命令行管理防火墙

1、可以通过以下三种方式来管理firewalld:

Ø 使用命令行工具firewall-cmd

Ø 使用图形工具firewall-config

Ø 使用/etc/firewalld/中的配置文件

在大部分情况下，不建议直接编辑配置文件，但是在使用配置管理工具时，以这种方法复制配置会很有用。

在 CentOS7 中，默认安装firewalld 和图形化用户接口配置工具firewall-config。作为 root 用户运行下列命令可以检查：

~]# yum install firewalld firewall-config

要禁用 firewalld，则作为 root 用户运行下列命令：

~]# systemctl disable firewalld.service

~]# systemctl stop firewalld.service

要用iptables和ip6tables服务代替firewalld则以 root 身份运行以下命令，先禁用 firewalld,然后安装 iptables-services程序包,以root身份输入以下命令:

~]# yum install iptables-services

iptables-services 程序包包含了iptables服务和ip6tables服务。然后以 root身份运行 iptables 和 ip6tables 命令：

~]# systemctl start iptables

~]# systemctl start ip6tables

~]# systemctl enable iptables

~]# systemctl enable ip6tables

要启动 firewalld，则以root用户身份输入以下命令：

~]# systemctl start firewalld.service

开机启动firewalld，则以root用户身份输入以下命令：

~]# systemctl enable firewalld.service

如果firewalld在运行，输入以下命令检查：

[root@localhost ~]# systemctl status firewalld.service

● firewalld.service - firewalld - dynamic firewall daemon

Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: enabled)

Active: active (running) since 一 2017-05-15 21:55:26 CST; 12min ago

Docs: man:firewalld(1)

Main PID: 2565 (firewalld)

CGroup: /system.slice/firewalld.service

└─2565 /usr/bin/python -Es /usr/sbin/firewalld --nofork --nopid

另外，还可以通过firewall-cmd命令来连接后台程序进行检查：

[root@localhost ~]# firewall-cmd --state

running

2、命令行工具firewall-cmd支持全部防火墙特性，基本应用如下：

1、 获取firewalld状态

[root@sunday-test ~]# firewall-cmd --state

```
[root@server1 ~]# firewall-cmd --state
running
```

2、在不改变状态的条件下重新加载防火墙：

[root@sunday-test ~]# firewall-cmd --reload

```
[root@server1 ~]# firewall-cmd --reload
success
```

如果你使用--complete-reload，状态信息将会丢失。

3、获取支持的区域列表

[root@sunday-test ~]# firewall-cmd --get-zones

```
[root@server1 ~]# firewall-cmd --get-zones
block dmz drop external home internal public trusted work
```

这条命令输出用空格分隔的列表

4、获取所有支持的服务

[root@sunday-test ~]# firewall-cmd --get-services

```
[root@server1 ~]# firewall-cmd --get-services
amanda-client bacula bacula-client dhcp dhcpv6 dhcpv6-client dns ftp high-availability http https im
aps ipp ipp-client ipsec kerberos kpasswd ldap ldaps libvirt libvirt-tls mdns mountd ms-wbt mysql nf
s ntp openvpn pmcd pmproxy pmwebapi pmwebapis pop3s postgresql proxy-dhcp radius rpc-bind samba samb
a-client smtp ssh telnet tftp tftp-client transmission-client vnc-server wbem-https
```

这条命令输出用空格分隔的列表。

服务是firewalld所使用的有关端口和选项的规则集合。被启动的服务会在firewalld服务开启或者运行时自动加载。默认情况下，很多服务是有效的。使用下面命令可列出有效的服务。

