SourceIPSelection** | Which IP should be used as a source IP. (Default: Automatic) |
| **SourceIP** | The IP address to be used as source IP. |
| **RoutingTable** | Specifies the routing table the clients host route should be added to. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

# 3.105. RealTimeMonitorAlert

## Description

Monitors a statistical value. Log messages are generated if the value goes below the lower threshold or above the high threshold.

## Properties

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Monitor** | Statistical value. |
| **SampleTime** | Interval in seconds between checking the statistic. (Optional) |
| **LowThreshold** | Log if statistical value goes below this threshold. (Optional) |
| **HighThreshold** | Log if statistical value goes above this threshold. (Optional) |
| **BackoffInterval** | The minimum number of seconds between consecutive log messages. (Default: 60) |
| **Continuous** | If set, generate event if the value goes from being outside the threshold values, back to within acceptable limits again. (Default: No) |
| **LogMessageID** | ID of generated log messages. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

### Note

*If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.106. RemoteMgmtHTTP

**Description**

Configure HTTP/HTTPS management to enable remote management to the system.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. (Identifier) |
| **Interface** | Specifies the interface for which remote access is granted. |
| **Network** | Specifies the network for which remote access is granted. |
| **HTTP** | Enable remote management via HTTP. (Default: No) |
| **HTTPS** | Enable remote management via HTTPS. (Default: No) |
| **AuthSource** | Optionally enable authentication from an external source. Note that a Local User Database must ALWAYS be configured to prevent administrative lockout in cases where the external source may not be available. (Default: LocalOnly) |
| **AuthOrder** | Specifies if the local database should be queried before or after the external database. (Default: LocalLast) |
| **LocalUserDatabase** | Specifies the local user database to use for login. |
| **AccessLevel** | Optionally restrict the access level of users authenticated by the local database. (Default: Admin) |
| **RadiusServers** | Specifies the authentication servers that will be used to authenticate users matching this rule. |
| **RadiusMethod** | Specifies the authentication method used for encrypting the user password. (Default: PAP) |
| **ChallengeExpire** | How long, in seconds, before RADIUS challenge expires. (Default: 160) |
| **PrimaryRetryInterval** | How many seconds to wait before trying to use the primary server again if it has failed. (Default: 0) |
| **AdminGroups** | Restricts administration access to specific user groups. (Optional) |
| **AuditGroups** | Restricts auditing access to specific user groups. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

# 3.107. RemoteMgmtREST

**Description**

Configure REST API management to enable API management to the system.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. (Identifier) |
| **Interface** | Specifies the interface for which remote access is granted. |
| **Network** | Specifies the network for which remote access is granted. |
| **HTTP** | Enable remote management via HTTP. (Default: No) |
| **HTTPS** | Enable remote management via HTTPS. (Default: No) |
| **AccessLevel** | Restrict access level to the REST API. (Default: ReadWrite) |
| **BasicAUTH** | Require authentication using Basic AUTH. (Default: No) |
| **Username** | Specifies the username used for Basic AUTH. |
| **Password** | Specifies the password used for Basic AUTH. |
| **Comments** | Text describing the current object. (Optional) |

# 3.108. RemoteMgmtSettings

**Description**

Setup and configure methods and permissions for remote management of this system.

**Properties**

**NetconBiDirTimeout**                    Specifies the amount of seconds to wait for the administrator to log in before reverting to the previous configuration. (Default: 30)

**WebUIBeforeRules**                      Enable HTTP(S) traffic to the firewall regardless of configured IP Rules. (Default: Yes)

**WWWSrv_HTTPPort**                       Specifies the HTTP port for the web user interface. (Default: 80)

**WWWSrv_HTTPSPort**                      Specifies the HTTPS port for the web user interface. (Default: 443)

**WebUIAllowLoginAutoComplete**          Allow the web browser to remember the username and password on the login page. (Default: No)

**SSHBeforeRules**                        Enable SSH traffic to the firewall regardless of configured IP Rules. (Default: Yes)

**HTTPSCertificate**                      Specifies host certificate to use for HTTPS traffic. Only RSA certificates are supported. (Optional)

**HTTPSRootCertificates**                 Specifies eventual root certificates to use for HTTPS traffic. (Optional)

**SNMPBeforeRules**                       Enable SNMP traffic to the firewall regardless of configured IP Rules. (Default: Yes)

**SNMPRequestLimit**                      Maximum number of SNMP packets that will be processed each second. (Default: 100)

**SNMPSysContact**                        The contact person for this managed node. (Default: N/A)

**SNMPSysName**                           The name for this managed node. (Default: N/A)

**SNMPSysLocation**                       The physical location of this node. (Default: N/A)

**SNMPIfDescription**                     What to display in the SNMP MIB-II ifDescr variables. (Default: Name)

**SNMPIfAlias**                           What to display in the SNMP ifMIB ifAlias variables. (Default: Hardware)

**LocalConsoleIdleTimeout**               Number of seconds of inactivity until the local (serial) console user is automatically logged out. (Default: 900)

**WebUIIdleTimeout**                      Number of seconds of inactivity until the HTTP(S) session is closed. (Default: 900)

**SNMPPersistentIfIndexes**               Make SNMP interface indexes persistent over

reboots. Disabling and later re-enabling this setting will trigger a re-numbering of all interfaces in the system. (Default: No)

**Note**
*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.109. RemoteMgmtSNMP

**Description**

Configure SNMP management to enable SNMP polling.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. (Identifier) |
| **Interface** | Specifies the interface for which remote access is granted. |
| **Network** | Specifies the network for which remote access is granted. |
| **SnmpVersion** | Enabled SNMP version. (Default: SNMPv1_SNMPv2c) |
| **Snmp3SecurityLevel** | Enabled SNMPv3 security level. (Default: noAuthNoPriv) |
| **SNMPGetCommunity** | Specifies the name of the community to be granted rights to remotely monitor the firewall. |
| **LocalUserDatabase** | Specifies the local user database to use for authentication. |
| **Comments** | Text describing the current object. (Optional) |

# 3.110. RemoteMgmtSSH

**Description**

Configure a Secure Shell (SSH) Server to enable remote management access to the system.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the SSH server. (Identifier) |
| **Interface** | Specifies the interface for which remote access is granted. |
| **Network** | Specifies the network for which remote access is granted. |
| **Port** | The listening port for the SSH server. (Default: 22) |
| **AllowAuthMethodPassword** | Allow password client authentication. (Default: Yes) |
| **AllowAuthMethodPublicKey** | Allow public key client authentication. (Default: Yes) |
| **AllowHostKeyDSA** | Allow DSA public key algorithm. (Default: Yes) |
| **AllowHostKeyRSA** | Allow RSA public key algorithm. (Default: Yes) |
| **AllowKexDH14** | Allow Diffie-Hellman Group 14 key exchange algorithm. (Default: Yes) |
| **AllowKexDH1** | Allow Diffie-Hellman Group 1 key exchange algorithm. (Default: Yes) |
| **AllowAES128** | Allow AES-128-CBC encryption algorithm. (Default: Yes) |
| **AllowAES192** | Allow AES-192-CBC encryption algorithm. (Default: Yes) |
| **AllowAES256** | Allow AES-256-CBC encryption algorithm. (Default: Yes) |
| **AllowAES128Ctr** | Allow AES-128-CTR encryption algorithm. (Default: Yes) |
| **AllowAES192Ctr** | Allow AES-192-CTR encryption algorithm. (Default: Yes) |
| **AllowAES256Ctr** | Allow AES-256-CTR encryption algorithm. (Default: Yes) |
| **AllowBlowfish** | Allow Blowfish encryption algorithm. (Default: Yes) |
| **Allow3DES** | Allow 3DES encryption algorithm. (Default: Yes) |
| **AllowMACSHA1** | Allow SHA1 integrity algorithm. (Default: Yes) |
| **AllowMACMD5** | Allow MD5 integrity algorithm. (Default: No) |

