Hi Linux

CentOS 7下使用FirewallD构建动态防火墙

2016-05-26 | FirewallD | 59

FirewallD介绍

FirewallD提供了支持网络/防火墙区域(zone)定义网络链接以及接口安全等级的动态防火墙管理工具。它支持IPv4, IPv6 防火墙设置以及以太网桥接，也支持允许服务或者应用程序直接添加防火墙规则的接口。FirewallD拥有运行时配置和永久配置选项。

采用 `firewall-cmd` (command)或 `firewall-config` (gui)来动态的管理kernel netfilter的临时或永久的接口规则，并实时生效而无需重启服务。

FirewallD特性

- Zone

FirewallD使用区域(zone)的概念来管理，网络区域定义了网络连接的可信等级。这是一个一对多的关系，这意味着一次连接可以仅仅是一个区域的一部分，而一个区域可以用于很多连接。每个网卡对应一个zone，这些zone的配置文件可在 `/usr/lib/firewalld/zones/` 下看到，默认的是public。

Zone提供了以下几个区域

> drop
>
> 任何流入网络的包都被丢弃，不作出任何响应，只允许流出的网络连接。即使开放了某些服务(比如http)，这些服务的数据也是不允许通过的。
>
> block
> 任何进入的网络连接都被拒绝，并返回IPv4的icmp-host-prohibited报文或者IPv6的icmp

> 6-adm-prohibited报文。只允许由该系统初始化的网络连接。
>
> public(默认)
> 用以可以公开的部分。你认为网络中其他的计算机不可信并且可能伤害你的计算机，只允许选中的服务通过。
>
> external
> 用在路由器等启用伪装的外部网络。你认为网络中其他的计算机不可信并且可能伤害你的计算机，只允许选中的服务通过。
>
> dmz
> 用以允许隔离区(dmz)中的电脑有限地被外界网络访问，只允许选中的服务通过。
>
> work
> 用在工作网络。你信任网络中的大多数计算机不会影响你的计算机，只允许选中的服务通过。
>
> home
> 用在家庭网络。你信任网络中的大多数计算机不会影响你的计算机，只允许选中的服务通过。
>
> internal
> 用在内部网络。你信任网络中的大多数计算机不会影响你的计算机，只允许选中的服务通过。
>
> trusted
> 允许所有网络连接，即使没有开放任何服务，那么使用此zone的流量照样通过(一路绿灯)。

- 预定义的服务

服务是端口和/或协议入口的组合。备选内容包括netfilter助手模块以及 IPv4、IPv6地址。

- 端口和协议

定义了tcp或udp端口，端口可以是一个端口或者端口范围。

- ICMP阻塞

可以选择Internet控制报文协议的报文。这些报文可以是信息请求亦可是对信息请求或错误条件创建的响应。

- 伪装

私有网络地址可以被映射到公开的IP地址。这是一次正规的地址转换。

- 端口转发

端口可以映射到另一个端口以及/或者其他主机。

FirewallD安装

安装FirewallD和管理工具

```
$ yum install firewalld firewall-config
```

运行、停止、禁用FirewallD

```
$ systemctl start firewalld          # 启动
$ systemctl enable firewalld         # 开机启动
$ systemctl stop firewalld           # 关闭
$ systemctl disable firewalld        # 取消开机启动
$ systemctl status firewalld         # 查看状态
```

使用FirewallD

FirewallD规则管理可以直接修改配置文件(`/etc/firewalld/firewalld.conf`)进行配置,也可以通过图形界面工具 `firewall-config` 或者命令行客户端 `firewall-cmd` 来启用或者关闭防火墙特性。手动编辑配置文件相对还是比较麻烦，推荐使用工具进行配置。

一般应用

查看规则

```
$ firewall-cmd --help
```

查看运行状态

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

查看已被激活的Zone信息

```
$ firewall-cmd --get-active-zones
public
  interfaces: eth0 eth1
```

