firewall-cmd

Name

firewall-cmd — firewalld command line client

Synopsis

`firewall-cmd` [OPTIONS...]

Description

firewall-cmd is the command line client of the firewalld daemon. It provides an interface to manage the runtime and permanent configurations.

The runtime configuration in firewalld is separated from the permanent configuration. This means that things can get changed in the runtime or permanent configuration.

Options

Sequence options are the options that can be specified multiple times, the exit code is 0 if there is at least one item that succeeded. The `ALREADY_ENABLED` (11), `NOT_ENABLED` (12) and also `ZONE_ALREADY_SET` (16) errors are treated as succeeded. If there are issues while parsing the items, then these are treated as warnings and will not change the result as long as there is a succeeded one. Without any succeeded item, the exit code will depend on the error codes. If there is exactly one error code, then this is used. If there are more than one then `UNKNOWN_ERROR` (254) will be used.

The following options are supported:

General Options

`-h`, `--help`
Prints a short help text and exits.

`-V`, `--version`
Print the version string of firewalld. This option is not combinable with other options.

`-q`, `--quiet`
Do not print status messages.

Status Options

`--state`
Check whether the firewalld daemon is active (i.e. running). Returns an exit code 0 if it is active, `RUNNING_BUT_FAILED` if failure occurred on startup, `NOT_RUNNING` otherwise. See the section called “Exit Codes”. This will also print the state to `STDOUT`.

`--reload`
Reload firewall rules and keep state information. Current permanent configuration will become new runtime configuration, i.e. all runtime only changes done until reload are lost with reload if they have not been also in permanent configuration.

Note: If FlushAllOnReload=no, runtime changes applied via the direct interface are not affected and will therefore stay in place until firewalld daemon is restarted completely. For FlushAllOnReload, see firewalld.conf(5).

`--complete-reload`
Reload firewall completely, even netfilter kernel modules. This will most likely terminate active connections, because state information is lost. This option should only be used in case of severe

Search...

Recent Posts

Strict Filtering of Docker Containers
firewalld 2.1.0 release
firewalld 2.0.0 release
Software fastpath with nftables flowtable
Zone Priorities

Quick Links

Report a new issue
Browse issues

firewall problems. For example if there are state information problems that no connection can be established with correct firewall rules.

Note: If FlushAllOnReload=no, runtime changes applied via the direct interface are not affected and will therefore stay in place until firewalld daemon is restarted completely. For FlushAllOnReload, see firewalld.conf(5).

`--runtime-to-permanent`

Save active runtime configuration and overwrite permanent configuration with it. The way this is supposed to work is that when configuring firewalld you do runtime changes only and once you're happy with the configuration and you tested that it works the way you want, you save the configuration to disk.

`--check-config`

Run checks on the permanent configuration. This includes XML validity and semantics.

Log Denied Options

`--get-log-denied`

Print the log denied setting.

`--set-log-denied` = *`value`*

Add logging rules right before reject and drop rules in the INPUT, FORWARD and OUTPUT chains for the default rules and also final reject and drop rules in zones for the configured link-layer packet type. The possible values are: *`all`* , *`unicast`* , *`broadcast`* , *`multicast`* and *`off`* . The default setting is *`off`* , which disables the logging.

This is a runtime and permanent change and will also reload the firewall to be able to add the logging rules.

Permanent Options

`--permanent`

The permanent option `--permanent` can be used to set options permanently. These changes are not effective immediately, only after service restart/reload or system reboot. Without the `--permanent` option, a change will only be part of the runtime configuration.

If you want to make a change in runtime and permanent configuration, use the same call with and without the `--permanent` option.

The `--permanent` option can be optionally added to all options further down where it is supported.

Zone Options

`--get-default-zone`

Print default zone for connections and interfaces.

`--set-default-zone` = *`zone`*

Set default zone for connections and interfaces where no zone has been selected. Setting the default zone changes the zone for the connections or interfaces, that are using the default zone.

This is a runtime and permanent change.

`--get-active-zones`

Print currently active zones altogether with interfaces and sources used in these zones. Active zones are zones, that have a binding to an interface or source. The output format is:

```
zone1
  interfaces: interface1 interface2 ..
  sources: source1 ..
zone2
  interfaces: interface3 ..
zone3
  sources: source2 ..
```

If there are no interfaces or sources bound to the zone, the corresponding line will be omitted.

[`--permanent`] `--get-zones`

Print predefined zones as a space separated list.

[`--permanent`] `--get-services`
Print predefined services as a space separated list.

[`--permanent`] `--get-icmptypes`
Print predefined icmptypes as a space separated list.

[`--permanent`] `--get-zone-of-interface`=*`interface`*
Print the name of the zone the *`interface`* is bound to or *no zone*.

[`--permanent`] `--get-zone-of-source`=*`source`*[/*`mask`*]|*`MAC`*|ipset:*`ipset`*
Print the name of the zone the source is bound to or *no zone*.

[`--permanent`] `--info-zone=`*`zone`*
Print information about the zone *`zone`*. The output format is:

```
zone
  interfaces: interface1 ..
  sources: source1 ..
  services: service1 ..
  ports: port1 ..
  protocols: protocol1 ..
  forward-ports:
        forward-port1
        ..
  source-ports: source-port1 ..
  icmp-blocks: icmp-type1 ..
  rich rules:
        rich-rule1
        ..
```

[`--permanent`] `--list-all-zones`
List everything added for or enabled in all zones. The output format is:

```
zone1
  interfaces: interface1 ..
  sources: source1 ..
  services: service1 ..
  ports: port1 ..
  protocols: protocol1 ..
  forward-ports:
        forward-port1
        ..
  icmp-blocks: icmp-type1 ..
  rich rules:
        rich-rule1
        ..
..
```

`--permanent` `--new-zone`=*`zone`*
Add a new permanent and empty zone.

Zone names must be alphanumeric and may additionally include characters: '_' and '-'.

`--permanent` `--new-zone-from-file`=*`filename`* [`--name`=*`zone`*]
Add a new permanent zone from a prepared zone file with an optional name override.

`--permanent` `--delete-zone`=*`zone`*
Delete an existing permanent zone.

`--permanent` `--load-zone-defaults`=*`zone`*
Load zone default settings or report NO_DEFAULTS error.

`--permanent` `--path-zone=`*`zone`*
Print path of the zone configuration file.

Policy Options

[`--permanent`] `--get-policies`
Print predefined policies as a space separated list.

[`--permanent`] `--info-policy` = *`policy`*
Print information about the policy *`policy`*.

[`--permanent`] `--list-all-policies`
List everything added for or enabled in all policies.

