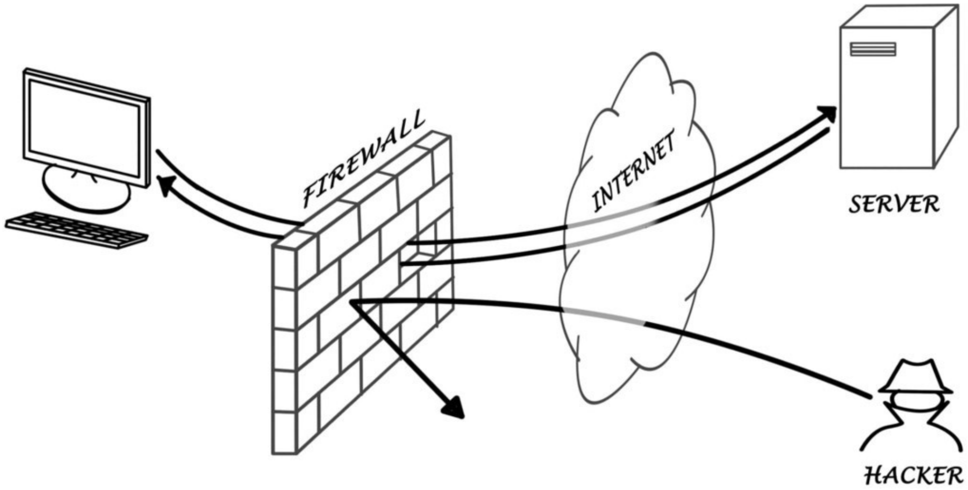
# IV. Firewall

***Feature Description***

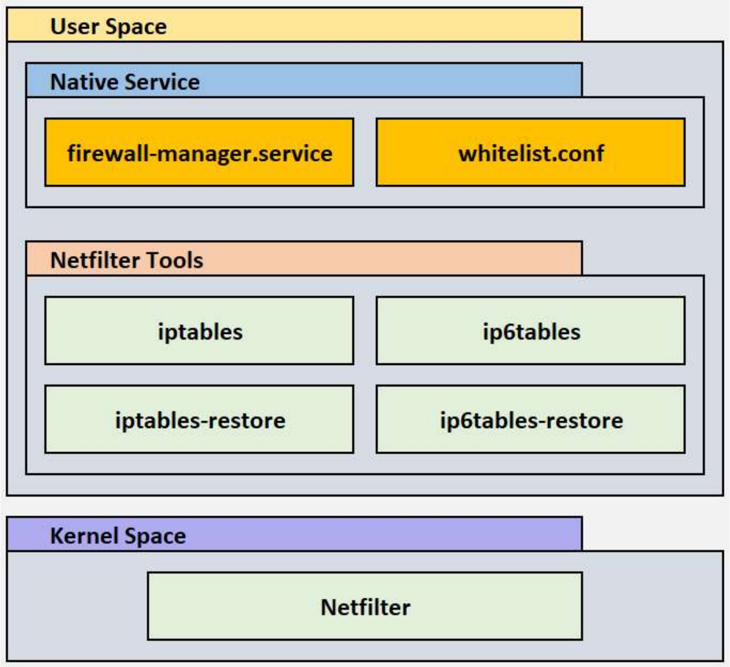


- Protect network system from network threats and external attacks.
- Tracking and scans all the packets and accordingly ACCEPT, REJECT, or DROP the packet, depending upon the rules configured on it.
- Detection and Prevention of Attacks.
- Logging and Monitoring.
- Firewall Technology:
  - Packet Filtering ( IP Address, protocols, Port, Incoming/Outgoing packet)
  - Connection Tracking ( INVALID, ESTABLISHED, NEW, RELATED )
  - Stateful Inspection Firewall (keep track of the state of a connection )

# IV. Firewall

## *Feature Description*

- ***Module Diagram***:

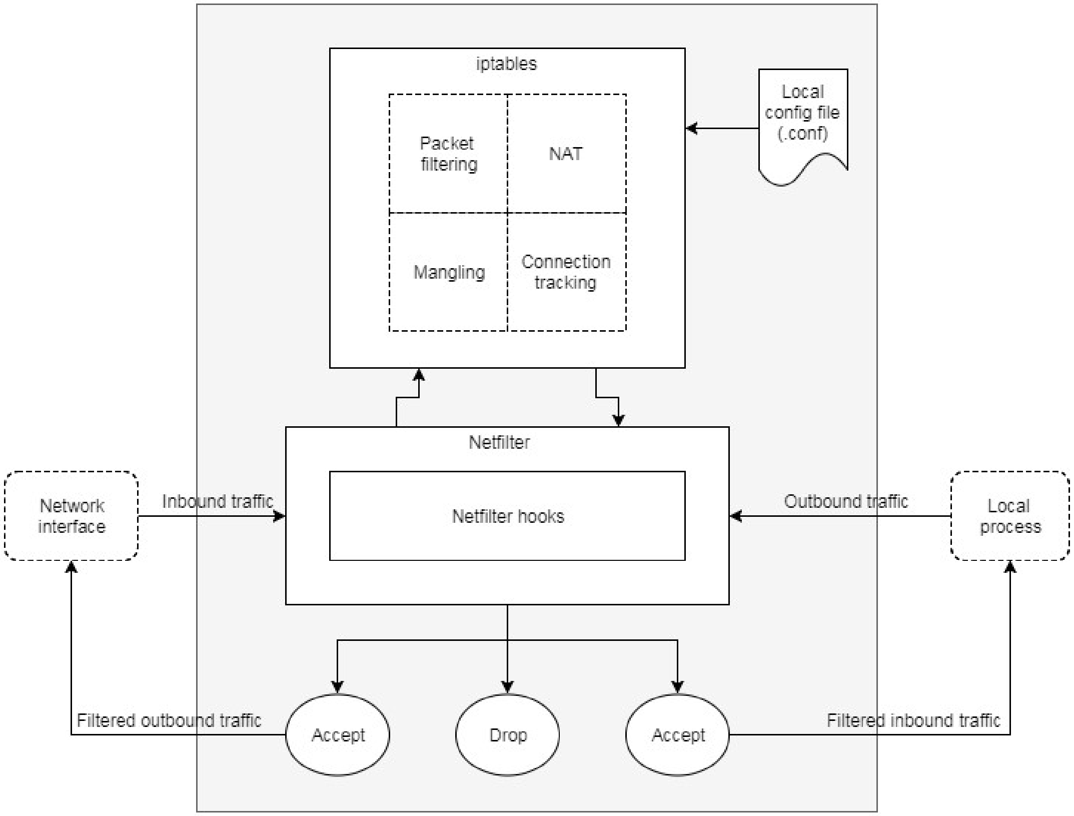


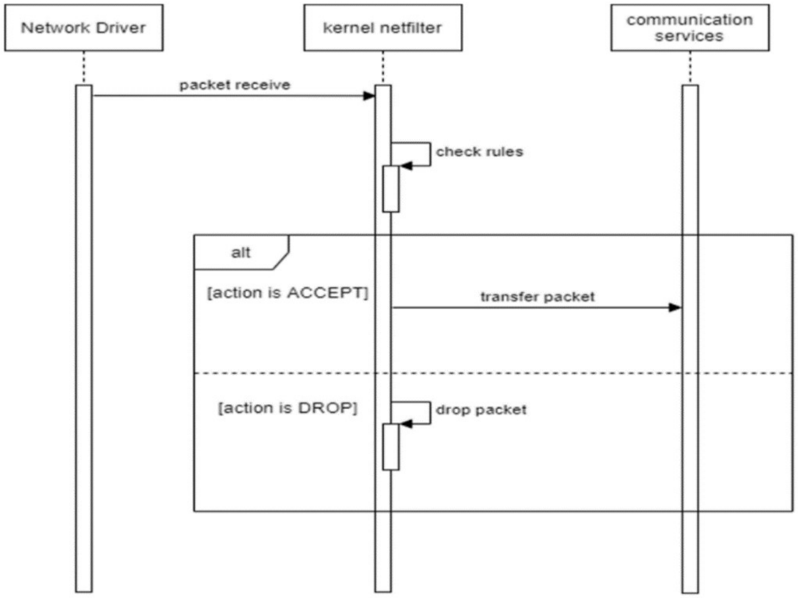
- **firewall-manager.service: load rules when system starts up.**
- **whitelist.conf: Network Firewall rule.**
- **iptables: Tool for IPv4 packet filtering and NAT.**
- **ip6tables: Tool for IPv6 packet filtering and NAT.**
- **iptables-restore: Binary file to load rules (for IPv4).**
- **ip6tables-restore: Binary file to load rules (for IPv6).**

