Using firewalld :: Fedora Docs

Using firewalld

What is firewalld?

A *firewall* is a way to protect machines from any unwanted traffic from outside. It enables users to control incoming network traffic on host machines by defining a set of *firewall rules*. These rules are used to sort the incoming traffic and either block it or allow through.

`firewalld` is a firewall service daemon that provides a dynamic customizable host-based firewall with a `D-Bus` interface. Being dynamic, it enables creating, changing, and deleting the rules without the necessity to restart the firewall daemon each time the rules are changed.

`firewalld` uses the concepts of *zones* and *services*, that simplify the traffic management.

`Zones` are predefined sets of rules. Network interfaces and sources can be assigned to a zone. The traffic allowed depends on the network your computer is connected to and the security level this network is assigned. Firewall services are predefined rules that cover all necessary settings to allow incoming traffic for a specific service and they apply within a zone.

`Services` use one or more ports or addresses for network communication. Firewalls filter communication based on ports. To allow network traffic for a service, its ports must be open. `firewalld` blocks all traffic on ports that are not explicitly set as open. Some zones, such as trusted, allow all traffic by default.

Checking the firewalld status

Viewing the current status of `firewalld`

The firewall service, `firewalld`, is installed on the system by default. Use the `firewalld` CLI interface to check that the service is running.

To see the status of the service:

```
$ sudo firewall-cmd --state
```

For more information about the service status, use the `systemctl status` sub-command:

```
$ sudo systemctl status firewalld
firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor pr
   Active: active (running) since Mon 2017-12-18 16:05:15 CET; 50min ago
     Docs: man:firewalld(1)
 Main PID: 705 (firewalld)
    Tasks: 2 (limit: 4915)
   CGroup: /system.slice/firewalld.service
```

```
└─705 /usr/bin/python3 -Es /usr/sbin/firewalld --nofork --nopid
```

Furthermore, it is important to know how `firewalld` is set up and which rules are in force before you try to edit the settings. To display the firewall settings, see Viewing current firewalld settings

Viewing current firewalld settings

Viewing allowed services using GUI

To view the list of services using the graphical **firewall-config** tool, press the `Super` key to enter the Activities Overview, type `firewall`, and press `Enter`. The **firewall-config** tool appears. You can now view the list of services under the `Services` tab.

Alternatively, to start the graphical firewall configuration tool using the command-line, enter the following command:

The `Firewall Configuration` window opens. Note that this command can be run as a normal user, but you are prompted for an administrator password occasionally.

Viewing firewalld settings using CLI

With the CLI client, it is possible to get different views of the current firewall settings. The `--list-all` option shows a complete overview of the `firewalld` settings.

`firewalld` uses zones to manage the traffic. If a zone is not specified by the `--zone` option, the command is effective in the default zone assigned to the active network interface and connection.

To list all the relevant information for the default zone:

```
$ firewall-cmd --list-all
public
  target: default
  icmp-block-inversion: no
  interfaces:
  sources:
  services: ssh dhcpv6-client
  ports:
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

> To specify the zone for which to display the settings, add the `--zone=`*`zone-name`* argument to the `firewall-cmd --list-all` command, for example:
>
> ```
> ~]# firewall-cmd --list-all --zone=home
> home
>   target: default
>   icmp-block-inversion: no
>   interfaces:
>   sources:
>   services: ssh mdns samba-client dhcpv6-client
> ```

```
... [output truncated]
```

To see the settings for particular information, such as services or ports, use a specific option. See the `firewalld` manual pages or get a list of the options using the command help:

```
$ firewall-cmd --help

Usage: firewall-cmd [OPTIONS...]

General Options
  -h, --help           Prints a short help text and exists
  -V, --version        Print the version string of firewalld
  -q, --quiet          Do not print status messages

Status Options
  --state              Return and print firewalld state
  --reload             Reload firewall and keep state information
... [output truncated]
```

For example, to see which services are allowed in the current zone:

```
$ firewall-cmd --list-services
samba-client ssh dhcpv6-client
```

Listing the settings for a certain subpart using the CLI tool can sometimes be difficult to interpret. For example, you allow the `SSH` service and `firewalld` opens the necessary port (22) for the service. Later, if you list the allowed services, the list shows the `SSH` service, but if you list open ports, it does not show any. Therefore, it is recommended to use the `--list-all` option to make sure you receive a complete information.

Installing firewalld

Install firewalld:

1. Run this command on the command line:

```
sudo dnf install firewalld
```

Starting firewalld

Start firewalld, by entering the following commands:

```
$ sudo systemctl unmask firewalld
$ sudo systemctl start firewalld
```

To make firewalld start automatically at system start:

```
$ sudo systemctl enable firewalld
```

Stopping firewalld

To stop firewalld, enter the following command as root:

```
$ sudo systemctl stop firewalld
```

Prevent firewalld from starting automatically at system start, enter the following command as root:

```
$ sudo systemctl disable firewalld
```

Make sure firewalld is not started by accessing the firewalld D-Bus interface and also if other services require firewalld, enter the following command as root:

```
$ sudo systemctl mask firewalld
```

Runtime and permanent settings

Any changes made while firewalld is running will be lost when firewalld is restarted. When firewalld is restarted, the settings revert to their permanent values.

These changes are said to be made in *runtime mode*.

To make the changes persistent across reboots, apply them again using the `--permanent` option. Alternatively, to make changes persistent while firewalld is running, use the `--runtime-to-permanent` *`firewall-cmd`* option.

If you make changes while firewalld is running using only the `--permanent` option, they do not become effective until firewalld is restarted. However, restarting firewalld briefly stops the networking traffic, causing disruption to your system.

Changing settings in runtime and permanent configuration using CLI

Using the CLI, you can only modify either runtime or permanent mode. To modify the firewall settings in permanent mode, use the `--permanent` option with the `firewall-cmd` command.

```
$ sudo firewall-cmd --permanent <other options>
```

Without this option, the command modifies runtime mode. To change settings in both modes, you can use two methods:

- Change runtime settings and then make them permanent as follows:
    1. Change the runtime settings:

       ```
       firewall-cmd <other options>
       ```

    2. Use `--runtime-to-permanent` to make the changes permanent.

       ```
       firewall-cmd --runtime-to-permanent
       ```

- Set permanent settings and reload the settings into runtime mode:
    1. Make the changes in permanent mode:

       ```
       firewall-cmd --permanent <other options>
       ```

    2. Reload the settings:

```
firewall-cmd --reload
```

The first method allows you to test the settings before you apply them to permanent mode.

> It is possible that an incorrect setting will result in a user locking themselves out of a machine. To prevent this, use the `--timeout` option. Using this option means that after a specified amount of time, any change reverts to its previous state. You can not use the `--permanent` option with the `--timeout` option.
>
> For example, to add the SSH service for 15 minutes use this command:
>
> ```
> $ sudo firewall-cmd --add-service=ssh --timeout 15m
> ```
>
> The SSH service will be available until access is removed after 15 minutes.

Controlling ports using firewalld

What are ports?

Ports are logical devices that enable an operating system to receive and distinguish network traffic and forward it accordingly to system services. These are usually represented by a daemon that listens on the port, that is it waits for any traffic coming to this port.

Normally, system services listen on standard ports that are reserved for them. The httpd daemon, for example, listens on port 80. However, system administrators may configure daemons to listen on different ports to enhance security.

Opening a port

Through open ports, the system is accessible from the outside, which represents a security risk. Generally, keep ports closed and only open them if they are required for certain services.

Opening a port using the command line

1. Get a list of allowed ports in the current zone:

   ```
   $ firewall-cmd --list-ports
   ```

2. Add a port to the allowed ports to open it for incoming traffic:

   ```
   $ sudo firewall-cmd --add-port=port-number/port-type
   ```

3. Make the new settings persistent:

   ```
   $ sudo firewall-cmd --runtime-to-permanent
   ```

The port types are either tcp, udp, sctp, or dccp. The type must match the type of network communication.

Closing a port

When an open port is no longer needed, close that port in firewalld. It is highly recommended to close all unnecessary ports as soon as they are not used because leaving a port open represents a security risk.

Closing a port using the command line

To close a port, remove it from the list of allowed ports:

1. List all allowed ports:

   ```
   $ firewall-cmd --list-ports
   ```

   > This command will only give you a list of ports that have been opened as ports. You will not be able to see any open ports that have been opened as a service. Therefore, you should consider using the --list-all option instead of --list-ports.

2. Remove the port from the allowed ports to close it for the incoming traffic:

   ```
   $ sudo firewall-cmd --remove-port=port-number/port-type
   ```

3. Make the new settings persistent:

   ```
   $ sudo firewall-cmd --runtime-to-permanent
   ```