获取活动的区域

```
$ firewall-cmd --get-active-zones
```

查看指定接口的Zone信息

```
$ firewall-cmd --get-zone-of-interface=eth0
```

这条命令将输出接口所属的区域名称。

查看指定级别的接口

```
$ firewall-cmd --zone=public --list-interfaces
eth0
```

获取支持的区域列表

```
$ firewall-cmd --get-zones
```

查看指定级别的所有信息，譬如public

```
$ firewall-cmd --zone=public --list-all
public (default, active)
  interfaces: eth0
  sources:
  services: dhcpv6-client http ssh
  ports:
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:
```

查看所有级别被允许的服务

```
$ firewall-cmd --get-service
```

查看重启后所有Zones级别中被允许的服务，即永久放行的服务

```
$ firewall-cmd --get-service --permanent
```

更新规则，不重启服务

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

更新规则，重启服务。状态信息将会丢失。

```
$ firewall-cmd --complete-reload
```

这个选项应当仅用于处理防火墙问题时，例如，状态信息和防火墙规则都正常，但是不能建立任何连接的情况。

添加某接口至某信任等级，譬如添加eth0至public，再永久生效

```
$ firewall-cmd --zone=public --add-interface=eth0 --permanent
```

修改接口所属区域

```
1   $ firewall-cmd [--zone=<zone>] --change-interface=<interface>
```

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。

将接口增加到区域

```
1   $ firewall-cmd [--zone=<zone>] --add-interface=<interface>
```

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。

设置public为默认的信任级别

```
1   $ firewall-cmd --set-default-zone=public
```

设置默认区域

```
1   $ firewall-cmd --set-default-zone=<zone>
```

获取所有支持的服务

```
1   $ firewall-cmd --get-services
```

获取所有支持的ICMP类型

```
1   $ firewall-cmd --get-icmptypes
```

列出全部启用的区域的特性

```
$ firewall-cmd --list-all-zones
```

输出格式是

```
<zone>
  interfaces: <interface1> ..
  services: <service1> ..
  ports: <port1> ..
  forward-ports: <forward port1> ..
  icmp-blocks: <icmp type1> ..
```

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=<zone>] --list-all
```

获取默认区域的网络设置

```
$ firewall-cmd --get-default-zone
```

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。

从区域中删除一个接口

```
$ firewall-cmd [--zone=<zone>] --remove-interface=<interface>
```

查询区域中是否包含某接口

```
$ firewall-cmd [--zone=<zone>] --query-interface=<interface>
```

列举区域中启用的服务

```
$ firewall-cmd [ --zone=<zone> ] --list-services
```

启用应急模式阻断所有网络连接，以防出现紧急状况

```
1  $ firewall-cmd --panic-on
```

禁用应急模式

```
1  $ firewall-cmd --panic-off
```

在0.3.0之前的FirewallD版本中, panic选项是 `--enable-panic` 与 `--disable-panic` 。

查询应急模式

```
1  $ firewall-cmd --query-panic
```

处理运行时区域

运行时模式下对区域进行的修改不是永久有效的。重新加载或者重启后修改将失效。

管理端口

列出dmz级别的被允许的进入端口

```
1  $ firewall-cmd --zone=dmz --list-ports
```

允许tcp端口8080至dmz级别

```
1  $ firewall-cmd --zone=dmz --add-port=8080/tcp
```

管理服务

启用区域中的一种服务

```
$ firewall-cmd [--zone=<zone>] --add-service=<service> [--timeout=<seconds>]
```

此举启用区域中的一种服务。如果未指定区域，将使用默认区域。如果设定了超时时间，服务将只启用特定秒数。如果服务已经活跃，将不会有任何警告信息。

添加smtp服务至work zone

```
$ firewall-cmd --zone=work --add-service=smtp
```

使区域中的ipp-client服务生效60秒

```
$ firewall-cmd --zone=home --add-service=ipp-client --timeout=60
```

启用默认区域中的http服务

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

禁用区域中的某种服务

```
$ firewall-cmd [--zone=<zone>] --remove-service=<service>
```

此举禁用区域中的某种服务。如果未指定区域，将使用默认区域。

移除work zone中的smtp服务

```
$ firewall-cmd --zone=work --remove-service=smtp
```

禁止home区域中的http服务

```
$ firewall-cmd --zone=home --remove-service=http
```

区域种的服务将被禁用。如果服务没有启用，将不会有任何警告信息。

查询区域中是否启用了特定服务