`--permanent` `--new-policy` = *`policy`*
Add a new permanent policy.

Policy names must be alphanumeric and may additionally include characters: '_' and '-'.

`--permanent` `--new-policy-from-file` = *`filename`* [`--name` = *`policy`*]
Add a new permanent policy from a prepared policy file with an optional name override.

`--permanent` `--path-policy` = *`policy`*
Print path of the policy configuration file.

`--permanent` `--delete-policy` = *`policy`*
Delete an existing permanent policy.

`--permanent` `--load-policy-defaults` = *`policy`*
Load the shipped defaults for a policy. Only applies to policies shipped with firewalld. Does not apply to user defined policies.

Options to Adapt and Query Zones and Policies

Options in this section affect only one particular zone or policy. If used with `--zone` = *`zone`* or `--policy` = *`policy`* option, they affect the specified zone or policy. If both options are omitted, they affect the default zone (see `--get-default-zone`).

[`--permanent`] [`--zone` = *`zone`*] [`--policy` = *`policy`*] `--list-all`
List everything added or enabled.

`--permanent` [`--zone` = *`zone`*] [`--policy` = *`policy`*] `--get-target`
Get the target.

`--permanent` [`--zone` = *`zone`*] [`--policy` = *`policy`*] `--set-target` = *`target`*
Set the target.

For zones *`target`* is one of: `default`, `ACCEPT`, `DROP`, `REJECT`

For policies *`target`* is one of: `CONTINUE`, `ACCEPT`, `DROP`, `REJECT`

`default` is similar to `REJECT`, but it implicitly allows ICMP packets.

`--permanent` [`--zone` = *`zone`*] [`--policy` = *`policy`*] `--set-description` = *`description`*
Set description.

`--permanent` [`--zone` = *`zone`*] [`--policy` = *`policy`*] `--get-description`
Print description.

`--permanent` [`--zone` = *`zone`*] [`--policy` = *`policy`*] `--set-short` = *`description`*
Set short description.

`--permanent` [`--zone` = *`zone`*] [`--policy` = *`policy`*] `--get-short`
Print short description.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--list-services`
List services added as a space separated list.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--add-service` = *`service`* [`--timeout` = *`timeval`*]
Add a service. This option can be specified multiple times. If a timeout is supplied, the rule will be active for the specified amount of time and will be removed automatically afterwards. *`timeval`* is either a number (of seconds) or number followed by one of characters `s` (seconds), `m` (minutes), `h` (hours), for example `20m` or `1h`.

The service is one of the firewalld provided services. To get a list of the supported services, use **firewall-cmd --get-services**.

The `--timeout` option is not combinable with the `--permanent` option.

Note: Some services define connection tracking helpers. Helpers that may operate in client mode (e.g. tftp) must be added to an outbound policy instead of a zone to take effect for clients. Otherwise the helper will not be applied to the outbound traffic. The related traffic, as defined by the connection tracking helper, on the return path (ingress) will be allowed by the stateful firewall rules.

An example of an outbound policy for connection tracking helpers:

```
# firewall-cmd --permanent --new-policy clientConntrack
# firewall-cmd --permanent --policy clientConntrack --add-ingress-zone HOST
# firewall-cmd --permanent --policy clientConntrack --add-egress-zone ANY
# firewall-cmd --permanent --policy clientConntrack --add-service tftp
```

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--remove-service` = *`service`*

Remove a service. This option can be specified multiple times.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--query-service` = *`service`*

Return whether *`service`* has been added. Returns 0 if true, 1 otherwise.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--list-ports`

List ports added as a space separated list. A port is of the form *`portid`* [- *`portid`*]/ *`protocol`* , it can be either a port and protocol pair or a port range with a protocol.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--add-port` = *`portid`* [- *`portid`*]/ *`protocol`* [`--timeout` = *`timeval`*]

Add the port. This option can be specified multiple times. If a timeout is supplied, the rule will be active for the specified amount of time and will be removed automatically afterwards. *`timeval`* is either a number (of seconds) or number followed by one of characters `s` (seconds), `m` (minutes), `h` (hours), for example `20m` or `1h` .

The port can either be a single port number or a port range *`portid`* - *`portid`* . The protocol can either be `tcp` , `udp` , `sctp` or `dccp` .

The `--timeout` option is not combinable with the `--permanent` option.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--remove-port` = *`portid`* [- *`portid`*]/ *`protocol`*

Remove the port. This option can be specified multiple times.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--query-port` = *`portid`* [- *`portid`*]/ *`protocol`*

Return whether the port has been added. Returns 0 if true, 1 otherwise.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--list-protocols`

List protocols added as a space separated list.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--add-protocol` = *`protocol`* [`--timeout` = *`timeval`*]

Add the protocol. This option can be specified multiple times. If a timeout is supplied, the rule will be active for the specified amount of time and will be removed automatically afterwards. *`timeval`* is either a number (of seconds) or number followed by one of characters `s` (seconds), `m` (minutes), `h` (hours), for example `20m` or `1h` .

The protocol can be any protocol supported by the system. Please have a look at `/etc/protocols` for supported protocols.

The `--timeout` option is not combinable with the `--permanent` option.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--remove-protocol` = *`protocol`*

Remove the protocol. This option can be specified multiple times.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--query-protocol` = *`protocol`*

Return whether the protocol has been added. Returns 0 if true, 1 otherwise.

[`--permanent`] [`--zone` = *`zone`*] [`--permanent`] [`--policy` = *`policy`*] `--list-source-ports`

List source ports added as a space separated list. A port is of the form `portid` [- `portid`]/ `protocol` .

[`--permanent`] [`--zone` = `zone`] [`--permanent`] [`--policy` = `policy`] `--add-source-port` = `portid` [- `portid`]/ `protocol` [`--timeout` = `timeval`]

Add the source port. This option can be specified multiple times. If a timeout is supplied, the rule will be active for the specified amount of time and will be removed automatically afterwards. `timeval` is either a number (of seconds) or number followed by one of characters `s` (seconds), `m` (minutes), `h` (hours), for example `20m` or `1h` .

The port can either be a single port number or a port range `portid` - `portid` . The protocol can either be `tcp` , `udp` , `sctp` or `dccp` .

The `--timeout` option is not combinable with the `--permanent` option.

[`--permanent`] [`--zone` = `zone`] [`--permanent`] [`--policy` = `policy`] `--remove-source-port` = `portid` [- `portid`]/ `protocol`

Remove the source port. This option can be specified multiple times.

[`--permanent`] [`--zone` = `zone`] [`--permanent`] [`--policy` = `policy`] `--query-source-port` = `portid` [- `portid`]/ `protocol`

Return whether the source port has been added. Returns 0 if true, 1 otherwise.

