Palomuureista

- *Palomuurilla* tarkoitetaan sekä

  (1) Erillistä (fyysistä tai virtuaalista) konetta, joka välittää, suodattaa ja muokkaa kahden tai useamman muun koneen (tai verkon) välistä liikennettä (”rautapalomuuri”) että

  (2) Ohjelmaa tai ohjelmistoa, joka suodattaa ja muokkaa yhden koneen ja ulkomaailman välistä liikennettä (”softapalomuuri”).
- Linux-ympäristöissä molempien alla on yleensä Netfilter-projekti (www.netfilter.org) ja erityisesti **iptables** (ja mahdollisesti ip6tables).
- Korkeamman tason palomuuriratkaisuja, jotka tarjoavat helpomman käyttöliittymän iptables'in päälle, on paljon, esimerkiksi UFW (Ubuntun softapalomuuri) ja IpFire (Linux-pohjainen rautapalomuuriratkaisu).

iptables

- Linuxin (kernelin!) peruspalomuuriratkaisu IPv4-ympäristöön (IPv6:lle on vastaava ip6tables)
- Käsittelee *paketteja*: ei toimi korkeamman tason protokollien kanssa, ts. ei analysoi eikä yhdistele korkeammalla tasolla yhteen kuuluvia paketteja (ainakaan hyvin, joitakin virityksiä siihen on)
- Hyvin monipuolinen ja monimutkainenkin, kaikkien ominaisuuksien tehokas käyttö edellyttää TCP/IP:n ja siihen liittyvien protokollien syvällistä ymmärtämistä, perustilanteet kuitenkin kohtuuhelppoja
- Täydellinen (...) dokumentaatio http://www.iptables.info
- http://rlworkman.net/howtos/iptables/iptables-tutorial.html on hiukan vanhentunut mutta helppolukuisempi ja yhä olennaisesti oikea ja hyödyllinen kuvaus iptablesin toiminnasta

iptables: termejä

- drop/deny: paketti ”pudotetaan lattialle”, hävitetään eikä tehdä muuta
- reject: paketti hävitetään mutta lähettäjälle vastataan ja kerrotaan että näin kävi
- accept: paketti hyväksytään ja päästetään eteenpäin
- state: paketin tila suhteessa pakettivirtaan (new, established, related, invalid)
- chain (ketju): paketteihin sovellettava sääntöjoukko; sanotaan että paketti kulkee ketjun läpi (traverses through chain). Joukko vakioketjuja (INPUT jne), minkä lisäksi voi määritellä omia.
- table: taulu, johon palomuurisääntöjä talletetaan (raw, nat, mangle, filter)
- rule (sääntö): kokoelma ehtoja ja niiden toteutuessa suoritettava kohde
- match (ehto): säännössä ehto jonka pitää toteutua; käytetään myös koko säännöstä, jos kaikki sen ehdot toteutuvat
- target (kohde): säännössä kertoo mitä tehdään jos ehdot toteutuvat
- jump (hyppy): kohde, joka määrää hyppäämään toiseen ketjuun
- connection tracking: yhteyksien seuranta
- policy: oletustoimenpide sääntöjen loppuessa (drop, reject, accept)

Ketjut

- Vakioketjut ovat:

  INPUT: käsittelee paketit, joiden (lopullinen) määränpää on palomuurikone itse

  OUTPUT: koneesta ulos lähtevien (siellä syntyneiden) pakettien käsittely

  PREROUTING: pakettien käsittely ennen reititystä (päätöstä määränpäästä)

  POSTROUTING: pakettien käsittely reitityksen (ja yleensä suodatuspäätösten) jälkeen

  FORWARD: paketit, joiden lähde ja määränpää palomuurin ulkopuolella
- Kunkin ketjun sisällä käydään siihen kuuluvat taulut läpi järjestyksessä

  raw → mangle → nat → filter

  (kaikissa ketjuissa ei ole kaikkia tauluja)
- Mahdolliset omat ketjut toimivat ikäänkuin aliohjelmina muille ketjuille