想要列出默认有效的服务，也可以进入下面的目录也能够取得。

cd /usr/lib/firewalld/services/

```
[root@localhost ~]# cd /usr/lib/firewalld/services/
[root@localhost services]# ls
amanda-client.xml      http.xml          libvirt.xml  pmwebapis.xml     ssh.xml
bacula-client.xml      imaps.xml         mdns.xml     pmwebapi.xml      telnet.xml
bacula.xml             ipp-client.xml    mountd.xml   pop3s.xml         tftp-client.xml
dhcpv6-client.xml      ipp.xml           ms-wbt.xml   postgresql.xml    tftp.xml
dhcpv6.xml             ipsec.xml         mysql.xml    proxy-dhcp.xml    transmission-client.xml
dhcp.xml               kerberos.xml      nfs.xml      radius.xml        vnc-server.xml
dns.xml                kpasswd.xml       ntp.xml      rpc-bind.xml      wbem-https.xml
ftp.xml                ldaps.xml         openvpn.xml  samba-client.xml
high-availability.xml  ldap.xml          pmcd.xml     samba.xml
https.xml              libvirt-tls.xml   pmproxy.xml  smtp.xml
[root@localhost services]# cat ssh.xml
<?xml version="1.0" encoding="utf-8"?>
<service>
  <short>SSH</short>
  <description>Secure Shell (SSH) is a protocol for logging into and executing commands on remote ma
chines. It provides secure encrypted communications. If you plan on accessing your machine remotely
via SSH over a firewalled interface, enable this option. You need the openssh-server package install
ed for this option to be useful.</description>
  <port protocol="tcp" port="22"/>
</service>
```

想要创建自己的服务，需要在下面的目录下定义它。比如，现在我想添加一个rhmp服务，端口号1935。首先，任选一个服务复制过来。

```
[root@localhost ~]# cp /usr/lib/firewalld/services/ssh.xml /etc/firewalld/services/
[root@localhost ~]# cd /etc/firewalld/services/
[root@localhost services]# ls -l
total 4
-rw-r-----. 1 root root 463 Apr 23 23:04 ssh.xml
```

接下来把复制过来的文件重命名为“rtmp.xml”，

接下来打开并编辑文件的头部、描述、协议和端口号，以供RTMP服务使用，如下图所示。

```
<?xml version="1.0" encoding="utf-8"?>
<service>
  <short>rtmp</short>
  <description>to allow rtmp service</description>
  <port protocol="tcp" port="1935"/>
</service>
```

重启firewalld服务或者重新加载设置，以激活这些设置。

firewall-cmd --reload

为确认服务是否已经启动，运行下面的命令获取有效的服务列表。

firewall-cmd --get-services

```
[root@localhost services]# firewall-cmd --get-services
amanda-client bacula bacula-client dhcp dhcpv6 dhcpv6-client dns ftp high-availability http https im
aps ipp ipp-client ipsec kerberos kpasswd ldap ldaps libvirt libvirt-tls mdns mountd ms-wbt mysql nf
s ntp openvpn pmcd pmproxy pmwebapi pmwebapis pop3s postgresql proxy-dhcp radius rpc-bind rtmp samba
 samba-client smtp ssh telnet tftp tftp-client transmission-client vnc-server wbem-https
```

5、获取所有支持的ICMP类型

[root@sunday-test services]# firewall-cmd --get-icmptypes

```
[root@server1 ~]# firewall-cmd --get-icmptypes
destination-unreachable echo-reply echo-request parameter-problem redirect router-advertisement rout
er-solicitation source-quench time-exceeded
```

这条命令输出用空格分隔的列表。

6、列出全部启用的区域的特性（即查询当前防火墙策略）

[root@sunday-test services]# firewall-cmd --list-all-zones

解释：特性可以是定义的防火墙策略，如：服务、端口和协议的组合、端口/数据报转发、伪装、ICMP 拦截或自定义规则等

```
[root@server1 ~]# firewall-cmd --list-all-zones
block
  interfaces:
  sources:
  services:
  ports:
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:

dmz
  interfaces:
  sources:
  services: ssh
  ports:
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:
```

上面的命令将会列出每种区域如block、dmz、drop、external、home、internal、public、trusted以及work。如果区域还有其它详细规则（rich-rules）、启用的服务或者端口，这些区域信息也会分别被罗列出来

7、输出区域全部启用的特性。如果省略区域，将显示默认区域的信息。

firewall-cmd [--zone=] --list-all

```
[root@server1 ~]# firewall-cmd --list-all
trusted (default, active)
  interfaces: eno33554992
  sources:
  services:
  ports:
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:
```