[`--permanent`] [`--zone` = `zone`] [`--permanent`] [`--policy` = `policy`] `--list-icmp-blocks`

List Internet Control Message Protocol (ICMP) type blocks added as a space separated list.

[`--permanent`] [`--zone` = `zone`] [`--permanent`] [`--policy` = `policy`] `--add-icmp-block` = `icmptype` [`--timeout` = `timeval`]

Add an ICMP block for `icmptype` . This option can be specified multiple times. If a timeout is supplied, the rule will be active for the specified amount of time and will be removed automatically afterwards. `timeval` is either a number (of seconds) or number followed by one of characters `s` (seconds), `m` (minutes), `h` (hours), for example `20m` or `1h` .

The `icmptype` is the one of the icmp types firewalld supports. To get a listing of supported icmp types: **firewall-cmd --get-icmptypes**

The `--timeout` option is not combinable with the `--permanent` option.

[`--permanent`] [`--zone` = `zone`] [`--permanent`] [`--policy` = `policy`] `--remove-icmp-block` = `icmptype`

Remove the ICMP block for `icmptype` . This option can be specified multiple times.

[`--permanent`] [`--zone` = `zone`] [`--permanent`] [`--policy` = `policy`] `--query-icmp-block` = `icmptype`

Return whether an ICMP block for `icmptype` has been added. Returns 0 if true, 1 otherwise.

[`--permanent`] [`--zone` = `zone`] [`--permanent`] [`--policy` = `policy`] `--list-forward-ports`

List *IPv4* forward ports added as a space separated list.

For *IPv6* forward ports, please use the rich language.

[`--permanent`] [`--zone` = `zone`] [`--permanent`] [`--policy` = `policy`] `--add-forward-port` =port= `portid` [- `portid`]:proto= `protocol` [:toport= `portid` [- `portid`]] [:toaddr= `address` [/ `mask`]] [`--timeout` = `timeval`]

Add the *IPv4* forward port. This option can be specified multiple times. If a timeout is supplied, the rule will be active for the specified amount of time and will be removed automatically afterwards. `timeval` is either a number (of seconds) or number followed by one of characters `s` (seconds), `m` (minutes), `h` (hours), for example `20m` or `1h` .

The port can either be a single port number `portid` or a port range `portid` - `portid` . The protocol can either be `tcp` , `udp` , `sctp` or `dccp` . The destination address is a simple IP address.

The `--timeout` option is not combinable with the `--permanent` option.

For *IPv6* forward ports, please use the rich language.

Note: IP forwarding will be implicitly enabled if `toaddr` is specified.

[`--permanent`] [`--zone` = `zone`] [`--permanent`] [`--policy` = `policy`] `--remove-forward-port` =port= `portid` [- `portid`]:proto= `protocol` [:toport= `portid` [- `portid`]]

[:toaddr=`address`[/`mask`]]
Remove the *IPv4* forward port. This option can be specified multiple times.

For *IPv6* forward ports, please use the rich language.

[`--permanent`] [`--zone`=`zone`] [`--permanent`] [`--policy`=`policy`] `--query-forward-port`=port=`portid`[-`portid`]:proto=`protocol`[:toport=`portid`[-`portid`]]
[:toaddr=`address`[/`mask`]]
Return whether the *IPv4* forward port has been added. Returns 0 if true, 1 otherwise.

For *IPv6* forward ports, please use the rich language.

[`--permanent`] [`--zone`=`zone`] [`--permanent`] [`--policy`=`policy`] `--add-masquerade` [`--timeout`=`timeval`]
Enable *IPv4* masquerade. If a timeout is supplied, masquerading will be active for the specified amount of time. `timeval` is either a number (of seconds) or number followed by one of characters `s` (seconds), `m` (minutes), `h` (hours), for example `20m` or `1h`. Masquerading is useful if the machine is a router and machines connected over an interface in another zone should be able to use the first connection.

The `--timeout` option is not combinable with the `--permanent` option.

For *IPv6* masquerading, please use the rich language.

Note: IP forwarding will be implicitly enabled.

[`--permanent`] [`--zone`=`zone`] [`--permanent`] [`--policy`=`policy`] `--remove-masquerade`
Disable *IPv4* masquerade. If the masquerading was enabled with a timeout, it will be disabled also.

For *IPv6* masquerading, please use the rich language.

[`--permanent`] [`--zone`=`zone`] [`--permanent`] [`--policy`=`policy`] `--query-masquerade`
Return whether *IPv4* masquerading has been enabled. Returns 0 if true, 1 otherwise.

For *IPv6* masquerading, please use the rich language.

[`--permanent`] [`--zone`=`zone`] [`--permanent`] [`--policy`=`policy`] `--list-rich-rules`
List rich language rules added as a newline separated list.

[`--permanent`] [`--zone`=`zone`] [`--permanent`] [`--policy`=`policy`] `--add-rich-rule`='`rule`' [`--timeout`=`timeval`]
Add rich language rule '`rule`'. This option can be specified multiple times. If a timeout is supplied, the `rule` will be active for the specified amount of time and will be removed automatically afterwards. `timeval` is either a number (of seconds) or number followed by one of characters `s` (seconds), `m` (minutes), `h` (hours), for example `20m` or `1h`.

For the rich language rule syntax, please have a look at firewalld.richlanguage(5).

The `--timeout` option is not combinable with the `--permanent` option.

[`--permanent`] [`--zone`=`zone`] [`--permanent`] [`--policy`=`policy`] `--remove-rich-rule`='`rule`'
Remove rich language rule '`rule`'. This option can be specified multiple times.

For the rich language rule syntax, please have a look at firewalld.richlanguage(5).

[`--permanent`] [`--zone`=`zone`] [`--permanent`] [`--policy`=`policy`] `--query-rich-rule`='`rule`'
Return whether a rich language rule '`rule`' has been added. Returns 0 if true, 1 otherwise.

For the rich language rule syntax, please have a look at firewalld.richlanguage(5).

Options to Adapt and Query Zones

Options in this section affect only one particular zone. If used with `--zone`=`zone` option, they affect the specified zone. If the option is omitted, they affect default zone (see `--get-default-zone`).

[`--permanent`] [`--zone`=`zone`] `--add-icmp-block-inversion`
Enable ICMP block inversion.

[`--permanent`] [`--zone`=`zone`] `--remove-icmp-block-inversion`

Disable ICMP block inversion.

[`--permanent`] [`--zone` = *zone*] `--query-icmp-block-inversion`
Return whether ICMP block inversion is enabled. Returns 0 if true, 1 otherwise.

