Eine Tour durch OpenBSD

vmann

16. september 2023

Hintergrund

- 1995 von NetBSD geforked
- Halb-jährliche Releases
- Entwickelnde&Security-Fokus
- Heimat von OpenSSH, tmux, LibreSSL, ...

ports vs base

- Base-System ist vollständiges Betriebssystem (nicht nur Kernel)
- Hackbar: ein Source-Tree, ein Build-System (`man release`)
- `release` / `-current` / `-stable`
- `sysupgrade` / `syspatch`

ports vs base

- Base-System ist vollständiges Betriebssystem (nicht nur Kernel)
- Hackbar: ein Source-Tree, ein Build-System (`man release`)
- `release` / `-current` / `-stable`
- `sysupgrade` / `syspatch`

- Dritt-Software aus Ports-Tree
- Vorgebaute Binärpakete (Bauen aus dem Tree erzeugt auch Pakete)
- `pkg_add` / `pkg_delete` / `pkg_info` / `...`

Startup

- Kernel startet `init`
- `init` führt `/etc/rc` aus (ein Shell-Script)
- Konfiguration in `/etc/rc.conf.local`

Startup

- Kernel startet `init`
- `init` führt `/etc/rc` aus (ein Shell-Script)
- Konfiguration in `/etc/rc.conf.local`

```
apmd_flags=-A
httpd_flags=
pkg_scripts=cupsd transmission_daemon
sshd_flags=NO
transmission_daemon_rtable=1
vmd_flags=
xenodm_flags=
```

rcctl

- CLI um Dienste zu verwalten

```
rcctl set transmission_daemon rtable 1
rcctl enable transmission_daemon
rcctl start transmission_daemon
...
```

rc.d

- Deklarative Init-Scripts

```
#!/bin/ksh

daemon="/usr/local/bin/transmission-daemon"
daemon_user="_transmission"

. /etc/rc.d/rc.subr

rc_reload=NO

rc_cmd $1
```

WLAN mit `ifconfig`

- WLAN kann komplett mit `ifconfig` konfiguriert werden (außer 802.1x)

```
ifconfig iwm0 scan
ifconfig iwm0 join Quellcode
ifconfig iwm0 join WICMP wpakey Wint3Rn@t
```

/etc/hostname.if

- Netzwerkkonfiguration beim Startup wird aus `/etc/hostname.if` gelesen (`if` ist z.B. `iwm0`)

```
join Quellcode
join WICMP wpakey Wint3Rn@t
# 802.1x braucht wpa_supplicant :(
join 36C3 wpaakms 802.1x

inet autoconf
inet6 autoconf
```

/etc/hostname.if

- Netzwerkkonfiguration beim Startup wird aus `/etc/hostname.if` gelesen (`if` ist z.B. `iwm0`)

```
join Quellcode
join WICMP wpakey Wint3Rn@t
# 802.1x braucht wpa_supplicant :(
join 36C3 wpaakms 802.1x

inet autoconf
inet6 autoconf
```

- Kann bei Änderungen mit `sh /etc/netstart iwm0` angewendet werden

WireGuard mit `ifconfig`

- WireGuard kann komplett mit ifconfig konfiguriert werden

```
ifconfig wg0 create
ifconfig wg0 \
    wgkey aGllciBrb21tdCBkZXIgcHJpdmF0ZSBrZXkgaGluLgo=
ifconfig wg0 \
    wgpeer HQHCrq4J6bSpdW1fI5hR/bvcrYa6HgGgwaa5ZY749ik= \
    wgendpoint 185.213.155.73 51820 \
    wgaip 0.0.0.0/0 wgaip ::0/0
ifconfig wg0 destroy
```

Routing Domains

- Host kann mehre unabhängige Routingtabellen haben (Routing Domains)
- Interfaces gehören zu einer Routing Domain (default 0)

```
ifconfig wg0 rdomain 1
```

- Programme laufen in einer Routing Domain und nutzen die zugehörige Routingtabelle und Interfaces für Netzwerkverbindungen

```
rcctl set transmission_daemon rtable 1
route -T 1 exec curl ifconfig.co
```