**AllowMACSHA196**                    Allow SHA1-96 integrity algorithm. (Default: Yes)

**AllowMACMD596**                     Allow MD5-96 integrity algorithm. (Default: No)

**Banner**                            Specifies the greeting message to display when the user logs in. (Optional)

**MaxSessions**                       The maximum number of clients that can be connected at the same time. (Default: 5)

**SessionIdleTime**                   The number of seconds a user can be idle before the session is closed. (Default: 1800)

**LoginGraceTime**                    When the user has supplied the username, the password has to be provided within this number of seconds or the session will be closed. (Default: 30)

**AuthenticationRetries**             The number of retires allowed before the session is closed. (Default: 3)

**AuthSource**                        Optionally enable authentication from an external source. Note that a Local User Database must ALWAYS be configured to prevent administrative lockout in cases where the external source may not be available. (Default: LocalOnly)

**AuthOrder**                         Specifies if the local database should be queried before or after the external database. (Default: LocalLast)

**LocalUserDatabase**                 Specifies the local user database to use for login.

**AccessLevel**                       Optionally restrict the access level of users authenticated by the local database. (Default: Admin)

**RadiusServers**                     Specifies the authentication servers that will be used to authenticate users matching this rule.

**RadiusMethod**                      Specifies the authentication method used for encrypting the user password. (Default: PAP)

**ChallengeExpire**                   How long, in seconds, before RADIUS challenge expires. (Default: 160)

**PrimaryRetryInterval**              How many seconds to wait before trying to use the primary server again if it has failed. (Default: 0)

**AdminGroups**                       Restricts administration access to specific user groups. (Optional)

**AuditGroups**                       Restricts auditing access to specific user groups. (Optional)

**Comments**                          Text describing the current object. (Optional)

*287*

# 3.111. RoamingVPN

## Description

This type of IPsec tunnel is used when you want to create VPN tunnels to roaming clients like mobile phones or laptops. The tunnel type is pre-configured to successfully connect with iOS, macOS and Windows clients using IKEv2 and EAP-MSCHAPv2. (IKEv2 and EAP-MSCHAPv2 is the default setting on most clients.)

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **GatewayCertificate** | Selects the certificate the firewall uses to authenticate itself to the other IPsec peer. |
| **RootCertificates** | Selects one or more root certificates to use with this IPsec Tunnel. |
| **IPPoolAddress** | Specifies the set of IP addresses to use for assigning IP addresses to VPN clients. |
| **DNS** | Specifies the IP address of a DNS server that a VPN client should be able to connect to. (Optional) |
| **AuthSource** | RADIUS or Local. |
| **LocalUserDB** | Specifies the local user database that will be used to authenticate users matching this rule. |
| **RadiusServer** | Specifies the authentication server that will be used to authenticate users. |
| **SNMPIndex** | Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0) |
| **MemberOfRoutingTable** | All or Specific. (Default: All) |
| **RoutingTable** | Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main) |
| **Comments** | Text describing the current object. (Optional) |

# 3.112. RouteBalancingInstance

**Description**

A route balancing instance is associated with a routingtable and defines how to make use of multiple routes to the same destination.

**Properties**

**RoutingTable**                        Specify routingtable to deploy route load balancing in. (Identifier)

**Algorithm**                           Specify which algorithm to use when balancing the routes. (Default: RoundRobin)

**Comments**                            Text describing the current object. (Optional)

# 3.113. RouteBalancingSpilloverSettings

**Description**

Settings associated with the spillover algorithm.

**Properties**

| | |
|---|---|
| **Interface** | Interface to threshold limit. (Identifier) |
| **HoldTime** | Number of consecutive seconds over/under the threshold limit to trigger state change for the affected routes. (Default: 30) |
| **OutboundThreshold** | Outbound threshold limit. (Optional) |
| **OutboundUnit** | The outbound units. (Default: kbps) |
| **InboundThreshold** | Inbound threshold limit. (Optional) |
| **InboundUnit** | The inbound units. (Default: kbps) |
| **Comments** | Text describing the current object. (Optional) |

# 3.114. RouterAdvertisement

### Description

Enabling Router Advertisement will answer Solicitations and periodically send out Advertisements. Stateless address autoconfiguration (SLAAC) will only work correctly if the configured network prefix is 64 (RFC4862).

### Properties

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the Router Advertisement. |
| **Interface** | Specifies the name of the interface to advertise on. |
| **UseGlobalRASettings** | Use global RA advanced settings. (Default: Yes) |
| **RAMaxInterval** | Maximum time between sending unsolicited multicast Router Advertisement. (Default: 600s). (Default: 600) |
| **RAMinInterval** | Minimum time between sending unsolicited multicast Router Advertisement. Will be automatically adjusted if set to less than 3 seconds or greater than .75 * Max RA Interval). (Default: 200) |
| **RAAutoLifetime** | Auto adjust the Router Lifetime field using the following formula; 3 * MaxRtrAdvInterval. (Default: Yes) |
| **RADefaultLifetime** | The value to be placed in the Router Lifetime field of Router Advertisements sent from the SGW, in seconds. (Default: 1800s). (Default: 1800) |
| **RAReachableTime** | The value to be placed in the Reachable Time field in the Router Advertisement messages SGW. The value zero means unspecified. (Default: 0s). (Default: 0) |
| **RARetransTimer** | The value to be placed in the Retrans Timer field in the Router Advertisement messages sent by the SGW. The value zero means unspecified. (Default: 0s). (Default: 0) |
| **RAManagedFlag** | Indicates that addresses are available via DHCPv6. (Default: False). (Default: No) |
| **RAOtherConfigFlag** | Indicates that other configuration information is available via DHCPv6. (Default: False). (Default: No) |
| **RACurHopLimit** | The default value to be placed in the Cur Hop Limit field in the Router Advertisement messages sent by the SGW. The value zero means unspecified. (Default: 64). (Default: 64) |
| **RALinkMTU** | The value to be placed in MTU options sent. A value of zero indicates that no MTU options are |

sent. (Default: 0). (Default: 0)

**Comments**                                Text describing the current object. (Optional)

**Note**

*If no* Index *is specified when creating an instance of this type, the object will be placed last in the list and the* Index *will be equal to the length of the list.*

# 3.114.1. RA_PrefixInformation

### Description

Specifies a Router Advertisement Prefix Information option.

**Properties**

**Name**                                Specifies a symbolic name for the Prefix Information.

**Prefix**                              Specifies the network prefix.

**RAValidLifetime**                     The value to be placed in the Valid Lifetime in the Prefix Information option. The value of 999999999 represents infinity. (Default: 2592000s). (Default: 2592000)

**RAPreferredLifetime**                 The value to be placed in the Preferred Lifetime in the Prefix Information option. The value of 999999999 represents infinity. (Default: 604800s). (Default: 604800)

**RAOnLinkFlag**                        Indicates that the advertised prefix can be used for on-link determination. (Default: True). (Default: Yes)

**RAAutonomousFlag**                    Indicates that the advertised prefix can be used for stateless address configuration. (Default: True). (Default: Yes)

**Comments**                            Text describing the current object. (Optional)

**Note**

*If no* Index *is specified when creating an instance of this type, the object will be placed last in the list and the* Index *will be equal to the length of the list.*

# 3.115. RoutingRule

### Description

A Routing Rule forces the use of a routing table in the forward and/or return direction of traffic on a connection. The ordering parameter of the routing table determines if it is consulted before or after the main routing table.