[`--permanent`] [`--zone` = *zone*] `--add-forward`
Enable intra zone forwarding.

[`--permanent`] [`--zone` = *zone*] `--remove-forward`
Disable intra zone forwarding.

[`--permanent`] [`--zone` = *zone*] `--query-forward`
Return whether intra zone forwarding is enabled. Returns 0 if true, 1 otherwise.

Options to Adapt and Query Policies

Options in this section affect only one particular policy. It's required to specify `--policy` = *policy* with these options.

`--permanent` `--policy` = *policy* `--get-priority`
Get the priority.

`--permanent` `--policy` = *policy* `--set-priority` = *priority*
Set the priority. The priority determines the relative ordering of policies. This is an integer value between -32768 and 32767 where -1 is the default value for new policies and 0 is reserved for internal use.

If a priority is < 0, then the policy's rules will execute before all rules in all zones.

If a priority is > 0, then the policy's rules will execute after all rules in all zones.

[`--permanent`] `--policy` = *policy* `--list-ingress-zones`
List ingress zones added as a space separated list.

[`--permanent`] `--policy` = *policy* `--add-ingress-zone` = *zone*
Add an ingress zone. This option can be specified multiple times.

The ingress zone is one of the firewalld provided zones or one of the pseudo-zones: HOST, ANY.

HOST is used for traffic originating from the host machine, i.e. the host running firewalld.

ANY is used for traffic originating from any zone. This can be thought of as a wild card for zones. However it does not include traffic originating from the host machine - use HOST for that.

[`--permanent`] `--policy` = *policy* `--remove-ingress-zone` = *zone*
Remove an ingress zone. This option can be specified multiple times.

[`--permanent`] `--policy` = *policy* `--query-ingress-zone` = *zone*
Return whether *zone* has been added. Returns 0 if true, 1 otherwise.

[`--permanent`] `--policy` = *policy* `--list-egress-zones`
List egress zones added as a space separated list.

[`--permanent`] `--policy` = *policy* `--add-egress-zone` = *zone*
Add an egress zone. This option can be specified multiple times.

The egress zone is one of the firewalld provided zones or one of the pseudo-zones: HOST, ANY.

For clarification on HOST and ANY see option `--add-ingress-zone`.

[`--permanent`] `--policy` = *policy* `--remove-egress-zone` = *zone*
Remove an egress zone. This option can be specified multiple times.

[`--permanent`] `--policy` = *policy* `--query-egress-zone` = *zone*
Return whether *zone* has been added. Returns 0 if true, 1 otherwise.

Options to Handle Bindings of Interfaces

Binding an interface to a zone means that this zone settings are used to restrict traffic via the interface.

Options in this section affect only one particular zone. If used with `--zone` = *zone* option, they affect the zone *zone*. If the option is omitted, they affect default zone (see `--get-default-zone`).

For a list of predefined zones use **firewall-cmd --get-zones**.

An interface name is a string up to 16 characters long, that may not contain `' '`, `'/'`, `'!'` and `'*'`.

[`--permanent`] [`--zone` = *`zone`*] `--list-interfaces`
List interfaces that are bound to zone *`zone`* as a space separated list. If zone is omitted, default zone will be used.

[`--permanent`] [`--zone` = *`zone`*] `--add-interface` = *`interface`*
Bind interface *`interface`* to zone *`zone`*. If zone is omitted, default zone will be used.

If the interface is under control of NetworkManager, it is at first connected to change the zone for the connection that is using the interface. If this fails, the zone binding is created in firewalld and the limitations below apply. For interfaces that are not under control of NetworkManager, firewalld tries to change the ZONE setting in the ifcfg file, if the file exists.

As a end user you don't need this in most cases, because NetworkManager (or legacy network service) adds interfaces into zones automatically (according to `ZONE=` option from ifcfg-*`interface`* file) if *`NM_CONTROLLED=no`* is not set. You should do it only if there's no /etc/sysconfig/network-scripts/ifcfg-*`interface`* file. If there is such file and you add interface to zone with this `--add-interface` option, make sure the zone is the same in both cases, otherwise the behaviour would be undefined. Please also have a look at the firewalld(1) man page in the *`Concepts`* section. For permanent association of interface with a zone, see also 'How to set or change a zone for a connection?' in firewalld.zones(5).

[`--permanent`] [`--zone` = *`zone`*] `--change-interface` = *`interface`*
If the interface is under control of NetworkManager, it is at first connected to change the zone for the connection that is using the interface. If this fails, the zone binding is created in firewalld and the limitations below apply. For interfaces that are not under control of NetworkManager, firewalld tries to change the ZONE setting in the ifcfg file, if the file exists.

Change zone the interface *`interface`* is bound to to zone *`zone`*. It's basically `--remove-interface` followed by `--add-interface`. If the interface has not been bound to a zone before, it behaves like `--add-interface`. If zone is omitted, default zone will be used.

[`--permanent`] [`--zone` = *`zone`*] `--query-interface` = *`interface`*
Query whether interface *`interface`* is bound to zone *`zone`*. Returns 0 if true, 1 otherwise.

[`--permanent`] `--remove-interface` = *`interface`*
If the interface is under control of NetworkManager, it is at first connected to change the zone for the connection that is using the interface. If this fails, the zone binding is created in firewalld and the limitations below apply.

For the addition or change of interfaces that are not under control of NetworkManager: firewalld tries to change the ZONE setting in the ifcfg file, if an ifcfg file exists that is using the interface.

Only for the removal of interfaces that are not under control of NetworkManager: firewalld is not trying to change the ZONE setting in the ifcfg file. This is needed to make sure that an ifdown of the interface will not result in a reset of the zone setting to the default zone. Only the zone binding is then removed in firewalld then.

Remove binding of interface *`interface`* from zone it was previously added to.

Options to Handle Bindings of Sources

Binding a source to a zone means that this zone settings will be used to restrict traffic from this source.

A source address or address range is either an IP address or a network IP address with a mask for IPv4 or IPv6 or a MAC address or an ipset with the ipset: prefix. For IPv4, the mask can be a network mask or a plain number. For IPv6 the mask is a plain number. The use of host names is not supported.

Options in this section affect only one particular zone. If used with `--zone` = *`zone`* option, they affect the zone *`zone`*. If the option is omitted, they affect default zone (see `--get-default-zone`).

For a list of predefined zones use **firewall-cmd [`--permanent`] --get-zones**.

[`--permanent`] [`--zone` = *`zone`*] `--list-sources`

List sources that are bound to zone `zone` as a space separated list. If zone is omitted, default zone will be used.

