Linux Firewall
Concepts And rules

Firewall လို့ပြောလိုက်တာနဲ့ Network ပေါ်ကလာတဲ့ Traffic တွေ income, outcome connection တွေကို ကိုယ်လိုချင်တဲ့ connection မျိုးကို access ပေးဝင်ခြင်း၊ outbound connection သတ်မှတ်ပေးခြင်း၊ မလိုခြင်တဲ့ network တွေကိုဝိတ်ပြီး system ရဲ့လုံခြုံမှုကို စွမ်းဆောင်ပေးတယ်ဆိုတာလူတိုင်းသိပါတယ်။ firewall တွေဟာ Hardware Device အနေနဲ့ လည်းရှိသလို၊ software အနေနဲ့လည်းရှိကြပါတယ်။ ဒီခါမှာတော့ Linux ရဲ့ kernel အစိတ်အပိုင်းမှာ ပါဝင်တဲ့ netfilter modules ကိုခိုင်းစေတဲ့ firewall အမျိုးအစားကို လေ့လာရအောင်။

Firewalld

Firewalld ဆိုတာ Linux System ပေါ်မှာရှိတဲ့ Netfilter Modules နဲ့အလုပ်လုပ်နဲ့တွဲပြီးအလုပ်လုပ်တဲ့ daemon အမျိုးအစား firewall ဖြစ်ပါတယ်။ သူဟာ dynamic ဖြစ်ပါတယ်။ dynamic ဆိုတဲ့အတိုင်း ပြောင်းလွယ်ပြင်လွယ်ရှိသလို network ပေါ်ကလာတဲ့ incoming traffic, outgoing traffic connection တွေကို control လုပ်လို့ရပါတယ်။

Red Hat Linux Version 7 ကစပြီး release လုပ်လိုက်တာဖြစ်ပြီးတော့၊ အရင် version 6 ကမှာတုန်းကတော့ firewalld အစား iptables ကိုအသုံးပြုကြပါတယ်။ firewalld ဟာ iptables ထက်ပိုပြီးလွယ်ကူစွာ configure ချနိုင်ပြီး firewall rules တွေကို စုစည်းပြီး အလုပ်လုပ်ပေးနိုင်တဲ့ rich rules function တွေလည်းပါဝင်ပါတယ်။

ကျွန်တော်တို့ လိုချင်တဲ့ connection တွေကိုပဲ ဖွင့်ပေးမယ်။ မလိုအပ်တဲ့ connection တွေကို ဝိတ်ခြင်းအစရှိသဖြင့် system ရဲ့လိုအပ်ချက်ပေါ်မူတည်ပြီး သူ့ကို control လုပ်လို့ရပါတယ်။ ဒီလို control လုပ်တဲ့အခါ firewall ကို rules တွေသတ်မှတ်ပေးရပါတယ်။ firewalld ဟာ Dynamic ဖြစ်တယ်လို့ပြောခဲ့တယ်၊ ဒါကြောင့် ဒီ rule တွေကို creating, changing, and deleting စတဲ့ ကျွန်တော်တို့ customize လုပ်လိုက်တိုင်း၊ rule တွေချလိုက်တိုင်း firewalld service ကို restart ချစရာမလိုပဲ သူအလိုလို auto detect အလုပ်လုပ်ပေးပါတယ်။

Firewalld ကိုအသုံးပြုတဲ့အခါ zone နဲ့ service တွေသိထားဖို့လိုအပ်ပါတယ်။ zone ၉ ခု ရှိပါတယ်။ ဒီ zone 9 ခုတွေမှာကြိုတင်သတ်မှတ်ထားတဲ့ zone တစ်ခုခြင်းစီအတွက် rules

ကိုယ်စီရှိကြပါတယ်။ နောက်တစ်ခုက service တွေပေါ့။ Linux system တွေမှာ network ကိုအသုံးပြုပြီးအလုပ်လုပ်ကြတဲ့ service တွေမှာဆိုရင် Ip address တွေ ports တွေနဲ့အလုပ်လုပ်ကြရပါတယ်။ service တစ်ခုကို သုံးမယ်ဆိုရင် ကျွန်တော်တို့ ဒီ service အသုံးပြုတဲ့ ports တွေကို သိထားရပါတယ်။ ဒါမှ firewall မှာ သွားရောက်ပြီး service ရဲ့ port တွေကို သွားထည့်ပေးရပါတယ်။ သွားထည့်ပေးတဲ့အချိန်မှာ zone တွေကို ရွေးချယ်ရပါတယ်။

