1. firewalld 란

- Linux 커널 2.2 까지는 ipchains 이라는 패킷 필터링/방화벽 프레임워크가 구현되어 있었고 2.4부터는 더 유연하고 다양한 기능을 가진 netfilter 로 프레임워크가 교체 되었습니다.
- iptables 은 netfilter 프레임워크의 최상단에 위치하는 사용자 레벨의 프로그램으로 시스템 관리자는 iptables 명령어로 리눅스 서버로 들어오고 나가는 패킷을 필터링하거나 포트 포워딩을 설정할 수 있으며 방화벽으로도 사용할 수 있습니다.
- iptables 는 숙련된 관리자가 아니면 사용이 어려운 단점이 있었는데 이런 문제를 해결하고자 RHEL/CentOS 7 부터는 방화벽을 firewalld 라는 데몬으로 교체하였고 이에 따라 사용자 레벨의 프로그램은 iptables 명령어 대신 명령행에서는 firewall-cmd , GUI 환경에서는 firewall-config를 사용하게 되었습니다.

영구적 규칙과 정책 재구동

- 기본적으로 firewall-cmd 로 방화벽 정책을 변경했을 경우 현재 구동되고 있는 firewalld에 즉시 적용되지만 정책은 지속성이 없이 임시로 적용되며 정책을 재구동하는 명령어인 `firewall-cmd –reload`를 실행하거나 시스템을 재부팅하면 예전 정책으로 다시 초기화 되며 이로 인해 서비스의 장애가 발생할 수 있습니다.
- 이 때문에 방화벽 정책을 영구적으로 유지하기 위해서는 `–permanent`옵션을 추가해서 실행하면 되지만 이는 즉시 적용되지 않고 `firewall-cmd –reload` 명령어로 방화벽 정책을 재구동하거나 재부팅을 하기 전에는 변경한 방화벽 설정이 적용되지 않습니다.
- firewalld를 처음에 사용할 때 이 때문에 혼란을 겪고 정책 설정을 잘못한 걸로 오해하는 사용자가 많으므로 아래 표를 보고 방화벽 정책의 즉시 적용 여부와 지속성 여부를 꼭 익혀 두어야 합니다.

	firewall-cmd	**firewall-cmd -permanent**
즉시 적용(firewall-cmd –reload 불필요)	예	아니요
재부팅시 정책의 지속 여부	아니오	예

- 예로 `fireall-cmd`로 정책을 추가했을 경우 즉시 반영되지만 만약 잘못된 설정이었을 경우에 `firewall-cmd –reload` 명령어를 실행하면 예전 정책으로 복구됩니다.
- 하지만 `firewall-cmd –permanent`로 정책을 추가했을 경우 `firewall-cmd –reload` 명령어를 실행해야 변경한 정책이 적용되며 예전 정책으로 복구하려면 새로 추가한 정책을 삭제하고 다시 방화벽을 구동해야 합니다.

2. 구동상태 확인

방화벽 실행 중인지 확인

```
#방화벽 실행 확인(실행중이면 running, 실행중이 아니면 not running으로 표시)
[root@dns ~] firewall-cmd --state # 현재상태 확인

#not running일 경우 방화벽 실행
[root@dns ~] systemctl enable firewalld # 부팅 후에도 서비스가 자동으로 시작되게 설정한다.
[root@dns ~] systemctl start firewalld # firewalld를 시작한다.
```

3. 설치

```
[root@dns ~] yum install firewalld # 설치
[root@dns ~] systemctl start firewalld # 시작
[root@dns ~] systemctl enable firewalld # 부팅 후에도 서비스가 자동으로 시작되게 설정한다.
```

4. 특정 IP 및 포트만 허용하는 방법

IP 허용

```
#IP 허용
[root@dns ~] firewall-cmd --permanent --add-source=10.10.10.10

#IP 허용 제거
[root@dns ~] firewall-cmd --permanent --remove-source=10.10.10.10
```

IP대역 허용

```
#IP 대역 허용
[root@dns ~] firewall-cmd --permanent --add-source=10.10.10.0/24

#IP 대역 허용 제거
[root@dns ~] firewall-cmd --permanent --remove-source=10.10.10.0/24
```

- 여기서 /24는 해당 IP의 마지막까지 대역을 설정한다는 의미로, 위 코드를 예를 들면 10.10.10.0에서 10.10.10.255까지를 의미한다.