Taulut

- Neljä taulua:

  RAW: lähinnä haluttaessa ohittaa yhteysseuranta joillekin paketeille (kohde NOTRACK)

  MANGLE: paketin ohjauskenttien muuttamiseen (kohteet TOS, TTL, MARK, SECMARK, CONNSECMARK)

  FILTER: pakettien suodattamiseen, ts. hyväksymiseen tai hylkäämiseen niitä muuttamatta (kohteet DROP, REJECT, ACCEPT)

  NAT: pakettien osoitekenttien (IP ja porttinumero) muuttamiseen (kohteet DNAT, SNAT, MASQUERADE, REDIRECT)
- Useimmissa ketjuissa ei ole kaikkia tauluja, ja useimmiten käytetään joka tapauksessa vain FILTER- ja NAT-tauluja

Kohteet

- Joukko sisäänrakennettuja vakiokohteita, yleisimmät:

  ACCEPT: paketti hyväksytään, ketjun käsittely lopetetaan

  DROP: paketti hylätään, ketjun käsittely lopetetaan

  REJECT: paketti torjutaan, ketjun käsittely lopetetaan

  LOG: paketti lokitetaan, ketjun käsittely jatkuu

  - paljon muitakin, ks. http://www.iptables.info/en/iptables-targets-and-jumps.html
  - Kutsu aina -j *kohde*
- Omia ketjuja voi myös käyttää kohteina
  - Kaksi tapaa kutsua:

    -j *ketju:* palaa nykyiseen ketjuun kutsutun ketjun päätyttyä

    -g *ketju*: lopettaa nykyisen ketjun kutsutun päätyttyä (palaa ylemmälle tasolle)

Taulut ja ketjut

- Kolme perustilannetta:
  - Saapuva paketti (kohde palomuurikone itse - huom. tässä ”palomuurikone” voi olla esim. vain omia pakettejaan käsittelevä palvelin):

    PREROUTING → INPUT
  - Lähtevä paketti (luotu palomuurikoneessa):

    OUTPUT → POSTROUTING
  - Välitettävä paketti:

    PREROUTING → FORWARD → POSTROUTING
- Ketjujen käsittelyn välissä tehdään reitityspäätökset
- Kaikissa tilanteissa ei käytetä kaikkia tauluja, mutta niitä joita käytetään käytetään aina samassa järjestyksessä