`Block`

ဘယ်လို network connections မျိုးပဲလာလာ ဒီ zone ထဲမှာ ထည့်ထားတဲ့ service တွေ, ip တွေ, ports တွေကတော့ incoming network connections တွေကို rejects လုပ်ပါမယ်၊ icmp-packets တွေဆိုရင်လည်း auto ဝိတ်ပေးပါတယ်။ block လုပ်လိုက်တဲ့အကြောင်း notify လည်းပြပါတယ်။

`dmz`

Dmz ကတော့ demilitarized zone လို့ခေါ်တယ်။ ဒီထဲမှာ သတ်မှတ်ထားတဲ့ incoming connections တွေကို accepted လုပ်ပေးမှာပါ။ Limited Access နဲ့ ကန့်သတ်ထားတဲ့ rule တွေနဲ့ပဲ အလုပ်လုပ်နိုင်မှာဖြစ်ပါတယ်။ အားလုံးကိုအလုပ်ပေးမလုပ်ပါဘူး။

`Drop`

Incoming network packets တွေကို ဝိတ်ပေးတာပါ။ ဒါပေမယ့် သူက block zone လိုမျိုး notify မပြပါဘူး။ ဒါပေမယ့် outgoing network တွေတော့အလုပ်လုပ်နိုင်ပါတယ်။

`External`

ဥပမာ - ကျွန်တော်တို့ Laptop က တခြား outside တစ်ခုခုက wifi နဲ့သုံးနေတယ်ဆိုပါစို့ ဒီအခါစိတ်မချရတဲ့ device တွေကနေ unauthorized network traffic တွေမဝို့နိုင်အောင် ဒီ zone လေးက ကာကွယ်ပေးနိုင်ပါတယ်။ ဒီထဲမှာတော့ incoming connections တွေကိုပဲ accepts လုပ်နိုင်ပါတယ်။

`home`

ဒီ zone ကတော့ home ဆိုတဲ့အတိုင်း ကိုယ်စိတ်ချယုံကြည်လို့ရတဲ့ networked computers တွေလာမယ့် incoming connections တွေကို accepts လုပ်ပေးမယ့် zone ပဲဖြစ်ပါတယ်။

`internal`

Home zone နဲ့ သိပ်မကွာပါဘူး။ အတူတူပါပဲ၊ ဒီကောင်ကတော့ home zone ထက်ကို ပိုပြီး ယုံကြည်ရတဲ့ networked computers တွေ ကိုယ့်ရဲ့ network ပေါ်မှာလာရောက်ချိတ်ဆက်တဲ့အခါ အသုံးပြုနိုင်ပါတယ်။
သူလည်း ကျွန်တော်တို့ သတ်မှတ်ထားတဲ့ incoming connections တွေကို accepts လုပ်ပေးနိုင်ပါတယ်။

`public`

Public areas တွေမှာ သုံးဖို့အတွက်ဖြစ်ပါတယ်။ ဒီ zone သည် zone 9 ခု ထဲမှာ defaults zone ပဲဖြစ်ပါတယ်။ ဒီ zone ထဲမှာလည်း ကျွန်တော်တို့ သတ်မှတ်ထားတဲ့ incoming connections တွေကို accepts လုပ်ပေးမှာဖြစ်ပါတယ်။

`Trusted`

ဒီ zone ကတော့ network အားလုံး လက်ခံပေးပါတယ်။ firewalld ကြီးက trusted zone ပေါ်မှာ အလုပ်လုပ်နေမယ် ဆိုရင်တော့ connections အားလုံး လက်ခံပေးမှာဖြစ်ပါတယ်။