서비스 포트 허용

```
#서비스 포트 허용
[root@dns ~] firewall-cmd --permanent --add-port=80/tcp

#서비스 포트 허용 제거
[root@dns ~] firewall-cmd --permanent --remove-port=80/tcp
```

서비스 포트 범위 허용

```
#서비스 포트 범위 허용
[root@dns ~] firewall-cmd --permanent --add-port=1000-2000/tcp

#서비스 포트 범위 허용 제거
[root@dns ~] firewall-cmd --permanent --remove-port=1000-2000/tcp
```

IP에 대해 서비스 포트 허용

```
#IP에 대해 서비스 포트 허용
[root@dns ~] firewall-cmd --permanent --add-rich-rule='rule family="ipv4"
source address=10.10.10.10 port port="80" protocol="tcp" accept'

#IP에 대한 서비스 포트 허용 제거
[root@dns ~] firewall-cmd --permanent --remove-rich-rule='rule family="ipv4"
source address=10.10.10.10 port port="80" protocol="tcp" accept'
```

IP 차단

```
#IP 차단
[root@dns ~] firewall-cmd --permanent --add-rich-rule='rule family="ipv4"
source address=10.10.10.10 drop'

#IP 차단 제거
[root@dns ~] firewall-cmd --permanent --remove-rich-rule='rule family="ipv4"
source address=10.10.10.10 drop'
```

방화벽 Rule 적용

```
#방화벽 Rule 변경 내역 적용
[root@dns ~] firewall-cmd --reload

#방화벽 Rule 리스트 확인
[root@dns ~] firewall-cmd --list-all
```

5. Network Zone

- firewalld의 특징 중 하나는 이 장의 서두에서 설명한 네트워크 구성과 영역을 기본 설정으로 포함시켰다는 것입니다. 이에 따라 서버가 어떤 zone에 포함될 지만 정의하면 상대적으로 설정해야 하는 부분이 대폭 간소화 되었습니다.
- 먼저 기본 설정된 zone 의 목록을 확인하려면 firewall-cmd 명령에 –get-zones 옵션을 주고 실행하면 됩니다.

```
[root@dns ~] sudo firewall-cmd --get-zones

work drop internal external trusted home dmz public block
```

- 표시된 목록은 사전 정의된 zone으로 이제 서버의 용도에 맞게 zone을 할당하면 되며, 해당 zone이 어떤 정책을 갖고 있는지 확인하려면 –list-all-zones 옵션으로 사전 정의된 zone에 대한 자세한 정보를 볼 수 있습니다.

```
[root@dns ~] sudo firewall-cmd --list-all-zones

dmz
target: default
icmp-block-inversion: no
interfaces:
sources:
services: ssh
ports:
protocols:
masquerade: no
forward-ports:
sourceports:
icmp-blocks:
rich rules:


internal
target: default
icmp-block-inversion: no
interfaces:
sources:
services: dhcpv6-client mdns samba-client ssh
ports:
protocols:
masquerade: no
forward-ports:
sourceports:
icmp-blocks:
rich rules:
```

- 각 zone 별 services항목을 보면 해당 zone에서 기본적으로 허용하는 서비스 포트를 확인할 수 있으며, dmz는 ssh를, internal은 dhcpv6-client, mdns 등을 허용하는 것을 알 수 있습니다.
- 현재 활성화된 zone을 확인하려면 –get-active-zone 옵션을 실행하면 되며 현재 활성화된 zone은 public인 것을 알 수 있습니다.(default)

```
[root@dns ~] firewall-cmd --get-active-zone

public
interfaces: eno0
```

존 설정 정보 보기(--list-all)

- 현재 활성화된 존 정보를 확인하려면 –list-all 옵션을 사용하면 됩니다. 아래는 현재 활성화된

존이 dmz 이고 10022 등의 port가 열려 있음을 표시합니다.

```
[root@dns ~] sudo firewall-cmd --list-all

dmz (active)
target: default
icmp-block-inversion: no
interfaces: enp5s0f0 enp5s0f1
sources:
services: ssh http https
ports: 10022/tcp 2120-2121/tcp 20/tcp 2120-2142/tcp 10000/tcp
protocols:
masquerade: no
forward-ports:
source-ports:
icmp-blocks:
rich rules:
```

기본 존 바꾸기

- 기본 존을 변경하려면 –set-default-zone={ZONE_NAME}옵션을 사용하여 변경해주면 되며 다음은 dmz로 변경하는 예제입니다.

```
[root@dns ~] firewall-cmd --set-default-zone=dmz
```

존 만들기

- 만약 firewalld 에 내장된 zone설정이 현재 서비스에 딱 들어맞지 않는다면 –new-zone 옵션을 사용하여 새로 zone을 만들면 됩니다. 아래는 webserver라는 이름의 새로운 존을 만드는 예제입니다.

```
[root@dns ~] firewall-cmd --permanent --new-zone=webserver
```

- 생성한 존은 바로 반영되지 않으므로 방화벽 정책을 다시 읽어 들여야 합니다. –reload을 주어서 변경된 내역을 firewalld에 반영할 수 있습니다.