[`--permanent`] [`--zone` = `zone`] `--add-source` = `source` [/ `mask`]| `MAC` |ipset: `ipset`
Bind the source to zone `zone` . If zone is omitted, default zone will be used.

[`--zone` = `zone`] `--change-source` = `source` [/ `mask`]| `MAC` |ipset: `ipset`
Change zone the source is bound to to zone `zone` . It's basically `--remove-source` followed by `--add-source` . If the source has not been bound to a zone before, it behaves like `--add-source` . If zone is omitted, default zone will be used.

[`--permanent`] [`--zone` = `zone`] `--query-source` = `source` [/ `mask`]| `MAC` |ipset: `ipset`
Query whether the source is bound to the zone `zone` . Returns 0 if true, 1 otherwise.

[`--permanent`] `--remove-source` = `source` [/ `mask`]| `MAC` |ipset: `ipset`
Remove binding of the source from zone it was previously added to.

IPSet Options

`--get-ipset-types`
Print the supported ipset types.

`--permanent` `--new-ipset` = `ipset` `--type` = `type` [`--family` = `inet` | `inet6`] [`--option` = `key` [= `value`]]
Add a new permanent and empty ipset with specifying the type and optional the family and options like `timeout` , `hashsize` and `maxelem` . For more information please have a look at ipset(8) man page.

ipset names must be alphanumeric and may additionally include characters: '_' and '-'.

`--permanent` `--new-ipset-from-file` = `filename` [`--name` = `ipset`]
Add a new permanent ipset from a prepared ipset file with an optional name override.

`--permanent` `--delete-ipset` = `ipset`
Delete an existing permanent ipset.

`--permanent` `--load-ipset-defaults` = `ipset`
Load ipset default settings or report NO_DEFAULTS error.

[`--permanent`] `--info-ipset=` `ipset`
Print information about the ipset `ipset` . The output format is:

```
ipset
  type: type
  options: option1[=value1] ..
  entries: entry1 ..
```

[`--permanent`] `--get-ipsets`
Print predefined ipsets as a space separated list.

`--permanent` `--ipset` = `ipset` `--set-description` = `description`
Set new description to ipset

`--permanent` `--ipset` = `ipset` `--get-description`
Print description for ipset

`--permanent` `--ipset` = `ipset` `--set-short` = `description`
Set short description to ipset

`--permanent` `--ipset` = `ipset` `--get-short`
Print short description for ipset

[`--permanent`] `--ipset` = `ipset` `--add-entry` = `entry`
Add a new entry to the ipset.

Adding an entry to an ipset with option `timeout` is permitted, but these entries are not tracked by firewalld.

[`--permanent`] `--ipset` = `ipset` `--remove-entry` = `entry`
Remove an entry from the ipset.

[`--permanent`] `--ipset`=*`ipset`* `--query-entry`=*`entry`*
Return whether the entry has been added to an ipset. Returns 0 if true, 1 otherwise.

Querying an ipset with a timeout will yield an error. Entries are not tracked for ipsets with a timeout.

[`--permanent`] `--ipset`=*`ipset`* `--get-entries`
List all entries of the ipset.

[`--permanent`] `--ipset`=*`ipset`* `--add-entries-from-file`=*`filename`*
Add a new entries to the ipset from the file. For all entries that are listed in the file but already in the ipset, a warning will be printed.

The file should contain an entry per line. Lines starting with an hash or semicolon are ignored. Also empty lines.

[`--permanent`] `--ipset`=*`ipset`* `--remove-entries-from-file`=*`filename`*
Remove existing entries from the ipset from the file. For all entries that are listed in the file but not in the ipset, a warning will be printed.

The file should contain an entry per line. Lines starting with an hash or semicolon are ignored. Also empty lines.

`--permanent` `--path-ipset=`*`ipset`*
Print path of the ipset configuration file.

Service Options

Options in this section affect only one particular service.

[`--permanent`] `--info-service=`*`service`*
Print information about the service *`service`*. The output format is:

```
service
  ports: port1 ..
  protocols: protocol1 ..
  source-ports: source-port1 ..
  helpers: helper1 ..
  destination: ipv1:address1 ..
```

The following options are only usable in the permanent configuration.

`--permanent` `--new-service`=*`service`*
Add a new permanent and empty service.

Service names must be alphanumeric and may additionally include characters: '_' and '-'.

`--permanent` `--new-service-from-file`=*`filename`* [`--name`=*`service`*]
Add a new permanent service from a prepared service file with an optional name override.

`--permanent` `--delete-service`=*`service`*
Delete an existing permanent service.

`--permanent` `--load-service-defaults`=*`service`*
Load service default settings or report NO_DEFAULTS error.

`--permanent` `--path-service=`*`service`*
Print path of the service configuration file.

`--permanent` `--service`=*`service`* `--set-description`=*`description`*
Set new description to service

`--permanent` `--service`=*`service`* `--get-description`
Print description for service

`--permanent` `--service`=*`service`* `--set-short`=*`description`*
Set short description to service

`--permanent` `--service`=*`service`* `--get-short`
Print short description for service

`--permanent` `--service`=*`service`* `--add-port`=*`portid`*[-*`portid`*]/*`protocol`*

Add a new port to the permanent service.

`--permanent` `--service`=*`service`* `--remove-port`=*`portid`*[-*`portid`*]/*`protocol`*
Remove a port from the permanent service.

`--permanent` `--service`=*`service`* `--query-port`=*`portid`*[-*`portid`*]/*`protocol`*
Return whether the port has been added to the permanent service.

`--permanent` `--service`=*`service`* `--get-ports`
List ports added to the permanent service.

`--permanent` `--service`=*`service`* `--add-protocol`=*`protocol`*
Add a new protocol to the permanent service.

`--permanent` `--service`=*`service`* `--remove-protocol`=*`protocol`*
Remove a protocol from the permanent service.

`--permanent` `--service`=*`service`* `--query-protocol`=*`protocol`*
Return whether the protocol has been added to the permanent service.

`--permanent` `--service`=*`service`* `--get-protocols`
List protocols added to the permanent service.

`--permanent` `--service`=*`service`* `--add-source-port`=*`portid`*[-*`portid`*]/*`protocol`*
Add a new source port to the permanent service.

`--permanent` `--service`=*`service`* `--remove-source-port`=*`portid`*[-*`portid`*]/*`protocol`*
Remove a source port from the permanent service.

`--permanent` `--service`=*`service`* `--query-source-port`=*`portid`*[-*`portid`*]/*`protocol`*
Return whether the source port has been added to the permanent service.

`--permanent` `--service`=*`service`* `--get-source-ports`
List source ports added to the permanent service.

`--permanent` `--service`=*`service`* `--add-helper`=*`helper`*
Add a new helper to the permanent service.