Saapuva paketti

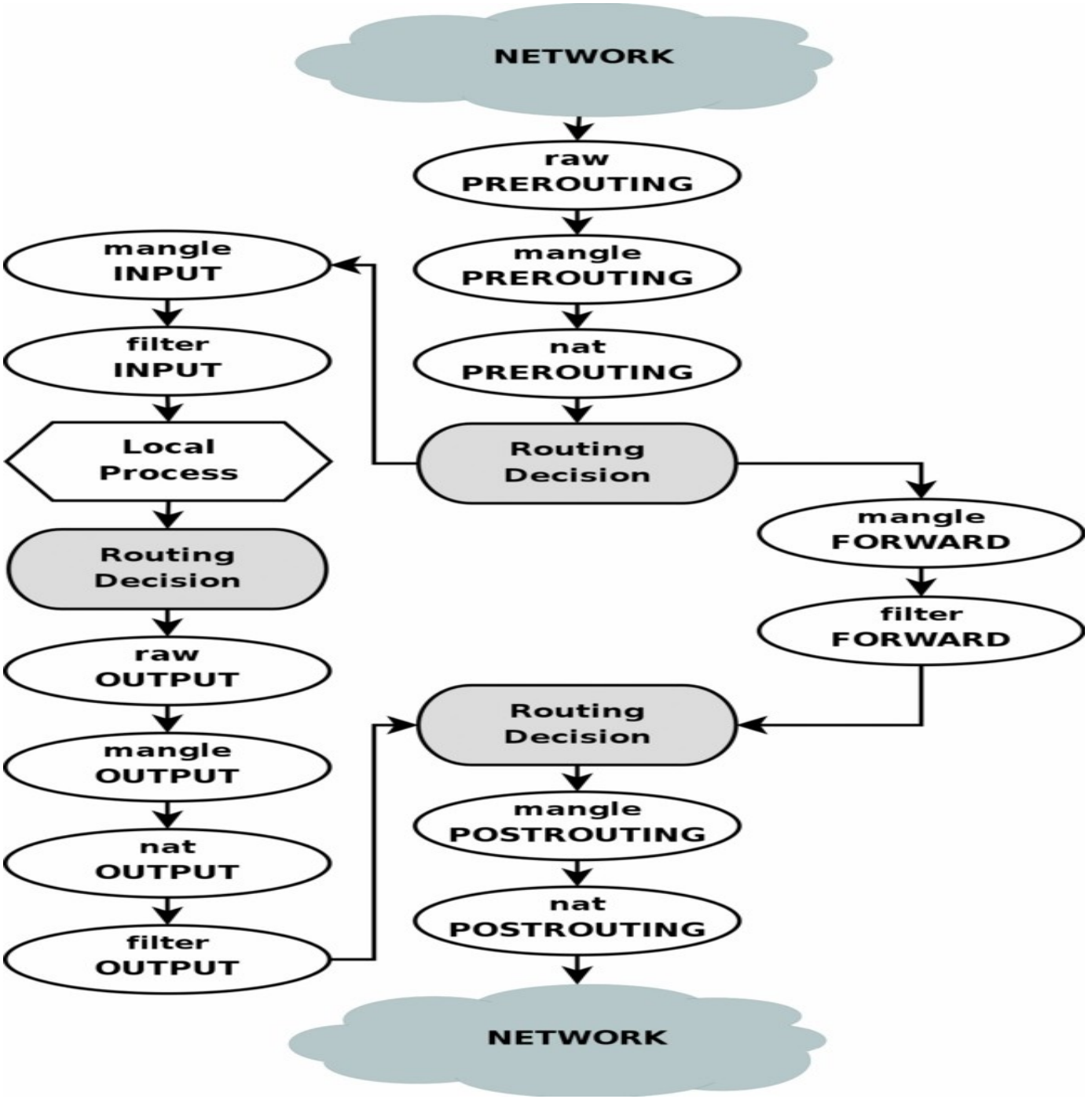
- Palomuurikoneelle osoitetun paketin käsittelyvaiheet:
  - PREROUTING - raw
  - yhteydenseurantakoodin käsittely
  - PREROUTING - mangle
  - PREROUTING - nat (DNAT)
  - reitityspäätös
  - INPUT - mangle
  - INPUT - filter

Lähtevä paketti

- Palomuurikoneelta lähtevän paketin käsittelyvaiheet:
  - reitityspäätös
  - OUTPUT - raw
  - yhteydenseurantakoodin käsittely
  - OUTPUT - mangle
  - OUTPUT - nat
  - OUTPUT - filter
  - reitityspäätös
  - POSTROUTING - mangle
  - POSTROUTING - nat (SNAT)

Välitettävä paketti

- Palomuurikoneen läpi kulkevan paketin käsittelyvaiheet:
  - PREROUTING - raw
  - yhteydenseurantakoodin käsittely
  - PREROUTING - mangle
  - PREROUTING - nat (DNAT)
  - reitityspäätös
  - FORWARD - mangle
  - FORWARD - filter
  - POSTROUTING - mangle
  - POSTROUTING - nat (SNAT)