### Properties

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the rule. (Optional) |
| **ForwardRoutingTable** | The forward routing table will be used for packets from the connection originator to the connection endpoint. |
| **ReturnRoutingTable** | The return routing table will be used for packets traveling in the reverse direction. |
| **SourceNetwork** | Specifies the sender span of IP addresses to be compared to the received packet. |
| **DestinationNetwork** | Specifies the span of IP addresses to be compared to the destination IP of the received packet. |
| **SourceInterface** | Specifies the name of the source interface to be compared to the received packet. |
| **DestinationInterface** | Specifies the destination interface to be compared to the received packet. |
| **Service** | Specifies a service that will be used as a filter parameter when matching traffic with this rule. |
| **Schedule** | By adding a schedule to a rule, the firewall will only allow that rule to trigger at those designated times. (Optional) |
| **LogEnabled** | Enable logging. (Default: Yes) |
| **LogSeverity** | Specifies with what severity log events will be sent to the specified log receivers. (Default: Default) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> *If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.116. RoutingSettings

**Description**

Configure the routing capabilities of the system.

**Properties**

| | |
|---|---|
| **RouteFailOver_IfacePollInterval** | Time (ms) between polling of interface failure. (Default: 500) |
| **RouteFailOver_ARPPollInterval** | Time (ms) between ARP-lookup of gateways. May be overridden for each route. (Default: 1000) |
| **RouteFailOver_PingPollInterval** | Time (ms) between PING'ing of gateways. (Default: 1000) |
| **RouteFailOver_GraceTime** | Time (s) between startup/reconfigure and monitoring start. (Default: 30) |
| **RouteFailOver_ConsecFails** | Number of consecutive failures before route is marked as unavailable. (Default: 5) |
| **RouteFailOver_ConsecSuccess** | Number of consecutive success before route is marked as available. (Default: 5) |
| **Transp_CAMToL3CDestLearning** | Do L3 Cache learning based on destination IPs and MACs in combination with CAM table contents. (Default: Yes) |
| **Transp_DecrementTTL** | Decrement TTL on packets forwarded between transparent interfaces. (Default: No) |
| **Transp_CAMSize_Dynamic** | Allocate the CAM Size value dynamically. (Default: Yes) |
| **Transp_CAMSize** | Maximum number of entries in each CAM table. (Default: 8192) |
| **Transp_L3CSize_Dynamic** | Allocate the L3 Cache Size value dynamically. (Default: Yes) |
| **Transp_L3CSize** | Maximum number of entries in each Layer 3 Cache. (Default: 8192) |
| **Transp_RelaySTP** | Relay Spanning-Tree (STP, RSTP and MSTP) Bridge Protocol Data Units to all switch interfaces. (Default: Drop) |
| **Transp_RelayMPLS** | Forward MPLS packets to all switch interfaces. (Default: Drop) |
| **RFO_GratuitousARPOnFail** | Send gratuitous ARP on failover to alert hosts about changed interface ethernet and IP addresses. (Default: Yes) |
| **RFO_GratuitousProxyARPOnFail** | Send gratuitous ARP packets on failover to alert hosts about changed interface ethernet address of Proxy ARPed hosts in ARP cache. (Default: No) |

**Transparency_ATSExpire**

Lifetime of an unanswered ATS entry in seconds. (Default: 3)

**Transparency_ATSSize**

Number of ATS entries, total. (Default: 4096)

**NullEnetSender**

Action to take if sender MAC in the ethernet header is the null address (0000:0000:0000). (Default: DropLog)

**BroadcastEnetSender**

Action to take if sender MAC in the ethernet header is the broadcast ethernet address (FFFF:FFFF:FFFF). (Default: DropLog)

**MulticastEnetSender**

Action to take if sender MAC in the ethernet header is a multicast ethernet address. (Default: DropLog)

*Note*
*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.117. RoutingTable

### Description

The system has a predefined main routing table. Alternate routing tables can be defined by the user.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the routing table. (Identifier) |
| **Ordering** | Specifies how a route lookup is done in a named routing table. (Default: Only) |
| **RemoveInterfaceIPRoutes** | Removes the interface routes. Makes the firewall completely transparent. (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

# 3.117.1. Route

### Description

A route defines what interface and gateway to use in order to reach a specified network.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. (Optional) |
| **Interface** | Specifies which interface packets destined for this route shall be sent through. |
| **Gateway** | Specifies the IP address of the next router hop used to reach the destination network. If the network is directly connected to the firewall interface, no gateway address is specified. (Optional) |
| **LocalIP** | The IP address specified here will be automatically published on the corresponding interface. This address will also be used as the sender address in ARP queries. If no address is specified, the firewall's interface IP address will be used. (Optional) |
| **Network** | Specifies the network address for this route. |
| **BroadcastFwd** | By default, this traffic is dropped. (Default: No) |
| **RouteMonitor** | Specifies if this route should be monitored for route changes for route failover purposes. (Default: No) |
| **MonitorLinkStatus** | Mark the route as down if the interface link status changes to down. (Default: No) |

| | |
|---|---|
| **MonitorGateway** | Mark the route as down if the next hop does not answer on ARP lookups during a specified time. (Default: No) |
| **MonitorGatewayARPInterval** | Specifies the ARP lookup interval in milliseconds. (Default: 1000) |
| **EnableHostMonitoring** | Enables the Host Monitoring functionality. (Default: No) |
| **Reachability** | Specifies the number of hosts that are required to be reachable to consider the route to be active. (Default: ALL) |
| **GracePeriod** | Specifies the time to wait after a reconfiguration until the monitoring begins. (Default: 5) |
| **ReachabilityCount** | Minimum number of reachable hosts to consider the route to be active. |
| **Metric** | Specifies the metric for this route. (Default: 100) |
| **ProxyARPAllInterfaces** | Always select all interfaces, including new ones, for publishing routes via Proxy ARP. (Default: No) |
| **ProxyARPInterfaces** | Specifies the interfaces on which the firewall should publish routes via Proxy ARP. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

> ### Note
> If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.

## 3.117.1.1. MonitoredHost

### Description

Specify a host and a monitoring method.

### Properties

| | |
|---|---|
| **Method** | Monitoring method. (Default: ICMP) |
| **IPAddress** | Specifies the IP address of the host to monitor. |
| **Port** | Specifies the TCP port to monitor. |
| **SourceIPSelection** | Which IP should be used as a source IP. (Default: Automatic) |
| **SourceIP** | The IP address to be used as source IP. |
| **PollingInterval** | Delay in milliseconds between each monitor attempt. (Default: 10000) |

| | |
|---|---|
| **ReachabilityRequired** | Specifies if this host is required to be reachable for monitoring to be successful. (Default: No) |
| **Samples** | Specifies the number of attempts to use for statistical calculations. (Default: 10) |
| **MaxPollFails** | Specifies the maximum number of failed attempts until host is considered to be unreachable. (Default: 2) |
| **MaxAverageLatency** | Specifies the max average latency for the sample attempts. (Default: 800) |
| **RequestURL** | Specifies the HTTP URL to monitor. |
| **ExpectedResponse** | Expected HTTP response. |
| **Comments** | Text describing the current object. (Optional) |

> ***Note***
> *If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

## 3.117.2. Route6

**Description**

A route defines what interface and gateway to use in order to reach a specified network.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. (Optional) |
| **Network** | Specifies the network address for this route. |
| **Interface** | Specifies which interface packets destined for this route shall be sent through. |
| **Gateway** | Specifies the IPv6 address of the next router hop used to reach the destination network. If the network is directly connected to the firewall interface, no gateway address is specified. (Optional) |
| **LocalIP** | The IPv6 address specified here will be automatically published on the corresponding interface. This address will also be used as the sender address in ARP queries. If no address is specified, the firewall's interface IPv6 address will be used. (Optional) |
| **Metric** | Specifies the metric for this route. (Default: 100) |
| **ProxyNDAllInterfaces** | Always select all interfaces, including new ones, for |

| | |
|---|---|
| | publishing routes via Proxy Neighbor Discovery. (Default: No) |
| **ProxyNDInterfaces** | Specifies the interfaces on which the firewall should publish routes via Proxy ARP. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