`--permanent` `--service`=*`service`* `--remove-helper`=*`helper`*
Remove a helper from the permanent service.

`--permanent` `--service`=*`service`* `--query-helper`=*`helper`*
Return whether the helper has been added to the permanent service.

`--permanent` `--service`=*`service`* `--get-service-helpers`
List helpers added to the permanent service.

`--permanent` `--service`=*`service`* `--set-destination`=*`ipv`*:*`address`*[/*`mask`*]
Set destination for ipv to address[/mask] in the permanent service.

`--permanent` `--service`=*`service`* `--remove-destination`=*`ipv`*
Remove the destination for ipv from the permanent service.

`--permanent` `--service`=*`service`* `--query-destination`=*`ipv`*:*`address`*[/*`mask`*]
Return whether the destination ipv to address[/mask] has been set in the permanent service.

`--permanent` `--service`=*`service`* `--get-destinations`
List destinations added to the permanent service.

`--permanent` `--service`=*`service`* `--add-include`=*`service`*
Add a new include to the permanent service.

`--permanent` `--service`=*`service`* `--remove-include`=*`service`*
Remove a include from the permanent service.

`--permanent` `--service`=*`service`* `--query-include`=*`service`*
Return whether the include has been added to the permanent service.

`--permanent` `--service`=*`service`* `--get-includes`
List includes added to the permanent service.

Helper Options

Options in this section affect only one particular helper.

[`--permanent`] `--info-helper=`*`helper`*

Print information about the helper *`helper`*. The output format is:

```
helper
  family: family
  module: module
  ports: port1 ..
```

The following options are only usable in the permanent configuration.

`--permanent` `--new-helper=`*`helper`* `--module=`*`nf_conntrack_module`* [`--family=ipv4` | `ipv6`]

Add a new permanent helper with module and optionally family defined.

Helper names must be alphanumeric and may additionally include characters: '-'.

`--permanent` `--new-helper-from-file=`*`filename`* [`--name=`*`helper`*]

Add a new permanent helper from a prepared helper file with an optional name override.

`--permanent` `--delete-helper=`*`helper`*

Delete an existing permanent helper.

`--permanent` `--load-helper-defaults=`*`helper`*

Load helper default settings or report NO_DEFAULTS error.

`--permanent` `--path-helper=`*`helper`*

Print path of the helper configuration file.

[`--permanent`] `--get-helpers`

Print predefined helpers as a space separated list.

`--permanent` `--helper=`*`helper`* `--set-description=`*`description`*

Set new description to helper

`--permanent` `--helper=`*`helper`* `--get-description`

Print description for helper

`--permanent` `--helper=`*`helper`* `--set-short=`*`description`*

Set short description to helper

`--permanent` `--helper=`*`helper`* `--get-short`

Print short description for helper

`--permanent` `--helper=`*`helper`* `--add-port=`*`portid`*[-*`portid`*]/*`protocol`*

Add a new port to the permanent helper.

`--permanent` `--helper=`*`helper`* `--remove-port=`*`portid`*[-*`portid`*]/*`protocol`*

Remove a port from the permanent helper.

`--permanent` `--helper=`*`helper`* `--query-port=`*`portid`*[-*`portid`*]/*`protocol`*

Return whether the port has been added to the permanent helper.

`--permanent` `--helper=`*`helper`* `--get-ports`

List ports added to the permanent helper.

`--permanent` `--helper=`*`helper`* `--set-module=`*`description`*

Set module description for helper

`--permanent` `--helper=`*`helper`* `--get-module`

Print module description for helper

`--permanent` `--helper=`*`helper`* `--set-family=`*`description`*

Set family description for helper

`--permanent` `--helper=`*`helper`* `--get-family`

Print family description of helper

Internet Control Message Protocol (ICMP) type Options

Options in this section affect only one particular icmptype.

[`--permanent`] `--info-icmptype=`*`icmptype`*

Print information about the icmptype *`icmptype`*. The output format is:

```
icmptype
  destination: ipv1 ..
```

The following options are only usable in the permanent configuration.

`--permanent` `--new-icmptype`=*`icmptype`*

Add a new permanent and empty icmptype.

ICMP type names must be alphanumeric and may additionally include characters: '_' and '-'.

`--permanent` `--new-icmptype-from-file`=*`filename`* [`--name`=*`icmptype`*]

Add a new permanent icmptype from a prepared icmptype file with an optional name override.

`--permanent` `--delete-icmptype`=*`icmptype`*

Delete an existing permanent icmptype.

`--permanent` `--load-icmptype-defaults`=*`icmptype`*

Load icmptype default settings or report NO_DEFAULTS error.

`--permanent` `--icmptype`=*`icmptype`* `--set-description`=*`description`*

Set new description to icmptype

`--permanent` `--icmptype`=*`icmptype`* `--get-description`

Print description for icmptype

`--permanent` `--icmptype`=*`icmptype`* `--set-short`=*`description`*

Set short description to icmptype

`--permanent` `--icmptype`=*`icmptype`* `--get-short`

Print short description for icmptype

`--permanent` `--icmptype`=*`icmptype`* `--add-destination`=*`ipv`*

Enable destination for ipv in permanent icmptype. ipv is one of `ipv4` or `ipv6`.

`--permanent` `--icmptype`=*`icmptype`* `--remove-destination`=*`ipv`*

Disable destination for ipv in permanent icmptype. ipv is one of `ipv4` or `ipv6`.

`--permanent` `--icmptype`=*`icmptype`* `--query-destination`=*`ipv`*

Return whether destination for ipv is enabled in permanent icmptype. ipv is one of `ipv4` or `ipv6`.

`--permanent` `--icmptype`=*`icmptype`* `--get-destinations`

List destinations in permanent icmptype.

`--permanent` `--path-icmptype=`*`icmptype`*

Print path of the icmptype configuration file.

Direct Options

DEPRECATED

> The direct interface has been deprecated. It will be removed in a future release. It is superseded by policies, see firewalld.policies(5).

The direct options give a more direct access to the firewall. These options require user to know basic iptables concepts, i.e. `table` (filter/mangle/nat/...), `chain` (INPUT/OUTPUT/FORWARD/...), `commands` (-A/-D/-I/...), `parameters` (-p/-s/-d/-j/...) and `targets` (ACCEPT/DROP/REJECT/...).

Direct options should be used only as a last resort when it's not possible to use for example `--add-service`=*`service`* or `--add-rich-rule`='*`rule`'.

Warning: Direct rules behavior is different depending on the value of `FirewallBackend`. See `CAVEATS` in firewalld.direct(5).