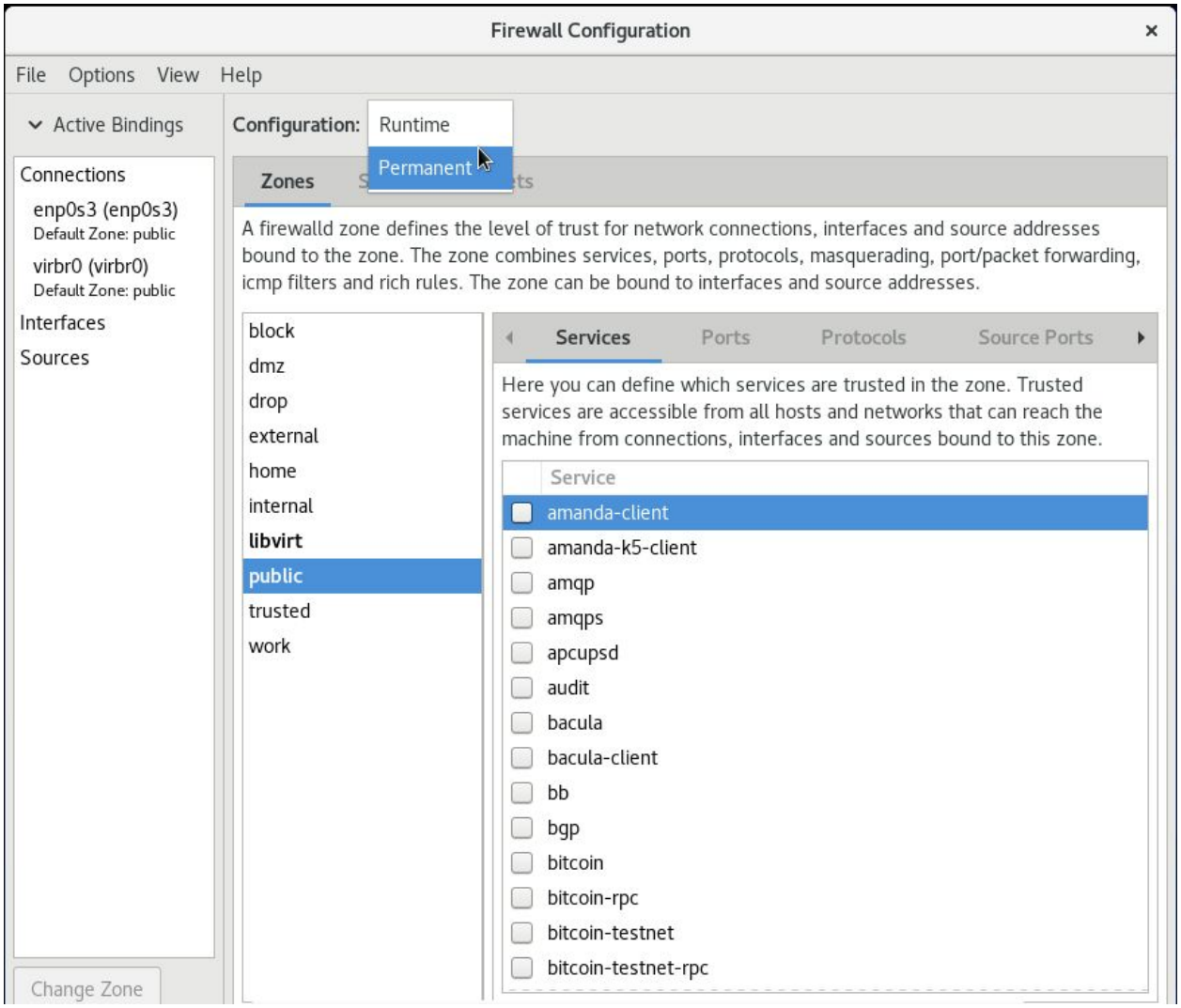
`work`

ဒါကတော့ working area မှာအသုံးပြုဖို့အတွက် သတ်မှတ်ပေးထားခြင်းဖြစ်ပါတယ်။
ဒီထဲမှာလည်း ကျွန်တော်တို့ ချမှတ်ထားတဲ့ firewall rules အတိုင်းအလုပ်လုပ်ပေးပါတယ်။
man firewalld လို့ကြည့်လိုက်ရင် rules တွေရဲ့ ပုံစံတွေကို အပြည့်အစုံ အသုံးပြုပုံတွေကိုလည်း သေချာမြင်ရမှာပါ။

