

# Using the D-Bus Interface to Firewalld

**Finnbarr P. Murphy**

(fpm@fpmurphy.com)

[Firewalld](#), a dynamic zone-based firewall daemon, has been under development since circa 2009, with the latest version - [firewalld](#) 0.6.3 - released on Oct 11, 2018. The main developer is Thomas Wörner who currently works for Red Hat. It became the default firewall mechanism in Fedora 18 and, subsequently, in RHEL7 and CentOS 7.

*Firewalld* has many advantages over the older *iptables* mechanism. Of note, it solved the problem whereby *iptables* required a firewall restart at each change thus breaking any stateful connections. It also provides a rich well-thought-out range of [D-Bus](#) methods, signals and properties.

This post assumes you are familiar with *firewalld* and [D-Bus](#) concepts and operation. I focus instead on how to use the rich (as in extensive) *firewalld* D-Bus interface to retrieve information or change settings.

*Firewalld* is configured as a *systemd* D-Bus service. Note the "Type=dbus" directive below.

```
# cat /usr/lib/systemd/system/firewalld.service
[Unit]
Description=firewalld - dynamic firewall daemon
Before=network.target
Before=libvirtd.service
Before=NetworkManager.service
Conflicts=iptables.service ip6tables.service ebtables.service

[Service]
EnvironmentFile=-/etc/sysconfig/firewalld
ExecStart=/usr/sbin/firewalld --nofork --nopid $FIREWALLD_ARGS
ExecReload=/bin/kill -HUP $MAINPID
# suppress to log debug and error output also to /var/log/messages
StandardOutput=null
StandardError=null
Type=dbus
BusName=org.fedoraproject.FirewallD1

[Install]
WantedBy=basic.target
Alias=dbus-org.fedoraproject.FirewallD1.service
```

Now that we know that the *firewalld* service is based on D-Bus, we need to find the D-Bus interface used by *firewalld*:

```
# dbus-send --system --dest=org.freedesktop.DBus --type=method_call --print-reply \
/org/freedesktop/DBus org.freedesktop.DBus.ListNames | grep FirewallD

string "org.fedoraproject.FirewallD1"
#
```

This shows that the interface is *org.fedoraproject.FirewallD1*.

Next we want to see what methods, properties and signals are supported by this D-Bus interface. To do this, we use one of the cooler features of D-Bus, i.e. introspection.

```
# dbus-send --system --dest=org.fedoraproject.FirewallD1 --print-reply \  
/org/fedoraproject/FirewallD1 org.freedesktop.DBus.Introspectable.Introspect  
  
method return sender=:1.7 -> dest=:1.116 reply_serial=2  
string "
```