### Note
*If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

## 3.117.3. SwitchRoute

### Description

A switch route defines which interfaces the specified network can be reached on. Proxy ARP defines between which interfaces ARP is allowed.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the object. (Optional) |
| **Interface** | Specifies which interface packets destined for this route shall be sent through. |
| **Network** | Specifies the network address for this route. |
| **BroadcastFwd** | By default, this traffic is dropped. (Default: No) |
| **Metric** | Specifies the metric for this route. (Default: 100) |
| **ProxyARPAllInterfaces** | Always select all interfaces, including new ones, for publishing routes via Proxy ARP. (Default: No) |
| **ProxyARPInterfaces** | Specifies the interfaces on which the firewall should publish routes via Proxy ARP. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

### Note
*If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

# 3.118. ScannerProtection

**Description**

Scanner Protection protects the firewall against various forms of scanning, probing and password brute force attacks. Detected scanner sources are automatically blacklisted for efficient blocking. Specific hosts can be excluded from Scanner Protection using the Whitelist.

**Properties**

**EnableScannerBlacklist**  Scanner Protection looks up source IP addresses in the IP reputation database and adds malicious sources to the Blacklist. (Default: No)

**Interfaces**  Interfaces to protect from attacks. Normally the interfaces towards the Internet. (Optional)

**LogEnabled**  Enable logging. (Default: Yes)

**LogSeverity**  Specifies with what severity log events will be sent to the specified log receivers. (Default: Default)

**Comments**  Text describing the current object. (Optional)

> ***Note***
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.119. ScheduleProfile

**Description**

A Schedule Profile defines days and dates and are then used by the various policies in the system.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the service. (Identifier) |
| **Mon** | Specifies during which intervals the schedule profile is active on Mondays. (Optional) |
| **Tue** | Specifies during which intervals the schedule profile is active on Tuesdays. (Optional) |
| **Wed** | Specifies during which intervals the schedule profile is active on Wednesdays. (Optional) |
| **Thu** | Specifies during which intervals the schedule profile is active on Thursdays. (Optional) |
| **Fri** | Specifies during which intervals the schedule profile is active on Fridays. (Optional) |
| **Sat** | Specifies during which intervals the schedule profile is active on Saturdays. (Optional) |
| **Sun** | Specifies during which intervals the schedule profile is active on Sundays. (Optional) |
| **StartDate** | The date after which this Schedule should be active. (Optional) |
| **EndDate** | The date after which this Schedule is not active anymore. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

# 3.120. ServiceGroup

## Description

A Service Group is a collection of service objects, which can then be used by different policies in the system.

## Properties

**Name**                          Specifies a symbolic name for the service. (Identifier)

**Members**                       Group members.

**Comments**                      Text describing the current object. (Optional)

# 3.121. ServiceICMP

**Description**

An ICMP Service is an object definition representing ICMP traffic with specific parameters.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the service. (Identifier) |
| **MessageTypes** | Specifies the ICMP message types that are applicable to this service. (Default: All) |
| **EchoRequest** | Enable matching of Echo Request messages. (Default: No) |
| **EchoRequestCodes** | Specifies which Echo Request message codes should be matched. (Default: 0-255) |
| **DestinationUnreachable** | Enable matching of Destination Unreachable messages. (Default: No) |
| **DestinationUnreachableCodes** | Specifies which Destination Unreachable message codes should be matched. (Default: 0-255) |
| **Redirect** | Enable matching of Redirect messages. (Default: No) |
| **RedirectCodes** | Specifies which Redirect message codes should be matched. (Default: 0-255) |
| **ParameterProblem** | Enable matching of Parameter Problem messages. (Default: No) |
| **ParameterProblemCodes** | Specifies which Parameter Problem message codes should be matched. (Default: 0-255) |
| **EchoReply** | Enable matching of Echo Reply messages. (Default: No) |
| **EchoReplyCodes** | Specifies which Echo Reply message codes should be matched. (Default: 0-255) |
| **SourceQuenching** | Enable matching of Source Quenching messages. (Default: No) |
| **SourceQuenchingCodes** | Specifies which Source Quenching message codes should be matched. (Default: 0-255) |
| **TimeExceeded** | Enable matching of Time Exceeded messages. (Default: No) |
| **TimeExceededCodes** | Specifies which Time Exceeded message codes should be matched. (Default: 0-255) |
| **ForwardICMPErrors** | Allow ICMP errors for active connections to be forwarded through the system. (Default: No) |
| **EnableIPv4PathMTUDiscovery** | Path MTU Discovery allows communicating |

|  |  |
|---|---|
|  | endpoints to negotiate optimal packet sizes. This prevents fragmentation by network equipment between the endpoints. Path MTU Discovery relies on ICMP message forwarding so ICMP forwarding must also be enabled. (Default: No) |
| **Protocol** | Protocol settings are only used by IP Policies. (Optional) |
| **MaxSessionsProtocol** | Specifies how many concurrent sessions that are permitted using this Protocol. (Default: 200) |
| **ALG** | An Application Layer Gateway (ALG), capable of managing advanced protocols, can be specified for this service. (Optional) |
| **MaxSessions** | Specifies how many concurrent sessions that are permitted using this service. (Default: 200) |
| **Comments** | Text describing the current object. (Optional) |

# 3.122. ServiceICMPv6

**Description**

An ICMPv6 Service is an object definition representing ICMPv6 traffic with specific parameters.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the service. (Identifier) |
| **MessageTypes** | Specifies the ICMPv6 message types that are applicable to this service. (Default: All) |
| **EchoRequest** | Enable matching of Echo Request messages. (Default: No) |
| **EchoRequestCodes** | Specifies which Echo Request message codes should be matched. (Default: 0-255) |
| **EchoReply** | Enable matching of Echo Reply messages. (Default: No) |
| **EchoReplyCodes** | Specifies which Echo Reply message codes should be matched. (Default: 0-255) |
| **DestinationUnreachable** | Enable matching of Destination Unreachable messages. (Default: No) |
| **DestinationUnreachableCodes** | Specifies which Destination Unreachable message codes should be matched. (Default: 0-255) |
| **PacketTooBig** | Enable matching of Packet Too Big messages. (Default: No) |
| **PacketTooBigCodes** | Specifies which Packet Too Big message codes should be matched. (Default: 0-255) |
| **TimeExceeded** | Enable matching of Time Exceeded messages. (Default: No) |
| **TimeExceededCodes** | Specifies which Time Exceeded message codes should be matched. (Default: 0-255) |
| **ParameterProblem** | Enable matching of Parameter Problem messages. (Default: No) |
| **ParameterProblemCodes** | Specifies which Parameter Problem message codes should be matched. (Default: 0-255) |
| **ForwardICMPErrors** | Allow ICMP errors for active connections to be forwarded through the system. (Default: No) |
| **EnableIPv4PathMTUDiscovery** | Path MTU Discovery allows communicating endpoints to negotiate optimal packet sizes. This prevents fragmentation by network equipment between the endpoints. Path MTU Discovery relies on ICMP message forwarding so ICMP forwarding must also be enabled. (Default: No) |

**Protocol**                              Protocol settings are only used by IP Policies.
                                          (Optional)

**MaxSessionsProtocol**                   Specifies how many concurrent sessions that are
                                          permitted using this Protocol. (Default: 200)

**ALG**                                   An Application Layer Gateway (ALG), capable of
                                          managing advanced protocols, can be specified for
                                          this service. (Optional)

**MaxSessions**                           Specifies how many concurrent sessions that are
                                          permitted using this service. (Default: 200)