```
1  $ firewall-cmd [--zone=<zone>] --query-service=<service>
```

如果服务启用，将返回1,否则返回0。没有输出信息。

启用区域端口和协议组合

```
1  $ firewall-cmd [--zone=<zone>] --add-port=<port>[-<port>]/<protocol> [--timeout=<seconds
```

此举将启用端口和协议的组合。端口可以是一个单独的端口 <port> 或者是一个端口范围 <port>-<port> 。协议可以是tcp或udp。

禁用端口和协议组合

```
1  $ firewall-cmd [--zone=<zone>] --remove-port=<port>[-<port>]/<protocol>
```

查询区域中是否启用了端口和协议组合

```
1  $ firewall-cmd [--zone=<zone>] --query-port=<port>[-<port>]/<protocol>
```

如果启用，此命令将有返回值。没有输出信息。

启用区域中的IP伪装功能

```
1  $ firewall-cmd [--zone=<zone>] --add-masquerade
```

此举启用区域的伪装功能。私有网络的地址将被隐藏并映射到一个公有IP。这是地址转换的一种形式，常用于路由。由于内核的限制，伪装功能仅可用于IPv4。

禁用区域中的IP伪装

```
1   $ firewall-cmd [--zone=<zone>] --remove-masquerade
```

查询区域的伪装状态

```
1   $ firewall-cmd [--zone=<zone>] --query-masquerade
```

如果启用，此命令将有返回值。没有输出信息。

启用区域的ICMP阻塞功能

```
1   $ firewall-cmd [--zone=<zone>] --add-icmp-block=<icmptype>
```

此举将启用选中的Internet控制报文协议(ICMP)报文进行阻塞。 ICMP 报文可以是请求信息或者创建的应答报文，以及错误应答。

禁止区域的ICMP阻塞功能

```
1   $ firewall-cmd [--zone=<zone>] --remove-icmp-block=<icmptype>
```

查询区域的ICMP阻塞功能

```
1   $ firewall-cmd [--zone=<zone>] --query-icmp-block=<icmptype>
```

如果启用，此命令将有返回值。没有输出信息。

阻塞区域的响应应答报文

```
1   $ firewall-cmd --zone=public --add-icmp-block=echo-reply
```

在区域中启用端口转发或映射

```
1  $ firewall-cmd [--zone=<zone>] --add-forward-port=port=<port>[-<port>]:proto=<protocol>
```

端口可以映射到另一台主机的同一端口，也可以是同一主机或另一主机的不同端口。端口号可以是一个单独的端口 <port> 或者是端口范围 <port>-<port> 。协议可以为tcp或udp 。目标端口可以是端口号 <port> 或者是端口范围 <port>-<port> 。目标地址可以是IPv4地址。受内核限制，端口转发功能仅可用于IPv4。

禁止区域的端口转发或者端口映射

```
1  $ firewall-cmd [--zone=<zone>] --remove-forward-port=port=<port>[-<port>]:proto=<protoco
```

查询区域的端口转发或者端口映射

```
1  $ firewall-cmd [--zone=<zone>] --query-forward-port=port=<port>[-<port>]:proto=<protocol
```

如果启用，此命令将有返回值。没有输出信息。

端口转发实例

要打开端口转发，则需要先

```
1  $ firewall-cmd --zone=external --add-masquerade
```

然后转发tcp 22端口至3753

```
1  $ firewall-cmd --zone=external --add-forward-port=port=22:proto=tcp:toport=3753
```

转发22端口数据至另一个ip的相同端口上

```
1  $ firewall-cmd --zone=external --add-forward-port=port=22:proto=tcp:toaddr=192.168.1.100
```

转发22端口数据至另一ip的2055端口上

```
1  $ firewall-cmd --zone=external --add-forward-port=port=22:proto=tcp:toport=2055:toaddr=1
```

将区域home的ssh转发到127.0.0.2

```
1  $ firewall-cmd --zone=home --add-forward-port=port=22:proto=tcp:toaddr=127.0.0.2
```

处理永久区域

永久选项不直接影响运行时的状态。这些选项仅在重载或者重启服务时可用。为了使用运行时和永久设置，需要分别设置两者。 选项 `--permanent` 需要是永久设置的第一个参数。

获取永久选项所支持的服务