The first argument of each option has to be `ipv4` or `ipv6` or `eb`. With `ipv4` it will be for IPv4 (iptables(8)), with `ipv6` for IPv6 (ip6tables(8)) and with `eb` for ethernet bridges (ebtables(8)).

[`--permanent`] `--direct` `--get-all-chains`
Get all chains added to all tables. This option concerns only chains previously added with `--direct --add-chain`.

[`--permanent`] `--direct` `--get-chains` { `ipv4` | `ipv6` | `eb` } *`table`*
Get all chains added to table *`table`* as a space separated list. This option concerns only chains previously added with `--direct --add-chain`.

[`--permanent`] `--direct` `--add-chain` { `ipv4` | `ipv6` | `eb` } *`table`* *`chain`*
Add a new chain with name *`chain`* to table *`table`*. Make sure there's no other chain with this name already.

There already exist basic chains to use with direct options, for example `INPUT_direct` chain (see `iptables-save | grep direct` output for all of them). These chains are jumped into before chains for zones, i.e. every rule put into `INPUT_direct` will be checked before rules in zones.

[`--permanent`] `--direct` `--remove-chain` { `ipv4` | `ipv6` | `eb` } *`table`* *`chain`*
Remove chain with name *`chain`* from table *`table`*. Only chains previously added with `--direct --add-chain` can be removed this way.

[`--permanent`] `--direct` `--query-chain` { `ipv4` | `ipv6` | `eb` } *`table`* *`chain`*
Return whether a chain with name *`chain`* exists in table *`table`*. Returns 0 if true, 1 otherwise. This option concerns only chains previously added with `--direct --add-chain`.

[`--permanent`] `--direct` `--get-all-rules`
Get all rules added to all chains in all tables as a newline separated list of the priority and arguments. This option concerns only rules previously added with `--direct --add-rule`.

[`--permanent`] `--direct` `--get-rules` { `ipv4` | `ipv6` | `eb` } *`table`* *`chain`*
Get all rules added to chain *`chain`* in table *`table`* as a newline separated list of the priority and arguments. This option concerns only rules previously added with `--direct --add-rule`.

[`--permanent`] `--direct` `--add-rule` { `ipv4` | `ipv6` | `eb` } *`table`* *`chain`* *`priority`* *`args`*
Add a rule with the arguments *`args`* to chain *`chain`* in table *`table`* with priority *`priority`*.

The *`priority`* is used to order rules. Priority 0 means add rule on top of the chain, with a higher priority the rule will be added further down. Rules with the same priority are on the same level and the order of these rules is not fixed and may change. If you want to make sure that a rule will be added after another one, use a low priority for the first and a higher for the following.

[`--permanent`] `--direct` `--remove-rule` { `ipv4` | `ipv6` | `eb` } *`table`* *`chain`* *`priority`* *`args`*
Remove a rule with *`priority`* and the arguments *`args`* from chain *`chain`* in table *`table`*. Only rules previously added with `--direct --add-rule` can be removed this way.

[`--permanent`] `--direct` `--remove-rules` { `ipv4` | `ipv6` | `eb` } *`table`* *`chain`*
Remove all rules in the chain with name *`chain`* exists in table *`table`*. This option concerns only rules previously added with `--direct --add-rule` in this chain.

[`--permanent`] `--direct` `--query-rule` { `ipv4` | `ipv6` | `eb` } *`table`* *`chain`* *`priority`* *`args`*
Return whether a rule with *`priority`* and the arguments *`args`* exists in chain *`chain`* in table *`table`*. Returns 0 if true, 1 otherwise. This option concerns only rules previously added with `--direct --add-rule`.

`--direct` `--passthrough` { `ipv4` | `ipv6` | `eb` } *`args`*
Pass a command through to the firewall. *`args`* can be all **iptables**, **ip6tables** and **ebtables** command line arguments. This command is untracked, which means that firewalld is not able to provide information about this command later on, also not a listing of the untracked passthoughs.

[`--permanent`] `--direct` `--get-all-passthroughs`
Get all passthrough rules as a newline separated list of the ipv value and arguments.

[`--permanent`] `--direct` `--get-passthroughs` { `ipv4` | `ipv6` | `eb` }
Get all passthrough rules for the ipv value as a newline separated list of the priority and arguments.

[`--permanent`] `--direct` `--add-passthrough` { `ipv4` | `ipv6` | `eb` } *`args`*
Add a passthrough rule with the arguments *`args`* for the ipv value.

[`--permanent`] `--direct` `--remove-passthrough` { `ipv4` | `ipv6` | `eb` } *`args`*
Remove a passthrough rule with the arguments *`args`* for the ipv value.

[`--permanent`] `--direct` `--query-passthrough` { `ipv4` | `ipv6` | `eb` } *`args`*
Return whether a passthrough rule with the arguments *`args`* exists for the ipv value. Returns 0 if true, 1 otherwise.

Lockdown Options

Local applications or services are able to change the firewall configuration if they are running as root (example: libvirt) or are authenticated using PolicyKit. With this feature administrators can lock the firewall configuration so that only applications on lockdown whitelist are able to request firewall changes.

The lockdown access check limits D-Bus methods that are changing firewall rules. Query, list and get methods are not limited.

The lockdown feature is a very light version of user and application policies for firewalld and is turned off by default.

`--lockdown-on`
Enable lockdown. Be careful - if firewall-cmd is not on lockdown whitelist when you enable lockdown you won't be able to disable it again with firewall-cmd, you would need to edit firewalld.conf.

This is a runtime and permanent change.

`--lockdown-off`
Disable lockdown.

This is a runtime and permanent change.

`--query-lockdown`
Query whether lockdown is enabled. Returns 0 if lockdown is enabled, 1 otherwise.

Lockdown Whitelist Options

The lockdown whitelist can contain *`commands`*, *`contexts`*, *`users`* and *`user ids`*.

If a command entry on the whitelist ends with an asterisk '*', then all command lines starting with the command will match. If the '*' is not there the absolute command inclusive arguments must match.

Command paths for users are not always the same and depends on the users PATH. Some distributions symlink **/bin** to **/usr/bin** in which case it depends on the order they appear in the PATH environment variable.

The context is the security (SELinux) context of a running application or service. To get the context of a running application use **ps -e --context**.

Warning: If the context is unconfined, then this will open access for more than the desired application.

The lockdown whitelist entries are checked in the following order:

1. *`context`*
2. *`uid`*
3. *`user`*
4. *`command`*

[`--permanent`] `--list-lockdown-whitelist-commands`
List all command lines that are on the whitelist.

[`--permanent`] `--add-lockdown-whitelist-command`=*`command`*
Add the *`command`* to the whitelist.