**Comments**                              Text describing the current object. (Optional)

# 3.123. ServiceIPProto

### Description

An IP Protocol Service is a definition of an IP protocol with specific parameters.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the service. (Identifier) |
| **IPProto** | IP protocol number or range, e.g. "1-4,7" will match the protocols ICMP, IGMP, GGP, IP-in-IP and CBT. (Default: 0-255) |
| **ForwardICMPErrors** | Allow ICMP errors for active connections to be forwarded through the system. (Default: No) |
| **EnableIPv4PathMTUDiscovery** | Path MTU Discovery allows communicating endpoints to negotiate optimal packet sizes. This prevents fragmentation by network equipment between the endpoints. Path MTU Discovery relies on ICMP message forwarding so ICMP forwarding must also be enabled. (Default: No) |
| **Protocol** | Protocol settings are only used by IP Policies. (Optional) |
| **MaxSessionsProtocol** | Specifies how many concurrent sessions that are permitted using this Protocol. (Default: 200) |
| **ALG** | An Application Layer Gateway (ALG), capable of managing advanced protocols, can be specified for this service. (Optional) |
| **MaxSessions** | Specifies how many concurrent sessions that are permitted using this service. (Default: 200) |
| **Comments** | Text describing the current object. (Optional) |

# 3.124. ServiceTCPUDP

### Description

A TCP/UDP Service is a definition of an TCP or UDP protocol with specific parameters.

### Properties

**Name**                                    Specifies a symbolic name for the service. (Identifier)

**DestinationPorts**                        Specifies the destination port or the port ranges applicable to this service.

**Type**                                    Specifies whether this service uses the TCP or UDP protocol or both. (Default: TCP)

**SourcePorts**                             Specifies the source port or the port ranges applicable to this service. (Default: 0-65535)

**SYNRelay**                                Enable SYN flood protection (SYN Relay). (Default: No)

**ForwardICMPErrors**                       Allow ICMP errors for active connections to be forwarded through the system. (Default: No)

**EnableIPv4PathMTUDiscovery**              Path MTU Discovery allows communicating endpoints to negotiate optimal packet sizes. This prevents fragmentation by network equipment between the endpoints. Path MTU Discovery relies on ICMP message forwarding so ICMP forwarding must also be enabled. (Default: No)

**Protocol**                                Protocol settings are only used by IP Policies. (Optional)

**MaxSessionsProtocol**                     Specifies how many concurrent sessions that are permitted using this Protocol. (Default: 200)

**ALG**                                     An Application Layer Gateway (ALG), capable of managing advanced protocols, can be specified for this service. (Optional)

**MaxSessions**                             Specifies how many concurrent sessions that are permitted using this service. (Default: 200)

**Comments**                                Text describing the current object. (Optional)

# 3.125. SLBPolicy

The definitions here are the same as in Section 3.68.2, "SLBPolicy" .

# 3.126. SSHClientKey

**Description**

The public key of the client connecting to the SSH server.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the key. (Identifier) |
| **Type** | DSA or RSA. (Default: DSA) |
| **Subject** | Value of the Subject header tag of the public key file. (Optional) |
| **PublicKey** | Specifies the public key. |
| **Comments** | Text describing the current object. (Optional) |

# 3.127. SSLSettings

**Description**

Settings related to SSL (Secure Sockets Layer).

**Properties**

| | |
|---|---|
| **SSL_ProcessingPriority** | The amount of CPU time that SSL processing is allowed to use. (Default: Normal) |
| **MinTLSVersion** | Minimum allowed version. TLSv1.1 is not supported. (Default: TLSv12) |
| **TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256** | Enable cipher TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256. (Default: Yes) |
| **TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA1** | Enable cipher TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA1. (Default: Yes) |
| **TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA1** | Enable cipher TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA1. (Default: Yes) |
| **TLS_RSA_WITH_AES_256_CBC_SHA256** | Enable cipher TLS_RSA_WITH_AES_256_CBC_SHA256. (Default: Yes) |
| **TLS_RSA_WITH_AES_256_CBC_SHA1** | Enable cipher TLS_RSA_WITH_AES_256_CBC_SHA1. (Default: Yes) |
| **TLS_RSA_WITH_AES_128_CBC_SHA256** | Enable cipher TLS_RSA_WITH_AES_128_CBC_SHA256. (Default: Yes) |
| **TLS_RSA_WITH_AES_128_CBC_SHA1** | Enable cipher TLS_RSA_WITH_AES_128_CBC_SHA1. (Default: Yes) |
| **TLS_RSA_WITH_3DES_168_SHA1** | Enable cipher RSA_WITH_3DES_168_SHA1. (Default: Yes) |
| **TLS_RSA_WITH_RC4_128_SHA1** | Enable cipher RSA_WITH_RC4_128_SHA1. (Default: No) |
| **TLS_RSA_WITH_RC4_128_MD5** | Enable cipher TLS_RSA_WITH_RC4_128_MD5. (Default: No) |
| **TLS_RSA_EXPORT1024_WITH_RC4_56_SHA1** | Enable cipher TLS_RSA_EXPORT1024_WITH_RC4_56_SHA1. (Default: No) |
| **TLS_RSA_EXPORT512_WITH_RC4_40_MD5** | Enable cipher TLS_RSA_EXPORT1024_WITH_RC4_40_MD5. (Default: No) |

**TLS_RSA_EXPORT512_WITH_RC2_40_MD5** Enable cipher TLS_RSA_EXPORT1024_WITH_RC2_40_MD5. (Default: No)

**TLS_RSA_EXPORT_WITH_NULL_SHA1** Enable cipher TLS_RSA_EXPORT_WITH_NULL_SHA1 (no encryption, just message validation). (Default: No)

**TLS_RSA_EXPORT_WITH_NULL_MD5** Enable cipher TLS_RSA_EXPORT_WITH_NULL_MD5 (no encryption, just message validation). (Default: No)

> ***Note***
>
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.128. SSLVPNInterface

## Description

An SSL VPN interface, together with the bundled client, creates an easy to use tunnel solution for roaming users.

## Properties

**Name**

Specifies a symbolic name for the interface. (Identifier)

**OuterInterface**

The physical interface that the SSL VPN interface will listen on.

**ServerPort**

The listening port for the SSL VPN interface. (Default: 443)

**ServerIP**

Listening IP for the SSL VPN interface.

**ServerFQDN**

Optional. FQDN of the SSL VPN server given to clients, eg: (sslvpn.example.com). (Optional)

**IPAddressPool**

A range, group or network that will be the IP pool from which the SSL VPN clients will receive their IP addresses.

**InnerIP**

Local IP for the SSL VPN interface.

**PrimaryDNS**

IP of the primary DNS Server. (Optional)

**SecondaryDNS**

IP of the seconday DNS Server. (Optional)

**Routing**

Describes how the traffic from the client should be routed. (Default: All-Nets)

**ClientRoutes**

Networks to be routed through the SSL VPN tunnel in the client.

**SNMPIndex**

Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0)

**MemberOfRoutingTable**

All or Specific. (Default: All)

**RoutingTable**

Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main)

**ProxyARPAllInterfaces**

Always select all interfaces, including new ones, for publishing routes via Proxy ARP. (Default: No)

**ProxyARPInterfaces**

Specifies the interfaces on which the firewall should publish routes via Proxy ARP. (Optional)

**Comments**

Text describing the current object. (Optional)

# 3.129. SSLVPNInterfaceSettings

**Description**

SSL VPN interface settings.

**Properties**

**SSLVPNBeforeRules**                          Pass SSL VPN connections sent to the firewall
                                               directly to the SSL VPN engine without consulting
                                               the ruleset. (Default: Yes)

> ### *Note*
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.130. StatelessPolicy

The definitions here are the same as in Section 3.68.4, "StatelessPolicy" .