# IV. Firewall

## *Feature Description*

- ***Work Follow*:**

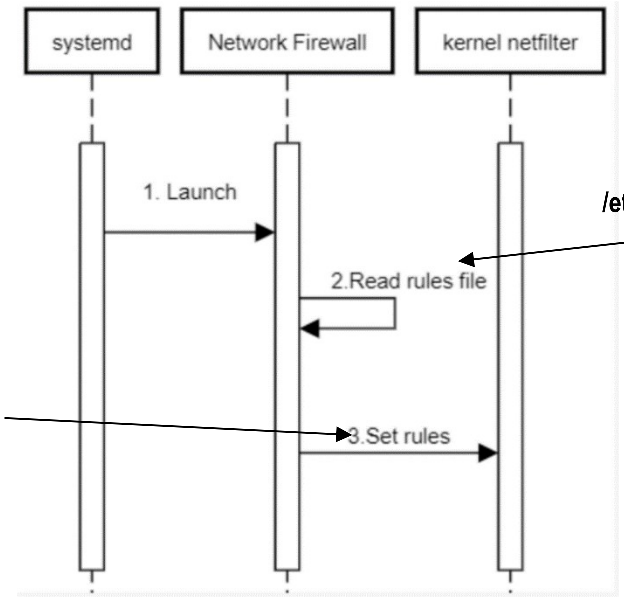




# IV. Firewall

## *Feature Description*

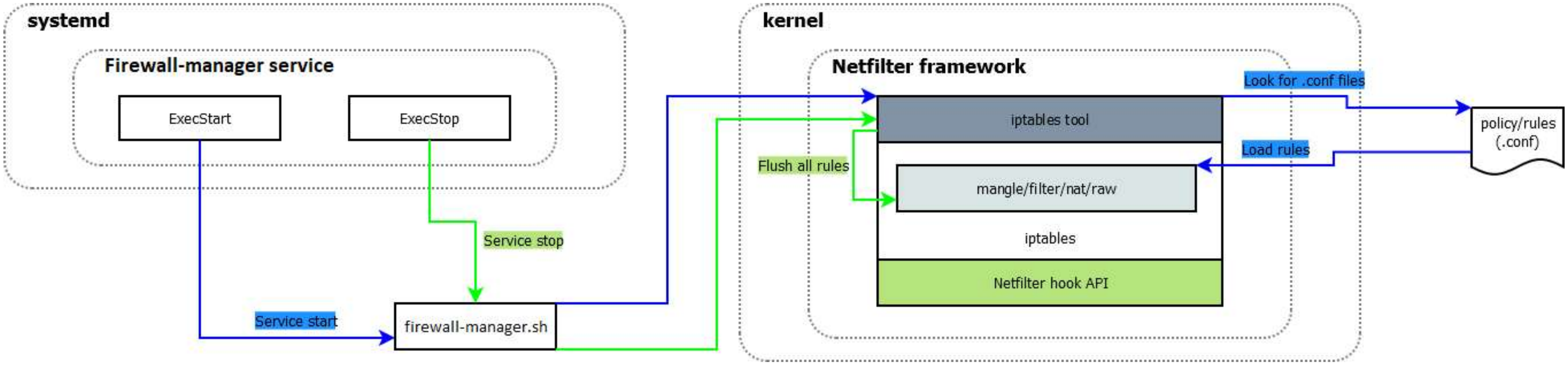
- ***Firewall-manager Service*:**



/etc/../firewall-manager.conf

iptables-restore<*.conf”