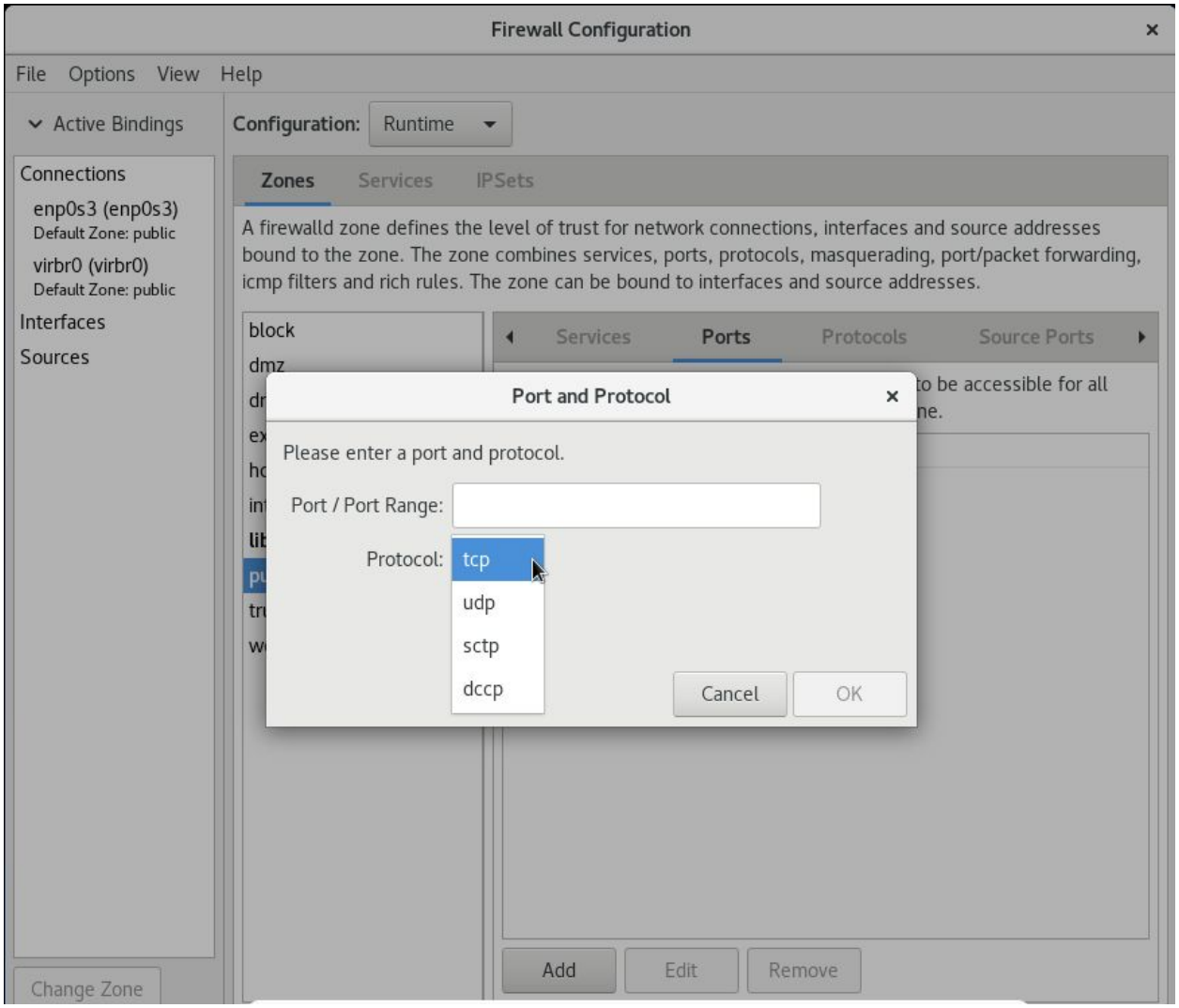
Firewall-config ဆိုတဲ့ firewalld ကို configure ချဖို့အတွက် GUI tools လေးလည်းရှိပါတယ်။ ဒီ ကောင်က Graphical အနေနဲ့အလုပ်လုပ်ဖို့ အတွက်ဖြစ်ပါတယ်။

```
# yum install firewall-config
```