Routing Domains - /etc/hostname.lo1

```
rdomain 1
inet 127.0.0.1/8
```

Routing Domains - /etc/hostname.wg0

```
wgkey aGllciBrb21tdCBkZXIgcHJpdmF0ZSBrZXkgaGluLgo=
wgpeer HQHCrq4J6bSpdW1fI5hR/bvcrYa6HgGgwaa5ZY749ik= \
    wgendpoint 185.213.155.73 51820 \
    wgaip 0.0.0.0/0 wgaip ::0/0

rdomain 1
inet 10.64.6.148/32
inet6 fc00:bbbb:bbbb:bb01::1:693/128
!route -T1 add default 10.64.6.148
!route -T1 add -inet6 default fc00:bbbb:bbbb:bb01::1:693
```

pf

- Stateful
- Letztes `pass`/`block` das matched gilt

pf

- Stateful
- Letztes `pass`/`block` das matched gilt

```
block return    # block stateless traffic
pass out        # establish keep-state
```

pf

- Stateful
- Letztes `pass`/`block` das matched gilt

```
block return    # block stateless traffic
pass out        # establish keep-state

# transmission web ui
pass in on rdomain 0 proto tcp to {127.0.0.1,::1} \
    port 9091 rtable 1
```

pf

- Stateful
- Letztes `pass`/`block` das matched gilt

```
block return    # block stateless traffic
pass out        # establish keep-state

# transmission web ui
pass in on rdomain 0 proto tcp to {127.0.0.1,::1} \
    port 9091 rtable 1

# Port build user does not need network
block return out log proto {tcp udp} user _pbuild
```

OpenSMTPD

- Seit OpenBSD 4.6
- “E-Mail für Menschen”

OpenSMTPD - /etc/mail/smtpd.conf

```
table aliases file:/etc/mail/aliases

listen on socket
listen on lo0

action "local_mail" mbox alias <aliases>
action "outbound" relay

match from local for local action "local_mail"
match from local for any action "outbound"
```

OpenSMTPD - /etc/mail/smtpd.conf

```
table aliases file:/etc/mail/aliases

listen on socket
listen on all

action "local_mail" mbox alias <aliases>
action "outbound" relay

match from any for domain "example.org" action "local_mail"
match from local for local action "local_mail"
match from local for any action "outbound"
```

OpenSMTPD

Empfangen werden benötigt bissl mehr Arbeit ... (Reverse DNS, SPF/DKIM, DMARC)

Mehr Infos:

- `man smtpd.conf`
- Chehade, Gilles (2019). *Setting up a mail server with OpenSMTPD, Dovecot and Rspamd*

httpd

- Früher erst Apache in base, dann nginx
- Maintainance-Aufwand (Patchsets für Priviledge Separation)
- Zu viele Features (mehr Code -> mehr Bugs)

httpd

- Früher erst Apache in base, dann nginx
- Maintainance-Aufwand (Patchsets für Priviledge Separation)
- Zu viele Features (mehr Code -> mehr Bugs)

- Seit OpenBSD 5.6 eigener `httpd`

```
server "example.com" {
    listen on * port 80
}
```

acme-client - /etc/acme-client.conf

```
authority letsencrypt {
    api url \
        "https://acme-v02.api.letsencrypt.org/directory"
    account key "/etc/acme/letsencrypt-privkey.pem"
}

domain example.com {
    domain key "/etc/ssl/private/example.com.key"
    domain full chain certificate \
        "/etc/ssl/example.com.fullchain.pem"
    sign with letsencrypt
}
```

acme-client - /etc/daily.local

```
acme-client example.com && rcctl reload httpd
```

acme-client - /etc/httpd.conf (1/2)

```
server "example.com" {
    listen on * port 80
    location "/.well-known/acme-challenge/*" {
        root "/acme"
        request strip 2
    }
    location * {
        block return 302 "https://$HTTP_HOST$REQUEST_URI"
    }
}
```

acme-client - /etc/httpd.conf (2/2)

```
server "example.com" {
    listen on * tls port 443
    tls {
        certificate "/etc/ssl/example.com.fullchain.pem"
        key "/etc/ssl/private/example.com.key"
    }
    location "/.well-known/acme-challenge/*" {
        root "/acme"
        request strip 2
    }
}
```

Andere Features

- doas
- unwind / resolvd
- spamd
- OpenBGPD / OpenOSPFD
- ...

Downsides

- Weniger Performance-optimiert
- “Altes” filesystem
- Kein Wayland (yet)

Q & A

Noch Fragen?

Secret Bonus-Slide