输出指定区域启动的特性

[root@sunday-test services]# firewall-cmd --list-all --zone=public

```
[root@server1 ~]# firewall-cmd --zone=public --list-all
public (active)
  interfaces: eno16777736
  sources:
  services: dhcpv6-client ssh
  ports:
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:
```

8、查看默认区域

[root@sunday-test services]# firewall-cmd --get-default-zone

```
[root@server1 ~]# firewall-cmd --get-default-zone
trusted
```

public区域是默认区域。

在文件/etc/firewalld/firewalld.conf中定义成DefaultZone=public。

9、设置默认区域

firewall-cmd --set-default-zone=区域名

```
[root@server1 ~]# firewall-cmd --set-default-zone=drop
success
[root@server1 ~]# firewall-cmd --get-default-zone
drop
```

流入默认区域中配置的接口的新访问请求将被置入新的默认区域。当前活动的连接将不受影响。

10、获取活动的区域

[root@sunday-test ~]# firewall-cmd --get-active-zones

```
[root@server1 ~]# firewall-cmd --get-active-zones
drop
  interfaces: eno33554992
public
  interfaces: eno16777736
```

这条命令将用以下格式输出每个区域所含接口：

区域名

interfaces： 接口名

11、根据接口获取区域即需要查看哪个区域和这个接口绑定即查看某个接口是属于哪个zone的：

firewall-cmd --get-zone-of-interface=接口名

```
[root@server1 ~]# firewall-cmd --get-zone-of-interface=eno16777736
public
```

这条命令将输出接口所属的区域名称。

12、将接口（网卡）增加到区域

firewall-cmd [--zone=] --add-interface=接口名

```
[root@server1 ~]# firewall-cmd --add-interface=eno16777736
success
```

如果接口不属于区域，接口将被增加到区域。如果区域被省略了，将使用默认区域。接口在重新加载后将重新应用。

13、修改接口所属区域

firewall-cmd [--zone=] --change-interface=接口名

```
[root@server1 ~]# firewall-cmd --zone=trusted --change-interface=eno16777736
success
```

这个选项与 --add-interface 选项相似，但是当接口已经存在于另一个区域的时候，该接口将被添加到新的区域。

14、从区域中删除一个接口

firewall-cmd [--zone=] --remove-interface=接口名

```
[root@server1 ~]# firewall-cmd --get-active-zones
drop
  interfaces: eno33554992
trusted
  interfaces: eno16777736
[root@server1 ~]# firewall-cmd --zone=drop --remove-interface=eno33554992
success
```

注：如果某个接口不属于任何Zone，那么这个接口的所有数据包使用默认的Zone的规则

15、查询接口是否属于一个区域

firewall-cmd [--zone=] --query-interface=接口名

```
[root@localhost ~]# firewall-cmd --query-interface=eno16777736
yes
```

如果区域被省略了，将使用默认区域

16、列举区域中启用的服务

firewall-cmd [--zone=] --list-services

```
[root@localhost ~]# firewall-cmd --list-services
dhcpv6-client ssh
```

如果区域被省略了，将使用默认区域

查看home区域中启用服务

[root@sunday-test ~]# firewall-cmd --list-services --zone=home

```
[root@localhost ~]# firewall-cmd --list-services --zone=home
dhcpv6-client ipp-client mdns samba-client ssh
```

17、启用应急模式阻断所有网络连接，以防出现紧急状况

[root@sunday-test ~]# firewall-cmd --panic-on

```
[root@localhost ~]# firewall-cmd --panic-on
success
```

18、禁用应急模式

firewall-cmd --panic-off

19、查询应急模式

firewall-cmd --query-panic

其他相关的配置项可以查看firewall-cmd的手册页：#man firewall-cmd

课后练习题：

1、打开一个虚拟机，查看firewalld防火墙的默认区域是哪个区域？

2、将防火墙默认区域改成work区域

3、将虚拟机的网卡添加到work区域

4、查看work区域的防火墙规则，并截图列出

5、使用真实机ping虚拟机的IP地址，发现会可以通信

6、配置启用虚拟机应急模式，阻断所有连接，会发现不再能ping通

7、关闭应急模式，重新可以ping通