```
1  $ firewall-cmd --permanent --get-services
```

获取永久选项所支持的ICMP类型列表

```
1  $ firewall-cmd --permanent --get-icmptypes
```

获取支持的永久区域

```
1  $ firewall-cmd --permanent --get-zones
```

启用区域中的服务

```
1  $ firewall-cmd --permanent [--zone=<zone>] --add-service=<service>
```

此举将永久启用区域中的服务。如果未指定区域，将使用默认区域。

例: 永久启用home区域中的ipp-client服务

```
1   $ firewall-cmd --permanent --zone=home --add-service=ipp-client
```

禁用区域中的一种服务

```
1   $ firewall-cmd --permanent [--zone=<zone>] --remove-service=<service>
```

查询区域中的服务是否启用

```
1   $ firewall-cmd --permanent [--zone=<zone>] --query-service=<service>
```

如果服务启用，此命令将有返回值。此命令没有输出信息。

永久启用区域中的一个端口-协议组合

```
1   $ firewall-cmd --permanent [--zone=<zone>] --add-port=<port>[-<port>]/<protocol>
```

例如

允许某范围的udp端口至public级别，并永久生效

```
1   $ firewall-cmd --zone=public --add-port=5060-5059/udp --permanent
```

永久启用home区域中的https(tcp 443)端口

```
1   $ firewall-cmd --permanent --zone=home --add-port=443/tcp
```

永久禁用区域中的一个端口-协议组合

```
1  $ firewall-cmd --permanent [--zone=<zone>] --remove-port=<port>[-<port>]/<protocol>
```

查询区域中的端口-协议组合是否永久启用

```
1  $ firewall-cmd --permanent [--zone=<zone>] --query-port=<port>[-<port>]/<protocol>
```

如果服务启用，此命令将有返回值。此命令没有输出信息。

永久启用区域中的伪装

```
1  $ firewall-cmd --permanent [--zone=<zone>] --add-masquerade
```

此举启用区域的伪装功能。私有网络的地址将被隐藏并映射到一个公有IP。这是地址转换的一种形式，常用于路由。由于内核的限制，伪装功能仅可用于IPv4。

永久禁用区域中的伪装

```
1  $ firewall-cmd --permanent [--zone=<zone>] --remove-masquerade
```

查询区域中的伪装的永久状态

```
1  $ firewall-cmd --permanent [--zone=<zone>] --query-masquerade
```

如果服务启用，此命令将有返回值。此命令没有输出信息。

永久启用区域中的ICMP阻塞

```
1  $ firewall-cmd --permanent [--zone=<zone>] --add-icmp-block=<icmptype>
```

此举将启用选中的 Internet 控制报文协议(ICMP)报文进行阻塞。 ICMP 报文可以是请求信息或者创建的应答报文或错误应答报文。

永久禁用区域中的ICMP阻塞

```
$ firewall-cmd --permanent [--zone=<zone>] --remove-icmp-block=<icmptype>
```

例: 阻塞公共区域中的响应应答报文

```
$ firewall-cmd --permanent --zone=public --add-icmp-block=echo-reply
```

查询区域中的ICMP永久状态

```
$ firewall-cmd --permanent [--zone=<zone>] --query-icmp-block=<icmptype>
```

如果服务启用，此命令将有返回值。此命令没有输出信息。

在区域中永久启用端口转发或映射

```
$ firewall-cmd --permanent [--zone=<zone>] --add-forward-port=port=<port>[-<port>]:proto
```

端口可以映射到另一台主机的同一端口，也可以是同一主机或另一主机的不同端口。端口号可以是一个单独的端口 <port> 或者是端口范围 <port>-<port> 。协议可以为tcp或udp 。目标端口可以是端口号 <port> 或者是端口范围 <port>-<port> 。目标地址可以是IPv4 地址。受内核限制，端口转发功能仅可用于IPv4。

例: 将home区域的ssh服务转发到127.0.0.2