# 3.131. StateSettings

**Description**

Parameters for the state engine in the system.

**Properties**

| | |
|---|---|
| **ConnReplace** | What to do when the connection table is full. (Default: ReplaceLog) |
| **LogOpenFails** | Log packets that are neither part of open connections nor valid new connections. (Default: Yes) |
| **LogReverseOpens** | Log reverse connection attempts through an established connection. (Default: Yes) |
| **LogStateViolations** | Log packets that violate stateful tracking rules; for instance, TCP connect sequences. (Default: Yes) |
| **LogConnections** | Log connections opening and closing. (Default: Log) |
| **IPReputationLogs** | Enable reputation logging of all IPv4 addresses seen by the system. (Default: Yes) |
| **LogConnectionUsage** | Log for every packet that passes through a connection. (Default: No) |
| **MaxConnections_Dynamic** | Allocate the Max Connection value dynamically. (Default: Yes) |
| **MaxConnections** | Maximum number of simultaneous connections. (Default: 8192) |

***Note***
*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.132. TCPSettings

**Description**

Settings related to the TCP protocol.

**Properties**

| | |
|---|---|
| **TCPOptionSizes** | Validity of TCP header option sizes. (Default: ValidateLogBad) |
| **TCPMSSMin** | Minimum allowed TCP MSS (Maximum Segment Size). (Default: 100) |
| **TCPMSSOnLow** | How to handle too low MSS values. (Default: DropLog) |
| **TCPMSSMax** | Maximum allowed TCP MSS (Maximum Segment Size). (Default: 1460) |
| **TCPMSSVPNMax** | Limits TCP MSS for VPN connections; minimizes fragmentation. (Default: 1400) |
| **TCPMSSOnHigh** | How to handle too high MSS values. (Default: Adjust) |
| **TCPMSSLogLevel** | When to log regarding too high TCP MSS, if not logged by "TCP MSS on high". (Default: 7000) |
| **TCPMSSAutoClamping** | Automatically clamp TCP MSS according to MTU of involved interfaces - in addition to "TCP MSS max". (Default: Yes) |
| **TCPZeroUnusedACK** | Force unused ACK fields to zero; helps prevent connection spoofing. (Default: Yes) |
| **TCPZeroUnusedURG** | Force unused URG fields to zero; prevents small information leak. (Default: Yes) |
| **TCPOPT_WSOPT** | The WSOPT (Window Scale) option (common). (Default: ValidateLogBad) |
| **TCPOPT_SACK** | The SACK/SACKPERMIT (Selective ACK) options (common). (Default: ValidateLogBad) |
| **TCPOPT_TSOPT** | The TSOPT (Timestamp) option (common). (Default: ValidateLogBad) |
| **TCPOPT_ALTCHKREQ** | The ALTCHKREQ (Alternate Checksum Request) option. (Default: StripLog) |
| **TCPOPT_ALTCHKDATA** | The ALTCHKDATA (Alternate Checksum Data) option. (Default: StripLog) |
| **TCPOPT_CC** | The CC (Connection Count) option series (semi common). (Default: StripLogBad) |
| **TCPOPT_OTHER** | How to handle TCP options not specified above. (Default: StripLog) |

*317*

**TCPSynUrg**                     The TCP URG flag together with SYN; normally invalid (strip=strip URG). (Default: DropLog)

**TCPSynPsh**                     The TCP PSH flag together with SYN; normally invalid but always used by some IP stacks (strip=strip PSH). (Default: StripSilent)

**TCPSynRst**                     The TCP RST flag together with SYN; normally invalid (strip=strip RST). (Default: DropLog)

**TCPSynFin**                     The TCP FIN flag together with SYN; normally invalid (strip=strip FIN). (Default: DropLog)

**TCPSynFrag**                    Fragmented data together with SYN; not invalid but can be used for DoS attacks. (Default: DropLog)

**TCPSynData**                    Payload data together with SYN; not invalid but can be used for DoS attacks. (Default: DropLog)

**TCPFinUrg**                     The TCP URG flag together with FIN; normally invalid (strip=strip URG). (Default: DropLog)

**TCPUrg**                        The TCP URG flag; many operating systems cannot handle this correctly. (Default: StripLog)

**TCPECN**                        The Explicit Congestion Notification (ECN) flags. Previously known as "XMAS"/"YMAS" flags. Also used in OS fingerprinting. (Default: StripLog)

**TCPRF**                         The TCP Reserved field: should be zero. Used in OS fingerprinting. Also part of ECN extension. (Default: StripLog)

**TCPNULL**                       TCP "NULL" packets without SYN, ACK, FIN or RST; normally invalid, used by scanners. (Default: DropLog)

**TCPSequenceNumbers**            Validation of TCP sequence numbers. (Default: ValidateLogBad)

**TCPAllowReopen**                Allow clients to re-open TCP connections that are in the closed state. (Default: No)

### Note

*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.133. ThresholdRule

**Description**

A Threshold Rule defines a filter for matching specific network traffic. When the filter criterion is met, the Threshold Rule Actions are evaluated and possible actions taken.

**Properties**

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the rule. (Optional) |
| **SourceInterface** | Specifies the name of the receiving interface to be compared to the received packet. |
| **SourceNetwork** | Specifies the sender span of IP addresses to be compared to the received packet. |
| **DestinationInterface** | Specifies the destination interface to be compared to the received packet. |
| **DestinationNetwork** | Specifies the span of IP addresses to be compared to the destination IP of the received packet. |
| **Service** | Specifies a service that will be used as a filter parameter when matching traffic with this rule. |
| **Schedule** | By adding a schedule to a rule, the firewall will only allow that rule to trigger at those designated times. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.

# 3.133.1. ThresholdAction

**Description**

A Threshold Rule Action specifies what thresholds to measure, and what action to take if those thresholds are reached.

**Properties**

| | |
|---|---|
| **Action** | Protect or Audit. (Default: Protect) |
| **GroupBy** | Specifies whether the threshold should be host- or network-based. (Default: SourceIP) |
| **Threshold** | Specifies the threshold. |

**ThresholdUnit**                    Specifies the threshold unit. (Default: ConnsSec)

**ZoneDefense**                      Activate ZoneDefense. (Default: No)

**BlackList**                        Activate BlackList. (Default: No)

**BlackListTimeToBlock**             The number of seconds that the dynamic black list should remain. (Optional)

**BlackListBlockOnlyService**        Only block the service that triggered the blacklisting. (Default: No)

**BlackListIgnoreEstablished**       Do not drop existing connection. (Default: No)

**LogEnabled**                       Enable logging. (Default: Yes)

**LogSeverity**                      Specifies with what severity log events will be sent to the specified log receivers. (Default: Default)

**Comments**                         Text describing the current object. (Optional)

---

### *Note*
*If no* Index *is specified when creating an instance of this type, the object will be placed last in the list and the* Index *will be equal to the length of the list.*

---

# 3.134. UpdateCenter

**Description**

Configure automatical updates.

**Properties**

| | |
|---|---|
| **AVEnabled** | Automatic updates of antivirus definitions and engine. (Default: No) |
| **IDPEnabled** | Automatic updates of IDP signatures. (Default: No) |
| **UpdateInterval** | Specifies the interval at which the automatic update runs. (Default: Daily) |
| **UpdateDate** | Specifies the day of month when the automatic update is run. |
| **UpdateWeekday** | Specifies the day of week when the automatic update is run. (Default: mon) |
| **Hourly** | Specifies the number of hours between periodical updates. |
| **UpdateHour** | Specifies the hour when the update is run. (Default: 0) |
| **UpdateMinute** | Specifies the minute when the update is run. (Default: 0) |
| **DisableUpdateAfterReconf** | Only update at the specified update interval (disables updates at startup and reconfiguration). (Default: No) |
| **Comments** | Text describing the current object. (Optional) |

> **Note**
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.135. UserAuthRule

**Description**

The User Authentication Ruleset specifies from where users are allowed to authenticate to the system, and how.