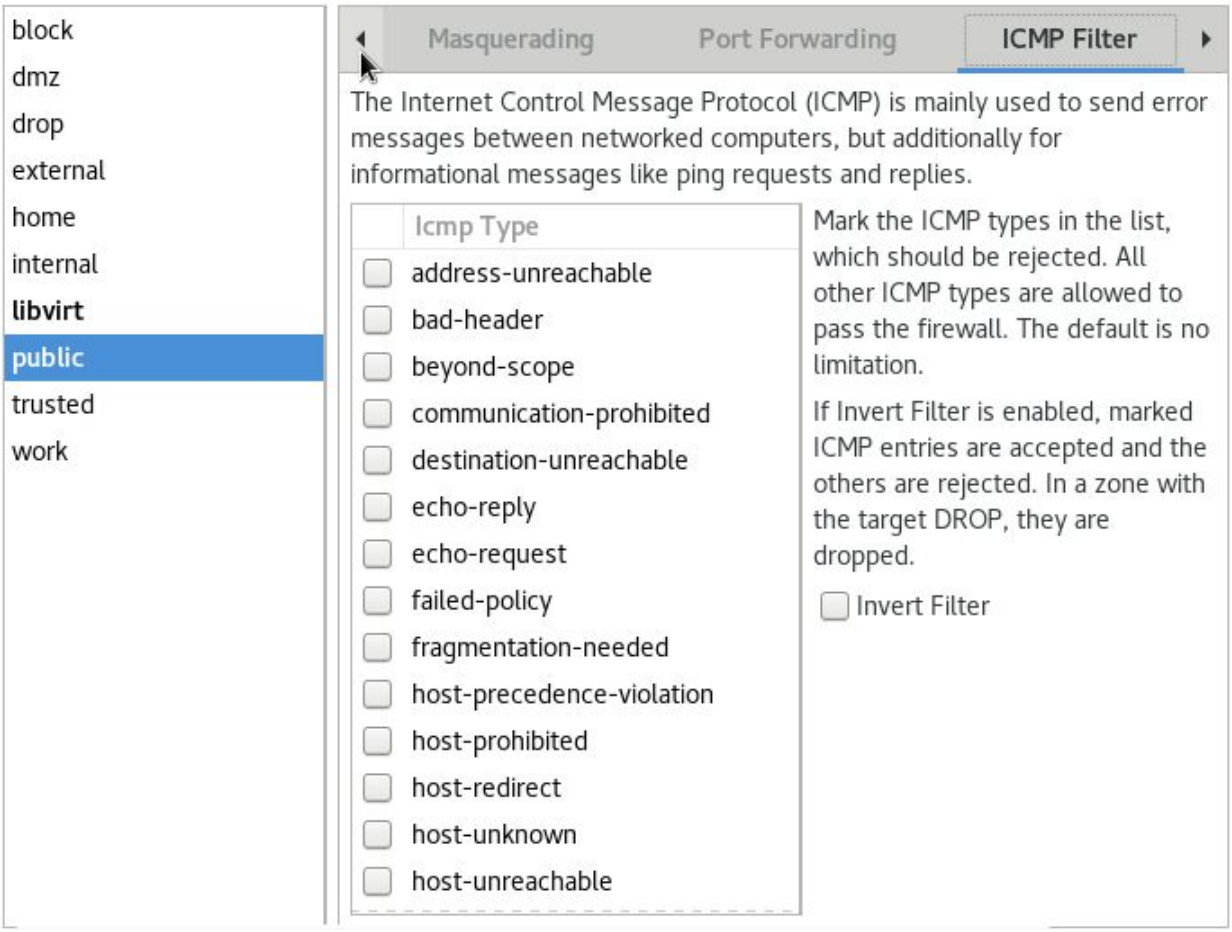
နဲ့ install လုပ်လိုက်ရင် GUI tools လေးကိုရရှိမှာဖြစ်ပါတယ်။



ဒီ GUI tools လေးထဲမှာလည်း ကျွန်တော်တို့ အသုံးပြုမယ့် network Card အမျိုးအစားကို ရွေးချယ်ပြီး rules တွေကို permanent, runtime, အစရှိသဖြင့်သတ်မှတ်နိုင်ပြီးပိုင်တယ်။



Service တွေ၊ port တွေ၊ protocols တွေ၊ အစရှိသဖြင့် ပုံပါအတိုင်းလည်း GUI နဲ့ထည့်သွင်း ပြီး rules တွေသတ်မှတ်ပေးနိုင်ပါတယ်။



GUI tools လေးဟာအတော့ကိုစုံလင်ပါတယ်။ port forwarding, IP masquerading နဲ့ ICMP packet filters အစရှိသည်တို့ကိုလည်းလုပ်ဆောင်နိုင်ပါတယ်။ အိုကေဒါဆိုရင် Firewall-cmd ဆိုတဲ့ command line အသုံးပြုပြီး rules ချမှတ်တာလေးတွေ လေ့လာကြည့်ရအောင်။

```
# systemctl start firewalld
# systemctl enable firewalld
# systemctl status firewalld
```