NETWORK
raw
PREROUTING
mangle
PREROUTING
nat
PREROUTING
Routing
Decision
mangle
INPUT
filter
INPUT
Local
Process
Routing
Decision
raw
OUTPUT
mangle
OUTPUT
nat
OUTPUT
filter
OUTPUT
mangle
FORWARD
filter
FORWARD
Routing
Decision
mangle
POSTROUTING
nat
POSTROUTING
NETWORK

iptables -komento

- Palomuuritauluja käsitellään iptables-komennolla:

  iptables [-t table] -A ketju sääntö # --append, lisäys ketjun loppuun

  iptables [-t table] -I ketju [numero] sääntö # --insert, lisäys ketjun keskelle

  iptables [-t table] -R ketju numero sääntö # --replace, säännön muutos

  iptables [-t table] -D ketju {sääntö|numero} # --delete, säännön poisto

  iptables [-t table] -L [ketju] # --list, sääntöjen tulostus

  iptables [-t table] -S ketju [numero] # --list-rules, sääntöjen tulostus komentomuodossa

  iptables [-t table] -F [ketju] # --flush, ketjun (tai kaikkien) tyhjennys

  iptables [-t table] -Z [ketju [numero]] # --zero, laskurien nollaus

  iptables [-t table] -N ketju # --new-chain, uuden ketjun luonti

  iptables [-t table] -X ketju # --delete-chain, tyhjän ketjun poisto

  iptables [-t table] -P ketju kohde # --policy, oletuskohteen määritys

  iptables [-t table] -E ketju1 ketju2 # --rename-chain, ketjun nimen vaihto

- Oletustaulu (ilman -t -optiota) aina filter. Lisäksi yleisiä optioita:

  -v, --verbose; -n, --numeric; -x, --exact; --line-numbers; --modprobe

iptables: sääntömäärittelyt

- Perussääntömäärittelyt (! = negaatio, ”ei”):

  [!] -p, --protocol {tcp|udp|udplite|icmp|esp|ah|sctp|all}

  [!] -s, --source *address[/mask][...]*

  [!] -d, --destination *address[/mask][...]*

  -j, --jump *kohde*

  -g, --goto *ketju*

  [!] -i, --interface *laite*

  [!] -o, --out-interface *laite*

  [!] -f, --fragment

  -c, --set-counters *paketteja tavuja*

- Lisäksi on laajennuksia (extensions), erityisesti *match*

iptables: match extensions

- Erilaisia laajennuksia on tarjolla moduleina, jotka ladataan optiolla -m tai --match (jotkut latautuvat myös -p:n sivuvaikutuksena). Seuraavassa esitellään vain pieni valikoima.

  -p tcp {--sport *port[:port]*|--dport *port[:port]*|--tcp-flags *...*|--syn|--tcp-option *n}*

  -p udp {--sport *port[:port]*|--dport *port[:port]}*

  -p icmp --icmp-type *type*

  -m state --state {NEW|ESTABLISHED|RELATED|INVALID}

  -m iprange {--src-range|--dst-range} *from[-to]*

  -m connlimit [--connlimit_mask *n]* --connlimit-above *n*

  -m limit --limit *rate*[/second|/minute|/hour|/day]

  -m limit --limit-burst *n*

  -m mac [!] --mac-source address

iptables: esimerkki 1

- Halutaan sallia kaikki yhteydet ulos ja sisään kaikki *paitsi* porttia 25 (smtp) mistään, ssh sallitaan vain JY:n (130.234.0.0/16) alueelta eikä mitään alueelta 134.170.0.0/16:

```
iptables -F
iptables -P INPUT ACCEPT
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP
iptables -A INPUT -p tcp --dport 25 -j DROP
iptables -A INPUT -p tcp ! -s 130.234.0.0/16 --dport 22 -j DROP
iptables -A INPUT -s 123.170.0.0/16 -j DROP
```

- Tällainen ”blacklisting” tyyli, että sallitaan kaikki mitä ei ole erikseen kielletty, ei yleensä ole hyvä idea palvelimeen sisääntulevan liikenteen rajoittamiseen. Rajapalomuurin FORWARD-säännöissä se voi joskus olla järkevä ratkaisu, jos esimerkiksi asiakkaille halutaan sallia oletuksena melkein kaikki (esim. kaikki paitsi smtp-portti 25, joka yleensä halutaan sulkea spämmerien takia), muulloin ”whitelisting” (kiellettyä kaikki mitä ei erikseen sallita) on yleensä parempi.
- Interaktiivisesti huutomerkki voi aiheuttaa ongelmia shellin kanssa (set +H auttaa)