**Properties**

| | |
|---|---|
| **Index** | The index of the object, starting at 1. (Identifier) |
| **Name** | Specifies a symbolic name for the rule. |
| **Agent** | ARPCache, HTTP, HTTPS, XAuth, PPP or EAP. (Default: HTTP) |
| **ChallengeExpire** | How long, in seconds, before RADIUS challenge expires. (Default: 160) |
| **AuthSource** | Disallow, LDAP, RADIUS or Local. |
| **Interface** | The interface on which the connection was received. For agent type EAP or XAuth, this should be the IPsec tunnel interface the user connects through. |
| **OriginatorIP** | The network object that the incoming IP address must be a part of. |
| **TerminatorIP** | Specifies the destination IP configured on the PPTP/L2TP server configuration. Only used when agent is PPP or SSL. With SSL, this is the IP address of the listening interface. |
| **RadiusServers** | Specifies the authentication servers that will be used to authenticate users matching this rule. |
| **PrimaryRetryInterval** | How many seconds to wait before trying to use the primary server again if it has failed. (Default: 0) |
| **ResendingSTART** | If the RADIUS servers fail to respond system will retry to send a START message every Interim seconds. (Default: No) |
| **LDAPServers** | Specifies the authentication servers that will be used to authenticate users matching this rule. |
| **RadiusMethod** | Specifies the authentication method used for encrypting the user password. (Default: PAP) |
| **LocalUserDB** | Specifies the local user database that will be used to authenticate users matching this rule. |
| **LoginType** | HTML form or Basic authentication. (Default: HTMLForm) |
| **MACAuthSecret** | Password used to authenticate MAC user, if empty the MAC address will be sent as password. (Optional) |

| | |
|---|---|
| **MACAllowRouter** | Allow cliente connected through an Router. (Default: No) |
| **MACSendUpperCase** | Send upper cased MAC address. (Default: No) |
| **HTTPBanners** | HTTP Authentication HTML Banners. (Default: Default) |
| **RealmString** | The string that is presented as a part of the 401 - Authentication Required message. (Optional) |
| **HostCertificate** | Specifies the host certificate that the firewall sends to the client. Only RSA certificates are supported. |
| **RootCertificate** | Specifies the root certificate that was used to sign the host certificate. Only RSA certificates are supported. (Optional) |
| **PPPAuthNoAuth** | Allow no authentication. (Default: No) |
| **PPPAuthPAP** | Use PAP authentication protocol. User name and password are sent in plaintext. (Default: Yes) |
| **PPPAuthCHAP** | Use CHAP authentication protocol. (Default: Yes) |
| **PPPAuthMSCHAP** | Use MS-CHAP authentication protocol. (Default: Yes) |
| **PPPAuthMSCHAPv2** | Use MS-CHAP v2 authentication protocol. (Default: Yes) |
| **IdleTimeout** | A successfully authenticated user will be logged out automatically after this many seconds, if no traffic has been received from the user's IP address. (Default: 1800) |
| **SessionTimeout** | A successfully authenticated user will be logged out automatically after this many seconds, even if traffic has been received from the user's IP address. (Optional) |
| **UseServerTimeouts** | Use timeouts received from the authentication server. If no values are received, the manually specified values will be used. (Default: No) |
| **MultipleUsernameLogins** | Specifies how multiple username logins will be handled. (Default: AllowMultiple) |
| **ReplaceIdleTime** | Replace existing user if idle for more than this number of seconds. (Default: 10) |
| **AccountingServers** | Specifies the accounting servers that will be used to report user usage matching this rule. (Optional) |
| **PrimaryRetryIntervalAcc** | How many seconds to wait before trying to use the primary server again if it has failed. (Default: 0) |
| **BytesSent** | Enable reporting of the number of bytes sent by the user. (Default: Yes) |
| **PacketsSent** | Enable reporting of the number of packets sent by the user. (Default: Yes) |

*323*

**BytesReceived**                     Enable reporting of the number of bytes received by the user. (Default: Yes)

**PacketsReceived**                   Enable reporting of the number of packets received by the user. (Default: Yes)

**SessionTime**                       Enable reporting of the number of seconds the session lasted. (Default: Yes)

**SupportInterimAccounting**          Enable Interim Accounting Messages to update the accounting server with the current status of an authenticated user. (Default: No)

**ServerInterimControl**              Let the RADIUS server determine the interval that interim accounting events should be sent. (Default: Yes)

**InterimValue**                      The interval in seconds in which interim accounting events should be sent. (Default: 600)

**LogEnabled**                        Enable logging. (Default: Yes)

**LogSeverity**                       Specifies with what severity log events will be sent to the specified log receivers. (Default: Default)

**Comments**                          Text describing the current object. (Optional)

---

*Note*

*If no* `Index` *is specified when creating an instance of this type, the object will be placed last in the list and the* `Index` *will be equal to the length of the list.*

---

# 3.136. VLAN

### Description

Use a VLAN to define a virtual interface compatible with the IEEE 802.1Q / 802.1ad Virtual LAN standard.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the interface. (Identifier) |
| **VLANID** | The virtual LAN ID used for this virtual LAN interface. Two virtual LANs cannot have the same VLAN ID and type if they are based on the same interface. (Default: 0) |
| **BaseInterface** | Interface where this VLAN is being tunneled. |
| **Type** | VLAN type. (Default: 0x8100) |
| **IP** | The IP address of the virtual LAN interface. |
| **Network** | The network address of the virtual LAN interface. |
| **DefaultGateway** | The default gateway of the virtual LAN interface. (Optional) |
| **Broadcast** | The broadcast address of the virtual LAN interface. (Optional) |
| **DHCPEnabled** | Enable DHCP client on this interface. (Default: No) |
| **DHCPHostName** | Optional DHCP Host Name. Leave blank to use default name. (Optional) |
| **DHCPDNS1** | IP of the primary DNS server. (Optional) |
| **DHCPDNS2** | IP of the secondary DNS server. (Optional) |
| **EnableIPv6** | Enable processing of IPv6 traffic on this interface. (Default: No) |
| **IPv6IP** | The IPv6 address of the virtual LAN interface. |
| **IPv6Network** | The IPv6 network of the virtual LAN interface. |
| **IPv6DefaultGateway** | The default gateway of the virtual LAN interface. (Optional) |
| **RouterDiscovery** | Uses Router information (ND RA) from local network to auto-configure Network and Default Gateway addresses. (Default: No) |
| **AutoIPv6IP** | Automatically configures IP Address using Network Address and EUI-64. (Default: No) |
| **DHCPv6Enabled** | Enable DHCPv6 client on this interface. (Default: No) |

**PrivateIP**

The private IP address of this high availability node. (Optional)

**PrivateIP6**

The private IP6 address of this high availability node. (Default: localhost6)

**Metric**

Specifies the metric for the auto-created route. (Default: 100)

**AutoSwitchRoute**

Allows traffic to be forwarded transparently across all interfaces with Transparent Mode enabled that belong to the same routing table. (Default: No)

**DHCPPassthrough**

Allow DHCP to pass through transparently. (Default: No)

**NonIPPassthrough**

Allow non-IP protocols to pass through transparently. (Default: No)

**BroadcastFwd**

By default, this traffic is dropped. (Default: No)

**AutoInterfaceNetworkRoute**

Automatically add a route for this virtual LAN interface using the given network. (Default: Yes)

**AutoDefaultGatewayRoute**

Automatically add a default route for this virtual LAN interface using the given default gateway. (Default: Yes)

**DHCPv6DNS1**

IP of the primary IPv6 DNS server. (Optional)

**DHCPv6DNS2**

IP of the secondary IPv6 DNS server. (Optional)

**PrioCopyPolicy**

Set the QoS to VLAN priority copy policy. (Default: Inherit)

**EnableRouterAdvertisement**

Enable Router Advertisement for this interface. (Default: No)

**SNMPIndex**

Interface index assigned by the system when persistent interface indexes are enabled. (Default: 0)

**MemberOfRoutingTable**

All or Specific. (Default: All)

**RoutingTable**

Specifies the PBR table to insert the interface IP route into. It also means that the specified routing table will be used for all routing lookups, unless overridden by a PBR rule. (Default: main)

**Comments**

Text describing the current object. (Optional)

# 3.137. VLANSettings

### Description

Settings for IEEE 802.1Q based Virtual LAN interfaces.