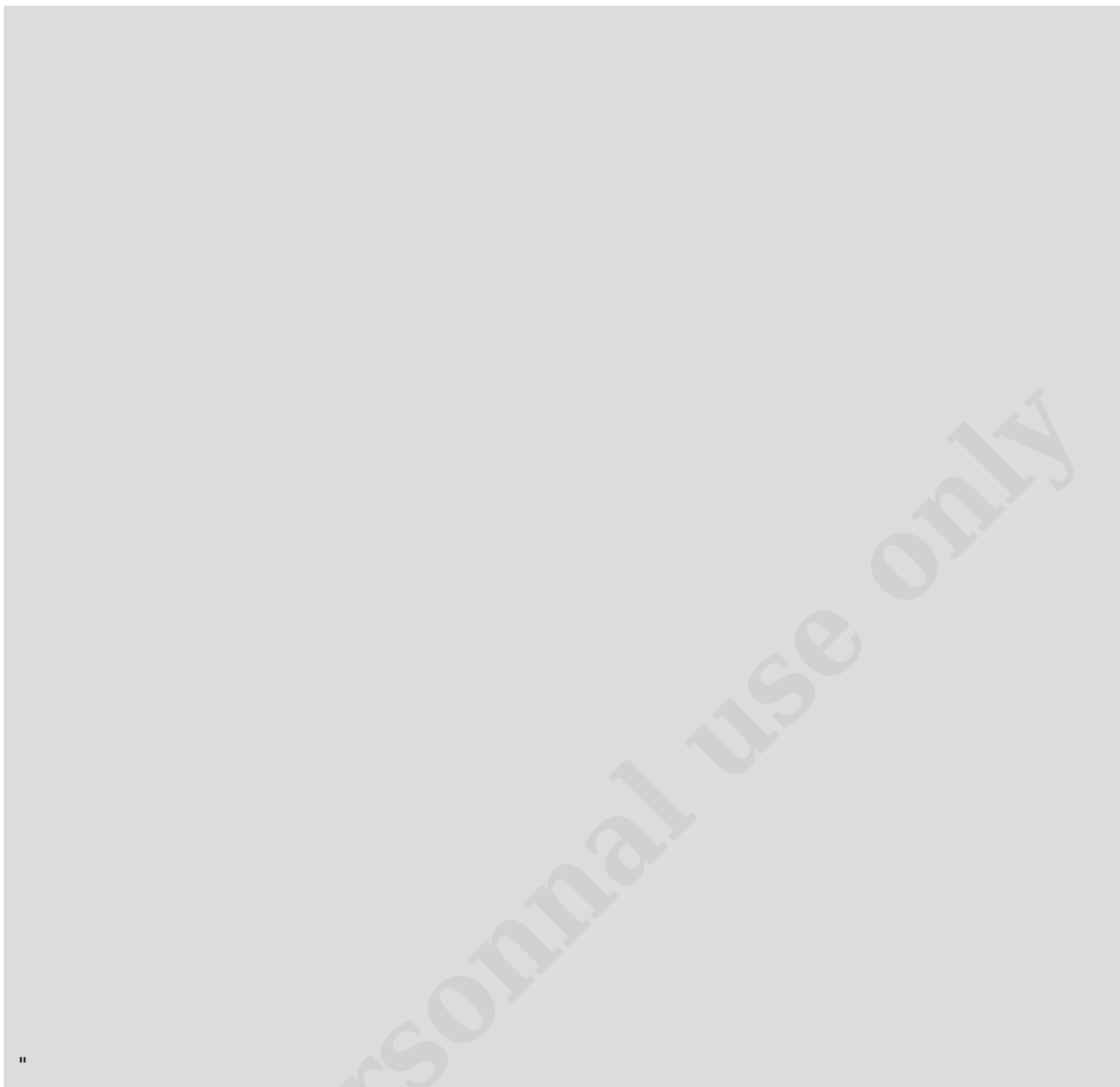












The above output lists all the methods, signals and properties available via the *firewalld* D-Bus interface. This is the standard D-Bus introspection output format based on the D-Bus DTD. All D-Bus services are required to implement the *org.freedesktop.DBus.Introspectable.Introspect* method.

With this extensive *firewalld* D-Bus interface, it is simple for services, applications and users to manage firewall settings. The interface is used by all the firewall configuration tools such as *firewall-cmd*, *firewall-config* and *firewall-applet*.

Turning now to our first example. Consider the following simple *firewall-cmd* which retrieves and prints the default zone, i.e. *public*.

```
# firewall-cmd --get-default-zone
public
#
```

Here is how to retrieve the same information using the *firewalld* D-Bus interface and *dbus-send*:

```
# dbus-send --system --dest=org.fedoraproject.FirewallD1 --print-reply --type=method_call
```

```

\
/org/fedoraproject/FirewallD1 org.fedoraproject.FirewallD1.getDefaultZone

method return sender=:1.7 -> dest=:1.133 reply_serial=2
  string "public"
#

```

The *dbus-send* command is used to send a message to a D-Bus message bus and display the result of that message. There are two well-known message buses: the systemwide message bus (option *-system*) and the per-user-login-session message (option *-session*). With *firewalld*, we use the system bus to talk to its interface. Also, nearly all uses of *dbus-send* must provide the *-dest* argument which is the name of an interface (connection) on the bus to send the message to. In our case, it is *org.fedoraproject.FirewallD1*. The object path and the name of the message to send must always be specified. Following arguments, if any, are the message contents (message arguments). These are given as type-specified values.

Here we use both *firewall-cmd* and *dbus-send* to retrieve a list of zones:

```

# firewall-cmd --get-zones
block dmz drop external home internal public trusted work

# dbus-send --system --dest=org.fedoraproject.FirewallD1 --print-reply --type=method_call
\
/org/fedoraproject/FirewallD1 org.fedoraproject.FirewallD1.zone.getZones

method return sender=:1.7 -> dest=:1.363 reply_serial=2
  array [
    string "block"
    string "dmz"
    string "drop"
    string "external"
    string "home"
    string "internal"
    string "public"
    string "trusted"
    string "work"
  ]
#

```

In the following example we use *firewall-cmd* and *dbus-send* to retrieve information about a zone:

```

# firewall-cmd --zone=public --list-all

public (default)
  interfaces:
  sources:
  services: dhcpv6-client ssh
  ports:
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:

# dbus-send --system --dest=org.fedoraproject.FirewallD1 --print-reply --type=method_call
\
/org/fedoraproject/FirewallD1 org.fedoraproject.FirewallD1.getZoneSettings string:"public"

method return sender=:1.7 -> dest=:1.351 reply_serial=2
  struct {
    string ""
    string "Public"
  }
#