6. 사전 정의 서비스

사전 정의된 서비스 확인

- firewalld 에는 방화벽 정책 변경시 용이하도록 많이 사용하는 서비스를 사전에 정의해 놓고 있으며 –get-services 옵션으로 전체 목록을 확인할 수 있습니다.

```
[root@dns ~] firewall-cmd --get-services
RH-Satellite-6 RH-Satellite-6-capsule amanda-client amanda-k5-client amqp
```

```
amqps apcupsd audit bacula bacula-client bgp bitcoin bitcoin-rpc bitcoin-
testnet bitcoin-testnet-rpc ceph ceph-mon cfengine condor-collector ctdb
dhcp dhcpv6 dhcpv6-client distcc dns docker-registry docker-swarm dropbox-
lansync elasticsearch etcd-client etcd-server finger freeipa-ldap freeipa-
ldaps freeipa-replication freeipa-trust ftp ganglia-client ganglia-master
git gre high-availability http https imap imaps ipp ipp-client ipsec irc
ircs iscsi-target isns jenkins kadmin kerberos kibana klogin kpasswd kprop
kshell ldap ldaps libvirt libvirt-tls lightning-network llmnr managesieve
matrix mdns minidlna mongodb mosh mountd mqtt mqtt-tls ms-wbt mssql murmur
mysql nfs nfs3 nmea-0183 nrpe ntp nut openvpn ovirt-imageio ovirt-
storageconsole ovirt-vmconsole plex pmcd pmproxy pmwebapi pmwebapis pop3
pop3s postgresql privoxy proxy-dhcp ptp pulseaudio puppetmaster quassel
radius redis rpc-bind rsh rsyncd rtsp salt-master samba samba-client samba-
dc sane sip sips slp smtp smtp-submission smtps snmp snmptrap spideroak-
lansync squid ssh steam-streaming svdrp svn syncthing syncthing-gui synergy
syslog syslog-tls telnet tftp tftp-client tinc tor-socks transmission-client
upnp-client vdsm vnc-server wbem-http wbem-https wsman wsmans xdmcp xmpp-
bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-server
```

서비스 추가

- 웹 서버는 http(80), https(443) 으로 접근을 허용해야 하므로 –add-service=SERVICENAME 구문으로 서비스를 추가해 줍니다. –add-service 는 여러개의 서비스를 동시에 기술할 수 없으므로 http와 https를 따로 기술해 줍니다.

```
[root@dns ~] firewall-cmd --permanent --zone=webserver --add-service=http
[root@dns ~] firewall-cmd --permanent --zone=webserver --add-service=http
```

서비스 삭제

- 서비스 삭제는 –remove-service=SERVICENAME 구문을 사용합니다. 아래는 http/https 서비스를 삭제합니다.

```
[root@dns ~] firewall-cmd --permanent --zone=webserver --remove-service=http
[root@dns ~] firewall-cmd --permanent --zone=webserver --remove-
service=https
```

Tip

유용한 명령어

```
firewall-cmd --state # 방화벽 상태 확인

firewall-cmd --get-zones # 모든 Zone 출력

firewall-cmd --get-default-zone # 현재 적용된 Zone 출력
```

```
firewall-cmd --list-all # 사용 가능한 서비스/포트 출력

firewall-cmd --reload # 방화벽 리로드
```

Troubleshooting

Failed to restart firewalld.service: Unit is masked. 에러

- mask 뜻 자체가 우리가 쓰는 마스크, 즉, 가린다는 뜻으로 firewalld가 가려져서 restart할 수 없다는 의미이다. 따라서, 이를 unmask해주면 해결된다.

```
[root@dns ~] systemctl unmask firewalld
```

- firewall-cmd error on polkit

Ref

- firewalld 방화벽 사용
- [Linux] firewalld 방화벽 설정하기
- [CentOS 7] Firewalld 방화벽
- CentOS 7 방화벽(Firewall)에서 특정 IP 및 포트만 허용
- [CentOS 7] 방화벽 해제하기 - firewall-cmd

와프, linux, centos, firewall, firewalld, 방화벽