```
$ firewall-cmd --permanent --zone=home --add-forward-port=port=22:proto=tcp:toaddr=127.0
```

永久禁止区域的端口转发或者端口映射

```
$ firewall-cmd --permanent [--zone=<zone>] --remove-forward-port=port=<port>[-<port>]:pr
```

查询区域的端口转发或者端口映射状态

```
$ firewall-cmd --permanent [--zone=<zone>] --query-forward-port=port=<port>[-<port>]:pro
```

如果服务启用，此命令将有返回值。此命令没有输出信息。

IP封禁

```
$ firewall-cmd --permanent --add-rich-rule="rule family='ipv4' source address='222.222.2
```

直接选项

直接选项主要用于使服务和应用程序能够增加规则。 规则不会被保存，在重新加载或者重启之后必须再次提交。传递的参数 `<args>` 与iptables, ip6tables以及ebtables一致。

选项 `--direct` 需要是直接选项的第一个参数。

将命令传递给防火墙。参数 `<args>` 可以是iptables, ip6tables以及ebtables命令行参数。

```
$ firewall-cmd --direct --passthrough { ipv4 | ipv6 | eb } <args>
```

为表 `<table>` 增加一个新链 `<chain>`

```
$ firewall-cmd --direct --add-chain { ipv4 | ipv6 | eb } <table> <chain>
```

从表 `<table>` 中删除链 `<chain>`

```
$ firewall-cmd --direct --remove-chain { ipv4 | ipv6 | eb } <table> <chain>
```

查询 <chain> 链是否存在与表 <table> . 如果是，返回0,否则返回1.

```
$ firewall-cmd --direct --query-chain { ipv4 | ipv6 | eb } <table> <chain>
```

如果启用，此命令将有返回值。此命令没有输出信息。

获取用空格分隔的表 <table> 中链的列表。

```
$ firewall-cmd --direct --get-chains { ipv4 | ipv6 | eb } <table>
```

为表 <table> 增加一条参数为 <args> 的链 <chain> ，优先级设定为 <priority> 。

```
$ firewall-cmd --direct --add-rule { ipv4 | ipv6 | eb } <table> <chain> <priority> <args
```

从表 <table> 中删除带参数 <args> 的链 <chain> 。

```
$ firewall-cmd --direct --remove-rule { ipv4 | ipv6 | eb } <table> <chain> <args>
```

查询带参数 <args> 的链 <chain> 是否存在表 <table> 中. 如果是返回0,否则返回1.

```
$ firewall-cmd --direct --query-rule { ipv4 | ipv6 | eb } <table> <chain> <args>
```

如果启用，此命令将有返回值。此命令没有输出信息。

获取表 <table> 中所有增加到链 <chain> 的规则，并用换行分隔。

```
$ firewall-cmd --direct --get-rules { ipv4 | ipv6 | eb } <table> <chain>
```

使用iptables静态防火墙

对于用惯了iptables的用户，FirewallD的使用起来需要熟悉一段时间。如果你想使用熟悉的iptables和ip6tables静态防火墙规则，方法如下

安装iptables-services

```
$ yum install iptables-services
```

禁用firewalld

```
$ systemctl mask firewalld.service
$ systemctl stop firewalld.service
```

启用iptables和ip6tables

```
$ systemctl enable iptables.service
$ systemctl enable ip6tables.service
```

静态防火墙规则配置文件是 `/etc/sysconfig/iptables` 以及 `/etc/sysconfig/ip6tables` 。

注：iptables与iptables-services软件包不提供与服务配套使用的防火墙规则. 这些服务是用来保障兼容性以及供想使用自己防火墙规则的人使用的. 你可以安装并使用 `system-config-firewall` 来创建服务所需要的规则. 为了能使用 `system-config-firewall` , 你必须停止firewalld。

参考文档

http://www.google.com
http://havee.me/linux/2015-01/using-firewalls-on-centos-7.html
https://fedoraproject.org/wiki/FirewallD/zh-cn

更多精彩内容，请关注微信公众号Hi-Linux，第一时间推送给您!

#Linux #FirewallD

❮ Linux命令行下抓取HTTP流量的工具--httpry

Systemd入门教程 ❯