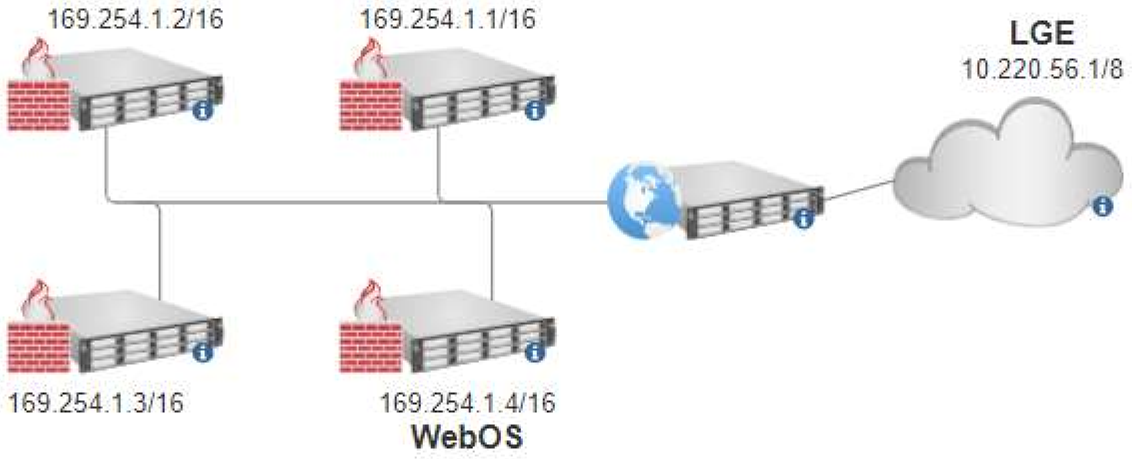
ip6tables-restore<*.conf

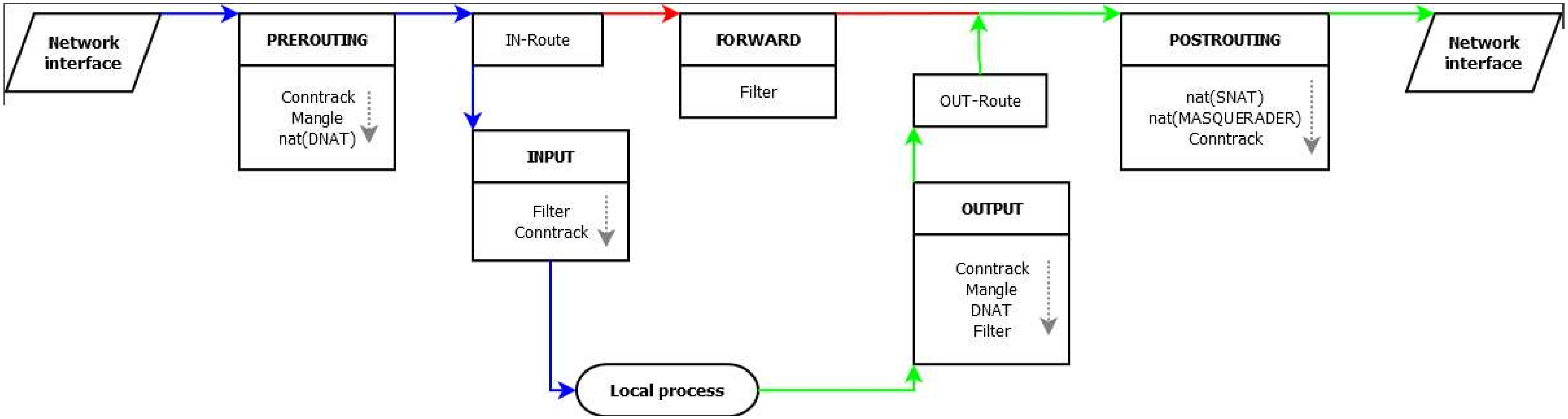


# IV. Firewall

## *Design and Imlpementation*

- ***Firewall Design***:



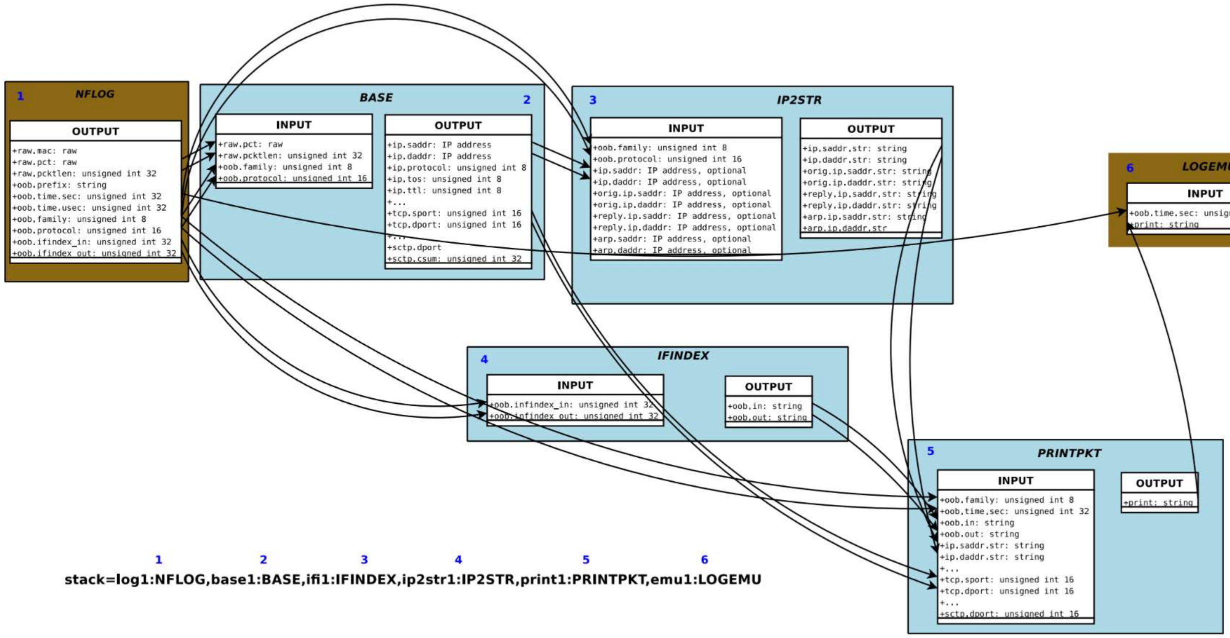


| Tables | Chain | Target |
|---|---|---|
| Filter | INPUT, OUTPUT, FORWARD | ACCEPT, DROP |

# IV. Firewall

## *Design and Imlpementation*

- ***Firewall Design***:
  - DROP all packets except for rules based on allowlist. Filter by TCP/UDP ports ( 22, 53, 80, 443,… )
  - Disable Port scanning ( “open” or “filtered” or “open|filtered” )
  - Connection tracking ( NEW, ESTABLISHED, INVALID )
  - DOS/DDOS Attack Protection (SYN Flood, traffic rate limiting, Request Limiting, Limit connections per source IP, Drop packets on INVALID state, Block packets with bogus TCP flags ).
  - Logging of security violations
- ***Ulogd Design***:

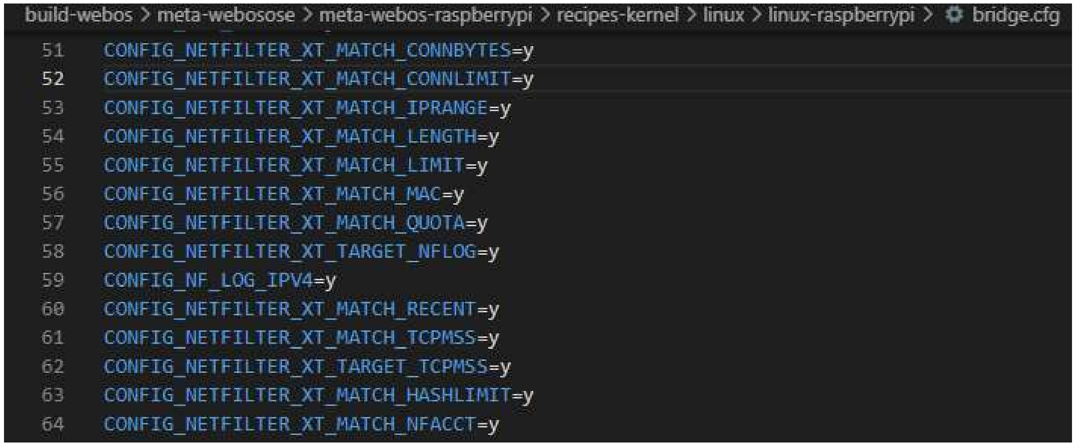

1
NFLOG
OUTPUT
+raw.mac: raw
+raw.pct: raw
+raw.pcktlen: unsigned int 32
+oob.prefix: string
+oob.time.sec: unsigned int 32
+oob.time.usec: unsigned int 32
+oob.family: unsigned int 8
+oob.protocol: unsigned int 16
+oob.ifindex_in: unsigned int 32
+oob.ifindex_out: unsigned int 32
BASE
2
INPUT
+raw.pct: raw
+raw.pcktlen: unsigned int 32
+oob.family: unsigned int 8
+oob.protocol: unsigned int 16
OUTPUT
+ip.saddr: IP address
+ip.daddr: IP address
+ip.protocol: unsigned int 8
+ip.tos: unsigned int 8
+ip.ttl: unsigned int 8
+...
+tcp.sport: unsigned int 16
+tcp.dport: unsigned int 16
+...
+sctp.dport
+sctp.csum: unsigned int 32
3
IP2STR
INPUT
+oob.family: unsigned int 8
+oob.protocol: unsigned int 16
+ip.saddr: IP address, optional
+ip.daddr: IP address, optional
+orig.ip.saddr: IP address, optional
+orig.ip.daddr: IP address, optional
+reply.ip.saddr: IP address, optional
+reply.ip.daddr: IP address, optional
+arp.saddr: IP address, optional
+arp.daddr: IP address, optional
OUTPUT
+ip.saddr.str: string
+ip.daddr.str: string
+orig.ip.saddr.str: string
+orig.ip.daddr.str: string
+reply.ip.saddr.str: string
+reply.ip.daddr.str: string
+arp.ip.saddr.str: string
+arp.ip.daddr.str
6
LOGEMU
INPUT
+oob.time.sec: unsigned int 32
+print: string
4
IFINDEX
INPUT
+oob.infindex_in: unsigned int 32
+oob.infindex_out: unsigned int 32
OUTPUT
+oob.in: string
+oob.out: string
5
PRINTPKT
INPUT
+oob.family: unsigned int 8
+oob.time.sec: unsigned int 32
+oob.in: string
+oob.out: string
+ip.saddr.str: string
+ip.daddr.str: string
+...
+tcp.sport: unsigned int 16
+tcp.dport: unsigned int 16
+...
+sctp.dport: unsigned int 16
OUTPUT
+print: string
1
2
3
4
5
6

stack=log1:NFLOG,base1:BASE,ifi1:IFINDEX,ip2str1:IP2STR,print1:PRINTPKT,emu1:LOGEMU

# IV. Firewall

## *Design and Imlpementation*

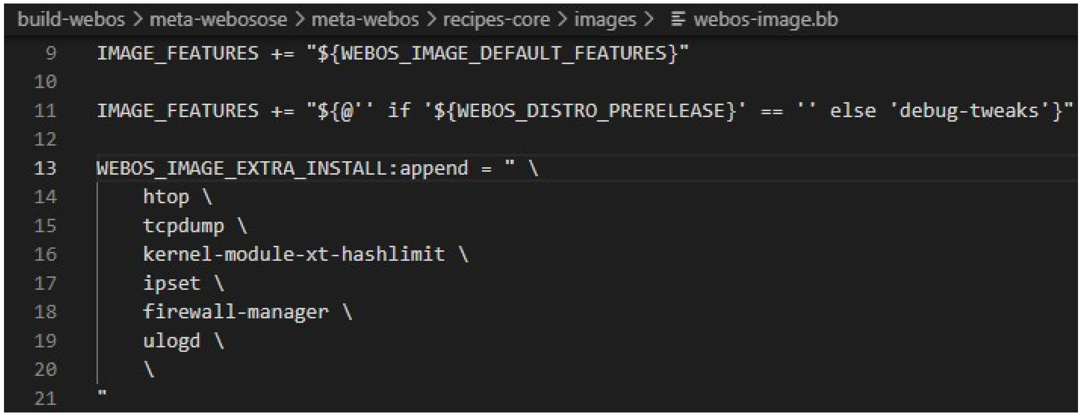
- Implementation:
  - Check and install kernels configuration (NFLOG, Netfilter limit).