### Properties

**UnknownVLANTags**                           VLAN packets tagged with an unknown ID.
                                              (Default: DropLog)

***Note***
*This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.138. VoIPProfile

### Description

A VoIP Profile can be used by one or many IP Policies which has its service object configured with SIP or H.323 as protocol.

### Properties

| | |
|---|---|
| **Name** | Specifies a symbolic name for the Profile. (Identifier) |
| **SIP** | Enables automatic pinhole creation for SIP sessions. (Default: Yes) |
| **SIPMaxSessionsPerId** | Maximum number of sessions per SIP URI. (Default: 5) |
| **SIPMaxRegistrationTime** | The maximum allowed time in seconds between registration requests. (Default: 3600) |
| **SIPSignalTimeout** | Timeout value for last seen SIP message (in seconds). (Default: 43200) |
| **SIPDataChannelTimeout** | Specifies how many seconds a data channel may remain inactive before it is closed. (Default: 120) |
| **SIPAllowMediaBypass** | Allow clients to exchange media directly when possible. (Default: Yes) |
| **SIPAllowTCPDataChannels** | Allow data channels to be established over TCP in addition to UDP. (Default: Yes) |
| **SIPMaxTCPDataChannels** | Maximum number of TCP data channels per call. (Default: 5) |
| **H323** | Enables automatic pinhole creation for H.323 sessions. (Default: Yes) |
| **H323AllowTCPDataChannels** | Allow data channels to be established over TCP in addition to UDP. (Default: Yes) |
| **H323MaxTCPDataChannels** | Maximum number of TCP data channels per call. (Default: 10) |
| **H323TranslateAddresses** | Specifies address translation behavior. (Default: Automatic) |
| **H323TranslateLogicalChannelAddresses** | Enable address translation for logical channels. (Default: Yes) |
| **H323MaxGKRegLifeTime** | The gatekeeper registration lifetime can be controlled in order to force re-registration by clients within a certain time. A shorter time forces more frequent registration by clients with the gatekeeper and less probability of a problem if the network becomes unavailable and the client thinks it is still registered. (Default: 1800) |

**H323ChannelSetupMode**                    Channel connection setup mode. (Default: Optimistic)

**Comments**                                Text describing the current object. (Optional)

# 3.139. WebProfile

**Description**

A Web Profile can be used by one or many IP Policies which has its service object configured with HTTP or HTTPS as protocol.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the Profile. (Identifier) |
| **HTTPBanners** | Specifies web page to present when access to a site is denied. (Default: Default) |
| **AllowProtocolUpgrade** | Allow the connection to be upgraded to another protocol e.g. a WebSocket. A protocol upgrade will disable further content inspection for the upgraded connection. (Default: Yes) |
| **WCF** | Use Web Content Filtering to monitor and/or deny access to restricted web sites based on a simple content category system. (Default: No) |
| **WCFAuditMode** | Use audit mode to allow, but still log, access to restricted sites. (Default: No) |
| **WCFCategories** | Specifies restricted web content categories. (Optional; Default: ADULT_CONTENT,BOTNETS,CHILD_ABUSE_MATERIAL,CRIME_TERROF |
| **WCFNonManagedAction** | Action to take for content that has not been classified. (Default: Allow) |
| **WCFAllowOverride** | Allows users to override the filter and gain access to blocked sites, with a warning that their actions will be logged. (Default: No) |
| **WCFOverrideTimeToLive** | Specifies how many seconds that a successful override remains in effect before the restricted site notice page reappears. (Default: 300) |
| **WCFOverrideUpdateOnAccess** | Reset the override timer on activity. (Default: Yes) |
| **WCFAllowReclassification** | Allows users to suggest new categories for blocked sites. This should under normal circumstances NEVER be enabled on profiles that affect end-users as it can be abused greatly. (Default: No) |
| **FailModeBehavior** | Standard behaviour on errors related to Web Profile features e.g. WCF. (Default: Allow) |
| **Comments** | Text describing the current object. (Optional) |

# 3.139.1. URLFilterPolicy_URL

**Description**

Blacklist URLs to deny access to complete sites, to file types by extension, or to URLs with certain words in them.

**Properties**

| | |
|---|---|
| **Action** | Whitelist, Blacklist or Redirect the matching URL filter. (Default: Blacklist) |
| **URL** | Specifies the URL to blacklist, whitelist or redirect. |
| **RedirectTo** | Specifies the URL to redirect requests to. Must be begin with 'http://' or 'https://'. (Default: http://) |
| **Comments** | Text describing the current object. (Optional) |

### Note
*If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.140. ZoneDefenseBlock

**Description**

Manually configured blocks are used to block a host/network on the switches either by default or based on schedule.

**Properties**

| | |
|---|---|
| **Addresses** | Specifies the addresses to block. |
| **Protocol** | All, TCP, UDP or ICMP. (Default: All) |
| **Port** | Specifies which UDP or TCP port to use. (Default: 0) |
| **Schedule** | Specifies the schedule when the given addresses should be blocked. (Optional) |
| **Comments** | Text describing the current object. (Optional) |

> *Note*
> *If no `Index` is specified when creating an instance of this type, the object will be placed last in the list and the `Index` will be equal to the length of the list.*

# 3.141. ZoneDefenseExcludeList

**Description**

The exclude list is used exclude certain hosts/networks from being blocked out by IDP/Threshold rule violations.

**Properties**

**Addresses**                                    Specifies the addresses that should not be blocked. (Optional)

**Comments**                                     Text describing the current object. (Optional)

> ***Note***
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# 3.142. ZoneDefenseSwitch

**Description**

A ZoneDefense switch will have its ACLs controlled and hosts/networks violating the IDP/Threshold rules will be blocked directly on the switch.

**Properties**

| | |
|---|---|
| **Name** | Specifies a symbolic name for the ZoneDefense switch. (Identifier) |
| **SwitchModel** | Specifies the switch model type. (Default: DES-3226S) |
| **IP** | The IP address of the management interface of the switch. |
| **Enabled** | Enable the ZoneDefense switch. (Default: Yes) |
| **SNMPCommunity** | The SNMP community string (write access). |
| **Comments** | Text describing the current object. (Optional) |

# 3.143. ZoneDefenseSwitchSettings

**Description**

Advanced ZoneDefense Switch Settings.

**Properties**

**SupervisorEnabled**  Enables automatic unblocking of hosts that has been blocked a configurable period of time. A host is only unblocked if the number of times it has been blocked during a supervision period (the contravention value) does not exceed the tolerance, otherwise it must be manually unblocked. (Default: Yes)

**ContraventionTolerance**  The maximum number of times ZoneDefense can unblock the host. Once a host exceeds this value it remains blocked until it is manually unblocked. (Default: 3)

**BlockTime**  A host is kept blocked this many seconds times the hosts contravention value. If the contravention value exceeds the configured tolerance it will remain blocked. (Default: 300)

> ***Note***
> *This object type does not have an identifier and is identified by the name of the type only. There can only be one instance of this type.*

# Index

# Commands

# Object types