https://en.m.wikipedia.org/wiki/PF_(firewall)

PF

- Available on OpenBSD, FreeBSD, and MacOS (10.7+)
- pflog (pcap based log format and interface.)
- Initial release 1/12/2001
- The basis for the things like m0n0wall and pfsense
- Intuitive configuration language

```
pfctl -s all | tee pf.conf

ether=en0
set skip on lo0
table <local> persist file "/etc/auth_ips"
block drop on $ether log all
pass on $ether from <local>
pass on $ether proto tcp from 134.60.0.0/16 to port 22
pass on $ether proto tcp from 134.60.0.0/16 to port 5900
pass inet proto icmp from 134.60.0.0/16 icmp-type echoreq

pfctl -f pf.conf

ifconfig pflog0 create

tcpdump -n -e -ttt -i pflog0

listening on pflog0, link-type PFLOG (OpenBSD pflog file), snapshot length 524288 bytes
 00:00:00.000000 rule 0/0(match): block in on bridge100: 192.168.64.5.22 >
192.168.64.1.50421: Flags [.], ack 1472325293, win 1761, options [nop,nop,TS val
3104429824 ecr 3500659065], length 0
  0x0000:  3d02 0100 6272 6964 6765 3130 3000 0000  =...bridge100...
  0x0010:  0000 0000 0000 0000 0000 0000 0000 0000  ................
  0x0020:  0000 0000 0000 0000 ffff ffff ffff ff7f  ................
  0x0030:  a086 0100 0000 0000 8de8 0000 0100 0000  ................
  0x0040:  4510 0034 7541 4000 4006 c41b c0a8 4005  E..4uA@.@.....@.
  0x0050:  c0a8 4001 0016 c4f5 a0c4 34e7 57c1 e6ad  ..@.......4.W...
  0x0060:  8010 06e1 6332 0000 0101 080a b909 d700  ….c2..........
  0x0070:  d0a7 d179                                ...y
```

https://www.openbsdhandbook.com/pf/cheat_sheet/

https://en.m.wikipedia.org/wiki/Iptables

IPTables

- Available on Linux since 1998
- ipchains (pre-1998)
- Linux Netfilter
- Firewalld, UFW, libvirt, Docker use IPTables; rudimentary support
- IPv6 is an afterthought

```
iptables -P INPUT DROP

iptables -A INPUT -m state --state ESTABLISHED -j ACCEPT

iptables -L INPUT
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere             state ESTABLISHED

iptables -I INPUT 1 -m state --state NEW -j ACCEPT

iptables -L INPUT
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere             state NEW
ACCEPT     all  --  anywhere             anywhere             state ESTABLISHED

iptables-save | tee rules.persist
*filter
:INPUT DROP [31:7573]
:FORWARD DROP [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state NEW -j ACCEPT
-A INPUT -m state --state ESTABLISHED -j ACCEPT
COMMIT

iptables -F ; iptables -X ; iptables-restore < rules.persist
```