[`--permanent`] `--remove-lockdown-whitelist-command`=*`command`*
Remove the *`command`* from the whitelist.

[`--permanent`] `--query-lockdown-whitelist-command`=*`command`*
Query whether the *`command`* is on the whitelist. Returns 0 if true, 1 otherwise.

[`--permanent`] `--list-lockdown-whitelist-contexts`
List all contexts that are on the whitelist.

[`--permanent`] `--add-lockdown-whitelist-context` = *`context`*
Add the context *`context`* to the whitelist.

[`--permanent`] `--remove-lockdown-whitelist-context` = *`context`*
Remove the *`context`* from the whitelist.

[`--permanent`] `--query-lockdown-whitelist-context` = *`context`*
Query whether the *`context`* is on the whitelist. Returns 0 if true, 1 otherwise.

[`--permanent`] `--list-lockdown-whitelist-uids`
List all user ids that are on the whitelist.

[`--permanent`] `--add-lockdown-whitelist-uid` = *`uid`*
Add the user id *`uid`* to the whitelist.

[`--permanent`] `--remove-lockdown-whitelist-uid` = *`uid`*
Remove the user id *`uid`* from the whitelist.

[`--permanent`] `--query-lockdown-whitelist-uid` = *`uid`*
Query whether the user id *`uid`* is on the whitelist. Returns 0 if true, 1 otherwise.

[`--permanent`] `--list-lockdown-whitelist-users`
List all user names that are on the whitelist.

[`--permanent`] `--add-lockdown-whitelist-user` = *`user`*
Add the user name *`user`* to the whitelist.

[`--permanent`] `--remove-lockdown-whitelist-user` = *`user`*
Remove the user name *`user`* from the whitelist.

[`--permanent`] `--query-lockdown-whitelist-user` = *`user`*
Query whether the user name *`user`* is on the whitelist. Returns 0 if true, 1 otherwise.

Panic Options

`--panic-on`
Enable panic mode. All incoming and outgoing packets are dropped, active connections will expire. Enable this only if there are serious problems with your network environment. For example if the machine is getting hacked in.

This is a runtime only change.

`--panic-off`
Disable panic mode. After disabling panic mode established connections might work again, if panic mode was enabled for a short period of time.

This is a runtime only change.

`--query-panic`
Returns 0 if panic mode is enabled, 1 otherwise.

Examples

For more examples see http://fedoraproject.org/wiki/FirewallD

Example 1

Enable http service in default zone. This is runtime only change, i.e. effective until restart.

```
firewall-cmd --add-service=http
```

Example 2

Enable port 443/tcp immediately and permanently in default zone. To make the change effective immediately and also after restart we need two commands. The first command makes the change in runtime configuration, i.e. makes it effective immediately, until restart. The second command makes the change in permanent configuration, i.e. makes it effective after restart.

```
firewall-cmd --add-port=443/tcp
firewall-cmd --permanent --add-port=443/tcp
```

Exit Codes

On success 0 is returned. On failure the output is red colored and exit code is either 2 in case of wrong command-line option usage or one of the following error codes in other cases:

String	Code
ALREADY_ENABLED	11
NOT_ENABLED	12
COMMAND_FAILED	13
NO_IPV6_NAT	14
PANIC_MODE	15
ZONE_ALREADY_SET	16
UNKNOWN_INTERFACE	17
ZONE_CONFLICT	18
BUILTIN_CHAIN	19
EBTABLES_NO_REJECT	20
NOT_OVERLOADABLE	21
NO_DEFAULTS	22
BUILTIN_ZONE	23
BUILTIN_SERVICE	24
BUILTIN_ICMPTYPE	25
NAME_CONFLICT	26
NAME_MISMATCH	27
PARSE_ERROR	28
ACCESS_DENIED	29
UNKNOWN_SOURCE	30
RT_TO_PERM_FAILED	31
IPSET_WITH_TIMEOUT	32
BUILTIN_IPSET	33
ALREADY_SET	34
MISSING_IMPORT	35
DBUS_ERROR	36
BUILTIN_HELPER	37
NOT_APPLIED	38
INVALID_ACTION	100
INVALID_SERVICE	101
INVALID_PORT	102
INVALID_PROTOCOL	103
INVALID_INTERFACE	104
INVALID_ADDR	105
INVALID_FORWARD	106
INVALID_ICMPTYPE	107
INVALID_TABLE	108
INVALID_CHAIN	109
INVALID_TARGET	110
INVALID_IPV	111
INVALID_ZONE	112

INVALID_PROPERTY	113
INVALID_VALUE	114
INVALID_OBJECT	115
INVALID_NAME	116
INVALID_FILENAME	117
INVALID_DIRECTORY	118
INVALID_TYPE	119
INVALID_SETTING	120
INVALID_DESTINATION	121
INVALID_RULE	122
INVALID_LIMIT	123
INVALID_FAMILY	124
INVALID_LOG_LEVEL	125
INVALID_AUDIT_TYPE	126
INVALID_MARK	127
INVALID_CONTEXT	128
INVALID_COMMAND	129
INVALID_USER	130
INVALID_UID	131
INVALID_MODULE	132
INVALID_PASSTHROUGH	133
INVALID_MAC	134
INVALID_IPSET	135
INVALID_ENTRY	136
INVALID_OPTION	137
INVALID_HELPER	138
INVALID_PRIORITY	139
INVALID_POLICY	140
INVALID_LOG_PREFIX	141
INVALID_NFLOG_GROUP	142
INVALID_NFLOG_QUEUE	143
MISSING_TABLE	200
MISSING_CHAIN	201
MISSING_PORT	202
MISSING_PROTOCOL	203
MISSING_ADDR	204
MISSING_NAME	205
MISSING_SETTING	206
MISSING_FAMILY	207
RUNNING_BUT_FAILED	251
NOT_RUNNING	252
NOT_AUTHORIZED	253
UNKNOWN_ERROR	254

Note that return codes of **--query-*** options are special: Successful queries return 0, unsuccessful ones return 1 unless an error occurred in which case the table above applies.

See Also

firewall-applet(1), firewalld(1), firewall-cmd(1), firewall-config(1), firewalld.conf(5), firewalld.direct(5), firewalld.dbus(5), firewalld.icmptype(5), firewalld.lockdown-whitelist(5), firewall-offline-cmd(1), firewalld.richlanguage(5), firewalld.service(5), firewalld.zone(5), firewalld.zones(5), firewalld.policy(5), firewalld.policies(5), firewalld.ipset(5), firewalld.helper(5)

Notes

firewalld home page:
http://firewalld.org

More documentation with examples:
http://fedoraproject.org/wiki/FirewallD