build-webos > meta-webosose > meta-webos-raspberrypi > recipes-kernel > linux > linux-raspberrypi > bridge.cfg

```
CONFIG_NETFILTER_XT_MATCH_CONNBYTES=y
CONFIG_NETFILTER_XT_MATCH_CONNLIMIT=y
CONFIG_NETFILTER_XT_MATCH_IPRANGE=y
CONFIG_NETFILTER_XT_MATCH_LENGTH=y
CONFIG_NETFILTER_XT_MATCH_LIMIT=y
CONFIG_NETFILTER_XT_MATCH_MAC=y
CONFIG_NETFILTER_XT_MATCH_QUOTA=y
CONFIG_NETFILTER_XT_TARGET_NFLOG=y
CONFIG_NF_LOG_IPV4=y
CONFIG_NETFILTER_XT_MATCH_RECENT=y
CONFIG_NETFILTER_XT_MATCH_TCPMSS=y
CONFIG_NETFILTER_XT_TARGET_TCPMSS=y
CONFIG_NETFILTER_XT_MATCH_HASHLIMIT=y
CONFIG_NETFILTER_XT_MATCH_NFACCT=y
```

  - Install package to webos-image ( htop, tcpdump, hashlimit, ulogd, firewall-manager )



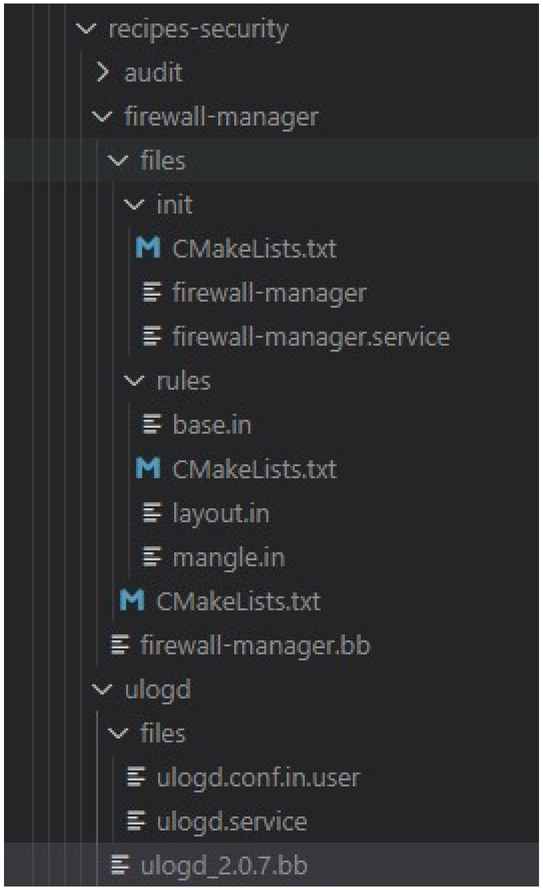
build-webos > meta-webosose > meta-webos > recipes-core > images > webos-image.bb

```
IMAGE_FEATURES += "${WEBOS_IMAGE_DEFAULT_FEATURES}"

IMAGE_FEATURES += "${@'' if '${WEBOS_DISTRO_PRERELEASE}' == '' else 'debug-tweaks'}"

WEBOS_IMAGE_EXTRA_INSTALL:append = " \
    htop \
    tcpdump \
    kernel-module-xt-hashlimit \
    ipset \
    firewall-manager \
    ulogd \
    \
"
```

# IV. Firewall

## ***Design and Imlpementation***

- **Implementation Firewall rules ( basic and advance ), Ulogd**



```
recipes-security
  audit
  firewall-manager
    files
      init
        CMakeLists.txt
        firewall-manager
        firewall-manager.service
      rules
        base.in
        CMakeLists.txt
        layout.in
        mangle.in
      CMakeLists.txt
    firewall-manager.bb
  ulogd
    files
      ulogd.conf.in.user
      ulogd.service
    ulogd_2.0.7.bb
```

# IV. Firewall

*__Demo__*

- **Service status.**
- **Load rules.**
- **Port scanning.**
- **DOS/DDOS Attack Protection.**
  - **SYN flood attack:** *hping3 -S -p 80 169.254.160.86 --flood*
  - **Random Source Attack:** *hping3 -S -p 80 169.254.160.86 --flood --rand-source*
  - **Smurf Attack***: hping3 --icmp --flood --spoof 169.254.153.216 169.254.160.86*
  - **LAND Attack***: hping3 -S -p 80 169.254.160.86 -a 169.254.160.86*