```

```

    string "For use in public areas. You do not trust the other computers on networks to
    not harm your computer. Only selected incoming connections are accepted."
    boolean false
    string "{chain}_{zone}"
    array [
        string "dhcpv6-client"
        string "ssh"
    ]
    array [
    ]
    array [
    ]
    boolean false
    array [
    ]
    array [
    ]
    array [
    ]
    array [
    ]
}
#

```

In the following example we retrieve all the properties of the *org.fedoraproject.FirewallD1* interface:

```

# dbus-send --system --print-reply --dest=org.fedoraproject.FirewallD1 /org/fedoraproject/
FirewallD1 \
org.freedesktop.DBus.Properties.GetAll string:"org.fedoraproject.FirewallD1"

method return sender=:1.7 -> dest=:1.311 reply_serial=2
array [
  dict entry(
    string "BRIDGE"
    variant          boolean true
  )
  dict entry(
    string "interface_version"
    variant          string "1.1"
  )
  dict entry(
    string "IPv6_rpfilter"
    variant          boolean true
  )
  dict entry(
    string "state"
    variant          string "RUNNING"
  )
  dict entry(
    string "version"
    variant          string "0.3.9"
  )
  dict entry(
    string "IPv4"
    variant          boolean true
  )
  dict entry(
    string "IPv6"
    variant          boolean true
  )
]

```

In this example we retrieve individual properties of the *org.fedoraproject.FirewallD1* interface:

```
# dbus-send --system --print-reply --dest=org.fedoraproject.FirewallD1 \
/org/fedoraproject/FirewallD1 org.freedesktop.DBus.Properties.Get \
string:"org.fedoraproject.FirewallD1" string:"version"

method return sender=:1.7 -> dest=:1.328 reply_serial=2
  variant      string "0.3.9"

# dbus-send --system --print-reply --dest=org.fedoraproject.FirewallD1 \
/org/fedoraproject/FirewallD1 org.freedesktop.DBus.Properties.Get \
string:"org.fedoraproject.FirewallD1" string:"interface_version"

method return sender=:1.7 -> dest=:1.329 reply_serial=2
  variant      string "1.1"

# dbus-send --system --print-reply --dest=org.fedoraproject.FirewallD1 \
/org/fedoraproject/FirewallD1 org.freedesktop.DBus.Properties.Get \
string:"org.fedoraproject.FirewallD1" string:"state"

method return sender=:1.7 -> dest=:1.330 reply_serial=2
  variant      string "RUNNING"

# dbus-send --system --print-reply=literal --dest=org.fedoraproject.FirewallD1 \
/org/fedoraproject/FirewallD1 org.freedesktop.DBus.Properties.Get \
string:"org.fedoraproject.FirewallD1" string:"state"

variant      RUNNING
```

The above output shows that RHEL 7.2 uses version *firewalld* release 0.3.9 – which is quite an old version of *firewalld*.

The following example shows you how to use *dbus-send* to get (view) and set (change) the default zone:

```
# dbus-send --system --dest=org.fedoraproject.FirewallD1 --print-reply --type=method_call \
/org/fedoraproject/FirewallD1 org.fedoraproject.FirewallD1.getDefaultZone

method return sender=:1.7 -> dest=:1.133 reply_serial=2
  string "public"

# dbus-send --system --dest=org.fedoraproject.FirewallD1 --print-reply --type=method_call \
/org/fedoraproject/FirewallD1 org.fedoraproject.FirewallD1.setDefaultZone string:"publiciii
ii"

Error org.freedesktop.DBus.Python.dbus.exceptions.DBusException: INVALID_ZONE: publiciiii

# dbus-send --system --dest=org.fedoraproject.FirewallD1 --print-reply --type=method_call \
/org/fedoraproject/FirewallD1 org.fedoraproject.FirewallD1.setDefaultZone string:"work"

method return sender=:1.7 -> dest=:1.344 reply_serial=2

# dbus-send --system --dest=org.fedoraproject.FirewallD1 --print-reply --type=method_call \
/org/fedoraproject/FirewallD1 org.fedoraproject.FirewallD1.getDefaultZone

method return sender=:1.7 -> dest=:1.345 reply_serial=2
  string "work"
#
```

Well, time to end this post. The above examples should be sufficient to enable you to gain an

understanding of how use the D-Bus interface to *firewalld*. Whilst I used the *dbus-send* utility, with slight syntax modifications the examples will also work with the Qt *qdbus* utility.

All the examples provided above work on RHEL 7.2 but there is no guarantee that they will work on later releases of RHEL, or downstream distributions, as *firewalld* is still in fairly active development.

Enjoy!

For personal use only