ဒီ command လေးနဲ့ ကြည့်မယ်ဆိုရင် firewalld service အလုပ်လုပ်နေသလားဆိုတာမြင် တွေ့နိုင်ပါတယ်။

```
# firewall-cmd --state
```

ဒါကတော့ firewalld ရဲ့ state ကိုကြည့်တာပါ။ firewall ရဲ့ option အတော်များများဟာ GNU options တွေနဲ့ အလုပ်လုပ်ကြတာများပါတယ်။

```
# firewall-cmd --list-all
```

```
[root@server ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ssh
  ports:
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

List-all ဆိုတဲ့ option လေးနဲ့ကြည့်မယ်ဆိုရင် firewall ရဲ့ defaults zone ဖြစ်တဲ့ public zone မှာ သတ်မှတ်ထားတဲ့ rules တွေကို မြင်တွေ့ရမှာပါ။

```
# firewall-cmd --list-all --zone=block
```

```
[root@server ~]# firewall-cmd --list-all --zone=block
block
  target: %%REJECT%%
  icmp-block-inversion: no
  interfaces:
  sources:
  services:
  ports:
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

Zone တစ်ခုအတွက်သီးသန့်ကြည့်ချင်တယ်ဆိုရင်တော့ --zone ဆိုတဲ့ option နဲ့ ကြည့်နိုင်ပါတယ်။command မှာပြထားတဲ့အတိုင်း block zone ကိုသီးသန့်ကြည့်ရှုတာဖြစ်ပါတယ်။

```
# firewall-cmd --help
```

ဒါကတော့ help file ကို ခေါ်ကြည့်တာပါ။

```
# firewall-cmd --list-services
```

Firewall အတွက် အသုံးပြုလို့ရတဲ့ service name တွေကို list-service နဲ့ list ထုတ်ပြီးကြည့်နိုင်ပါတယ်။ firewalld ဟာ ကိုယ်သုံးချင်တဲ့ service ကို ON ပေးလိုက်ရုံနဲ့ service ကအသုံးပြုတဲ့ network ports တွေဟာ auto ပွင့်ပေးပါတယ်။

Firewall Rules Examples

Firewall ကိုအသုံးပြုမယ်ဆိုရင် ကျွန်တော်တို့အသုံးပြုမယ့် service တွေ port တွေကို သတ်ဖွင့်ပေးခြင်း၊ deny လုပ်ခြင်း၊ drop လုပ်ပေးခြင်းအစရှိတဲ့ rules တွေကို သတ်မှတ်ပေးဖို့အတွက် rules သတ်မှတ်တဲ့ နည်းလမ်းများကိုသိဖို့လိုအပ်ပါတယ်။

Firewall rules တွေကို သတ်မှတ်တဲ့အခါ runtime, permanent, timeout အစရှိသဖြင့် သတ်မှတ်နိုင်ပါတယ်။ အသုံးပြုချင်တဲ့ service ကို permanent on ပေးချင်တယ်။ ကိုယ်အသုံးပြုချင်တဲ့ port ကို timeout လုပ်ပြီး အချိန်ကာလတစ်ခုအတွင်းမှာပဲ ON ပေးခြင်တယ်။ အစရှိသဖြင့် rules များကို သတ်မှတ်ရပါတယ်။

အိုကေ ကျွန်တော်တို့ ssh port 22 ဖြစ်ပြီး tcp (protocol) ကိုအသုံးပြုမယ်ဆိုပါစို့။

```
# firewall-cmd --add-port=22/tcp
# firewall-cmd --remove-port=22/tcp
```

< < remove လုပ်လိုက်တာဖြစ်ပါတယ်။

ဒါဆိုရင် port 22 ကို runtime မှာဖွင့်ပေးလိုက်တာဖြစ်ပါတယ်။ zone ကတော့ defaults အရ public zone မှာပေါ့။ တကယ်လို့ ကျွန်တော်တို့က public zone မဟုတ်ဘူး။ home ဆိုတဲ့ zone မှာဖွင့်ချင်တယ်ဆိုရင်တော့။

```
# firewall-cmd --add-port=22/tcp --zone=home
```