Yhteyksien seuranta

- Connection tracking, ”state machine”; iptables on ”stateful firewall”, pitää muistissa yhteyksien tiloja eikä vain käsittele jokaista pakettia uutena
- Yhteyksillä voi olla neljä (viisi) tilaa:
  - NEW: yhteyden ensimmäinen paketti (ei kuulu mihinkään muistissa olevaan yhteyteen)
  - ESTABLISHED: paketti kuuluu muodostettuun (ja muistettuun) yhteyteen (liikennettä nähty molempiin suuntiin), vastaus aiemmin lähetettyyn pakettiin
  - RELATED: olemassaolevaan ESTABLISHED-yhteyteen liittyvä mutta ei siihen kuuluva paketti (aikaisemman yhteyden luoma uusi yhteys)
  - INVALID: tilaa ei ole tai sitä ei tunnisteta (koneesta muisti loppunut tai jotain muuta ikävää)
  - UNTRACKED: ei varsinaisesti tila vaan merkki, että tämän paketin tilaa ei seurata (pakettikohtainen, ei yhteyskohtainen)
- Tilaa voidaan säännöissä testata tyyliin

  -m state --state *state*

iptables: esimerkki 2

- Sallitaan sisään http (portti 80) ja https (443) kaikkialta sekä ssh (22) alueelta 130.234.0.0/16, ei rajoiteta ulos lähtevää liikennettä:

```
iptables -F
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP
iptables -A INPUT -m state --state ESTABLISHED -j ACCEPT
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
iptables -A INPUT -p tcp -s 130.234.0.0/16  --dport 22 -j ACCEPT
iptables -A INPUT -j REJECT --reject-with icmp-port-unreachable
```

- Paluuliikenne (ESTABLISHED) pitää sallia erikseen; senkin voisi rajata haluttuihin portteihin.
- Tässä siis käytetään vain filter-taulua (joka on oletus).
- Viimeinen REJECT ei ole välttämätön, antaa vain siistin virheen.
- Huom. mm. ftp ulospäin ei toimi näillä säännöillä, koska sen paluuliikennettä ei tunnisteta ESTABLISHED-liikenteeksi.

iptables: oikoteitä

- Osoitteet voivat olla yksittäisiä tai maskilla rajattuja aliverkkoja
- Useita osoitteita voi antaa pilkulla erottaen (130.234.0.0/16,172.20.208.17) - iptables jakaa tällaiset sisäisesti kahtia (ja iptables -L jne tulostavat ne kahtena)
- Modulilla iprange voidaan antaa osoitevälejä tyyliin

  ... -m iprange --src-range 172.20.208.10-172.20.209.120
- Yksittäisen portin lisäksi voidaan antaa kaksoispisteellä erotettu väli, esim. 49152:59999; jos välin toisen pään jättää pois, se tarkoittaa kaikkia siitä eteenpäin, esim. 1024: on portit 1024-65535
- Normaalisti samalla kertaa ei voi antaa useita ei-peräkkäisiä portteja, mutta tarkoitusta varten on lisämoduli multiport:

  ... -m multiport --dports 80,443 -j ACCEPT
- Porttinumeroiden asemesta voi usein käyttää palveluiden lyhenteitä (ne katsotaan /etc/services -tiedostosta), mutta se hidastaa sääntöjen latausta joskus merkittävästikin

iptables: esimerkki 3

- Kuten esimerkki 1, mutta sallitaan ssh myös 172.20.0.0/16-alueelta ja ulospäin vain dns (portti 53) ja apt-proxy (3142):

```
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP
iptables -A INPUT  -m state --state ESTABLISHED -j ACCEPT
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
iptables -A INPUT -p tcp -s 130.234.0.0/16,172.20.0.0/16  --dport 22 -j ACCEPT
iptables -A INPUT -j REJECT --reject-with icmp-port-unreachable
iptables -A OUTPUT -m state --state ESTABLISHED -j ACCEPT
iptables -A OUTPUT -p udp --dport 53 -j ACCEPT
iptables -A OUTPUT -p tcp --dport 53 -j ACCEPT
iptables -A OUTPUT -p tcp -d 172.20.208.17 --dport 3142 -j ACCEPT
```

- Ilman nimipalvelua juuri mikään ei toimi; nykyisin syytä sallia sekä udp- että tcp-portit.
- Myös OUTPUT-ketjussa pitää nyt sallia paluuliikenne erikseen.

iptables: icmp

- ICMP-paketeista yleensä halutaan sallia ainakin 3 (destination-unreachable) ja 11 (time-exceeded), usein myös 12 (parameter-problem) ja 13 (timestamp-request, mm. openvpn käyttää sitä):

  iptables -A INPUT -p icmp -m icmp --icmp-type 3 -j ACCEPT

  iptables -A OUTPUT -p icmp -m icmp --icmp-type destination-unreachable -j ACCEPT

  ...

- Ulospäin ICMP:n voi yleensä sallia vapaamminkin (vaikka kaikki pakettityypit)
- Erityisesti icmp echo (8 echo-request, 0 echo-reply) eli ping halutaan usein sallia ainakin ulospäin, sisäänpäin yleensä vain rajoitetusti.
  - Ping ulospäin:

    iptables -A OUTPUT -p icmp --icmp-type 8 -j ACCEPT

    # lisäksi tarvitaan joko em. INPUT... ESTABLISHED-sääntö tai

    iptables -A INPUT -p icmp --icmp-type 0 -j ACCEPT
  - Ping sisäänpäin vastaavasti

    iptables -A INPUT -s 172.20.0.0/16 -p icmp --icmp-type echo-request -j ACCEPT

    # jälleen lisäksi joko OUTPUT...ESTABLISHED tai

    iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEPT

iptables: sääntöjen talletus

- Säännöstön voi tallettaa komennolla

  iptables-save >tiedosto

  ja palauttaa komennolla

  iptables-restore <tiedosto
- Uusia sääntöjä interaktiivisesti testatessa kätevä on

  iptables-apply [-t *seconds*] [*tiedosto*|-c [*skripti*]]

  joka kysyy onnistuiko ja palauttaa vanhat säännöt ellei saa vastausta määräajassa.

  Tiedoston oletusarvo on /etc/network/iptables.up.rules, syntaksi sama kuin iptables-restore'lla, oletusskripti taas on /etc/network/iptables.up.run (se voi suorittaa mitä tahansa komentoja, katsoa koneen oman osoitteen /etc/hosts'ista, tehdä sääntöjä monelle koneelle silmukassa jne; huom. verkko ei aina ole päällä sitä suoritettaessa).
- Jotta säännöt saa pysyviksi bootin yli, ne pitää ajaa päälle jossain käynnistysskriptissä. Vaihtoehtoja on monta, yksi luonteva paikka on /etc/network/interfaces -tiedostossa pre-up -säännöllä (suoritetaan ennen verkkoyhteyden käynnistämistä), esim. jompi kumpi näistä:

  pre-up iptables-restore < /etc/network/iptables.up.rules

  pre-up /etc/network/iptables.up.run

iptables: ftp

- Ftp on palomuureille vaikea protokolla, koska siinä on erotettu ohjaus ja datansiirto eri portteihin ja porttineuvottelut käydään palomuurin näkökulmasta siirrettävän datan sisällä. Sillä on lisäksi kaksi toimintatapaa:
  - Aktiivinen ftp:
    - Asiakas ottaa (ohjaus)yhteyden ftp-palvelimen porttiin 21.
    - Palvelin kertoo asiakkaalle mistä (satunnaisesta) portista datayhteys tulee.
    - Palvelin ottaa (data)yhteyden asiakkaan porttiin 20.
    - Data siirtyy portin 20 kautta edes ja takaisin, ohjauskomennot portin 21 kautta.
  - Passiivinen ftp:
    - Asiakas ottaa (ohjaus)yhteyden ftp-palvelimen porttiin 21.
    - Palvelin ilmoittaa asiakkaalle (valitsemansa satunnaisen) portin datayhteyttä varten.
    - Asiakas ottaa yhteyden palvelimen ilmoittamaan porttiin.
    - Data siirtyy edellä valitun portin kautta, ohjauskomennot taas portin 21 kautta.
- Tämä edellyttää palomuurilta yhteyden seurantaa ja ftp-protokollan tuntevaa modulia (nf_conntrack_ftp), joka lukee ohjausyhteyden pakettien sisältöjä selvittääkseen mikä datayhteys siihen liittyy.

iptables: ftp client

- Ftp-asiakasohjelmaa varten tarvitaan palomuuriin seuraavanlaiset säännöt:

```
modprobe nf_conntrack_ftp
# Yhteyden muodostus: sallitaan yhteydenotto porttiin 21 maailmalla
iptables -A OUTPUT -p tcp --dport 21 -m state --state NEW,ESTABLISHED -j ACCEPT
# sallitaan vastaus (ei tarpeen jos vastaukset yleisemminkin sallittu)
iptables -A INPUT -p tcp --sport 21 -m state --state ESTABLISHED -j ACCEPT
# Aktiivinen ftp: annetaan palvelimen ottaa yhteys takaisin porttiimme 20
iptables -A INPUT -p tcp --sport 20 -m state --state ESTABLISHED,RELATED -j ACCEPT
# ja sallitaan itsemme vastata sille
iptables -A OUTPUT -p tcp --dport 20 -m state --state ESTABLISHED -j ACCEPT
# Passiivinen ftp: sallitaan itsemme ottaa yhteys palvelimen ilmoittamaan porttiin
iptables -A OUTPUT -p tcp --sport 1024: --dport 1024: -m state --state ESTABLISHED,RELATED -j ACCEPT
# ja sallitaan palvelimen vastata
iptables -A INPUT -p tcp --sport 1024: --dport 1024: -m state --state ESTABLISHED -j ACCEPT
```

iptables: ftp server

- Ftp-palvelin vaatii olennaisesti vastaavat säännöt kuin asiakaskin, mutta peilikuvina (vertaa input/output ja dport/sport edelliseen):

```
modprobe nf_conntrack_ftp
# Yhteyden muodostus: sallitaan yhteydenotto omaan porttiimme 21 maailmalta
iptables -A INPUT -p tcp --dport 21 -m state --state NEW,ESTABLISHED -j ACCEPT
# sallitaan itsemme vastata
iptables -A OUTPUT -p tcp --sport 21 -m state --state ESTABLISHED -j ACCEPT
# Aktiivinen ftp: sallitaan itsemme ottaa paluuyhteys portin 20 kautta
iptables -A OUTPUT -p tcp --sport 20 -m state --state ESTABLISHED,RELATED -j ACCEPT
# ja sallitaan vastaukset siihen
iptables -A INPUT -p tcp --dport 20 -m state --state ESTABLISHED -j ACCEPT
# Passiivinen ftp: sallitaan yhteydenotto maailmalta uuteen omaan porttiimme
iptables -A INPUT -p tcp --sport 1024: --dport 1024: -m state --state ESTABLISHED,RELATED -j ACCEPT
# ja sallitaan itsemme vastata
iptables -A OUTPUT -p tcp --sport 1024: --dport 1024: -m state --state ESTABLISHED -j ACCEPT
```

- Passiivisen yhteyden testaus: ftp -p ...