ဒါဆိုရင် home zone မှာ tcp protocol (port 22) ကို ဖွင့်ပေးလိုက်တာဖြစ်ပါတယ်။ ဒီလိုဖွင့်ပေးလိုက်တာသည် runtime ပဲဖြစ်တယ်။ boottime ဖွင့်ချင်ရင်။

```
# firewall-cmd --add-port=22/tcp --zone=home --permanent
```

Remove ပြန်လုပ်ချင်ရင်, remove option ကိုအသုံးပြုလိုက်လို့ရပါတယ်။

```
# firewall-cmd --remove-port=22/tcp --zone=home --permanent
```

Rules ထဲမှာ permanent option ထည့်သုံးလိုက်ရင်အဆင်ပြေပါပြီ။ permanent ဖွင့်ပြီးရင် အမြဲလုပ်ဆောင်ပေးရမှာက firewall-cmd ကို reload ပေးရပါတယ်။ အာ့သလိုပဲ permanent remove လုပ်လိုက်ရင်လည်း reload သို့မဟုတ် runtime-to-permanent option ကိုအသုံးပြုရပါမယ်။

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

ဆိုပြီးတော့ runtime-to-permanent ဆိုတဲ့ option ကိုထည့်ပေးလည်းရတယ်။ reload လုပ်ပေးလည်း ရပါတယ်။

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

ဒီလိုပုံစံနဲ့ udp protocols တွေကိုလည်း rules တွေသတ်မှတ်ပေးနိုင်ပါတယ်။

```
# firewall-cmd --add-port=53/udp --permanent
# firewall-cmd --reload
```

ဒီလို protocol ports တွေထည့်ပေးလိုက်ရင် သက်ဆိုင်ရာ zone ပေါ်မူတည်ပြီး အလုပ်လုပ်ပေး သွားမှာဖြစ်ပါတယ်။ service nameတွေကိုလည်း rules သတ်မှတ်ပေးလို့ရပါတယ်။ ဥပမာ- http နဲ့ ssh services နှစ်ခုကို permanent နဲ့ allow လုပ်မယ်ဆိုပါစို့။

```
# firewall-cmd --add-service=ssh --permanent
# firewall-cmd --add-service=http --permanent
# firewall-cmd --reload
```

Remove ပြန်လုပ်ချင်တယ်ဆိုရင်တော့။

```
# firewall-cmd --remove-service=ssh --permanent
# firewall-cmd --remove-service=http --permanent
# firewall-cmd --reload
```

အကယ်လို့များ ကျွန်တော်တို့က ssh service ကိုတော့ ဖွင့်ပေးချင်တယ်၊ ၁၅ မိနစ်ပဲဖွင့်ပေးချင်တယ်ဆိုရင်တော့

```
# firewall-cmd --add-service=ssh --timeout 15m
```

Defaults zone ဟာ public zone ဖြစ်တဲ့အတွက် အခု လက်ရှိ active ဖြစ်ပြီး ဘယ် zone နဲ့အလုပ်လုပ်နေသလဲဆိုရင် သေချာပါတယ်။ public zone နဲ့ဖြစ်ပါတယ်။

```
# firewall-cmd --get-active-zones
```

Get-active-zones ဆိုတဲ့ option လေးက ခုလက်ရှိ active ဖြစ်နေတဲ့ zone ကိုဖော်ပြပါလိမ့်မယ်။

```
[root@server ~]# firewall-cmd --get-active-zones
public
  interfaces: enp0s3
```

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

ခုလက်ရှိ zone ကတော့

```
[root@server ~]# firewall-cmd --get-default-zone
public
```

Public zone ဆိုတာတွေ့ရမှာပါ။

Public zone ကနေ ကျွန်တော်တို့ အသုံးပြုချင်တဲ့ ထားချင်တဲ့ default zone ကိုလည်း သတ်မှတ်ပေးနိုင်ပါတယ်။

```
# firewall-cmd --set-default-zone=trusted
```

```
[root@server ~]# firewall-cmd --set-default-zone=trusted
success
[root@server ~]# firewall-cmd --get-default-zone
trusted
[root@server ~]# firewall-cmd --get-active-zones
trusted
  interfaces: enp0s3
[root@server ~]#
```

အိုကေ နောက်ထပ် part မှာ rich rules, port range , port forwarding, masquerading, icmp packet filters စတဲ့ rules တွေကို အသုံးပြုမယ့် network interface အလိုက် လေ့လာကြရမှာဖြစ်ပါတယ်။