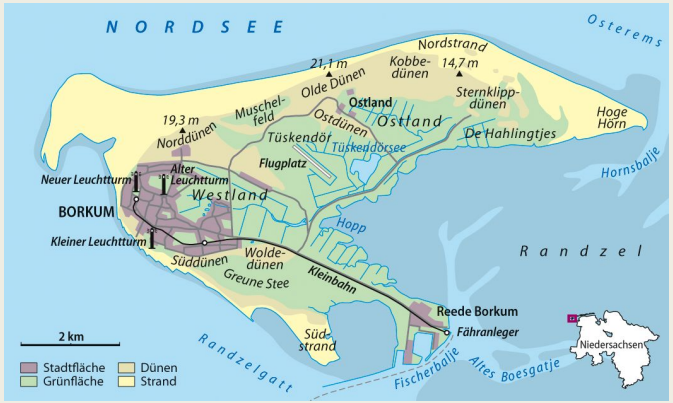
Linux&Netz Magie nicht nur fürs (Home)Setup?

obskures und vergessenes

/me

2022-now()	Cloud Gardener, HPC/AI Taiga Cloud
...	
2018-2022	Cloud Gardener, OpenStack, k8s, edge computing
...	
2012-2016	1&1, DNS Team, System Admin
pre-2012	Uni Paderborn, Freelancer
Studium	

Abgeschlossenes Studium
Mathe/Informatik für Lehramt an Gymnasien



Agenda

- Netzwerkboot w/o DHCP-Server Kontrolle
- Rescue Boot ohne Rescue CD

Motivation

- DC Like Tooling at Home
- Spass am System
- Rescue Boot oder Installer via Laptop/SBC/RPi
- Warum sind eigentlich alle Installer sch…?

Netzwerkboot w/o DHCP-Server Kontrolle

- get host config for this network
- network boot
  - *PXE*
  - ...

Client

Server

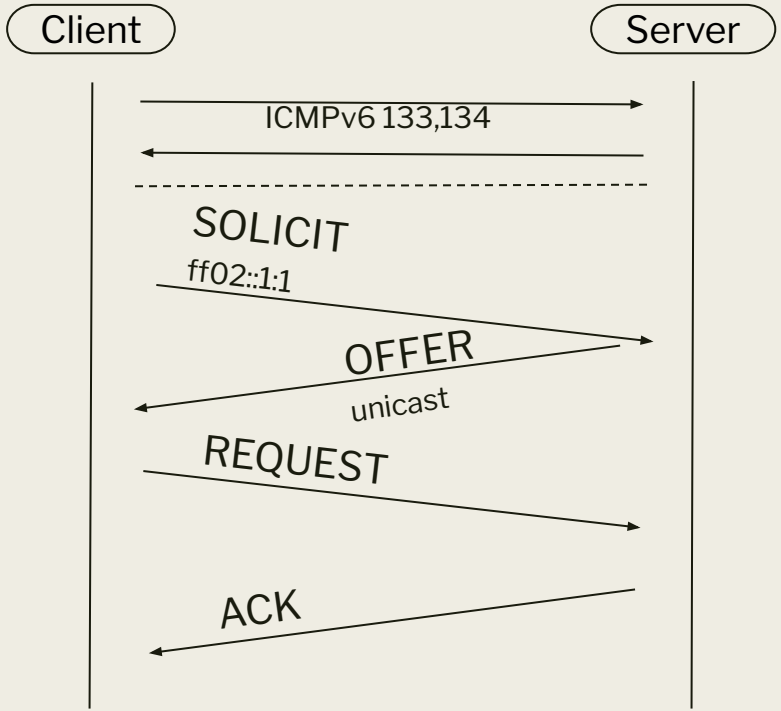
?

DHCP IPv6 Happy Path

RFC9951

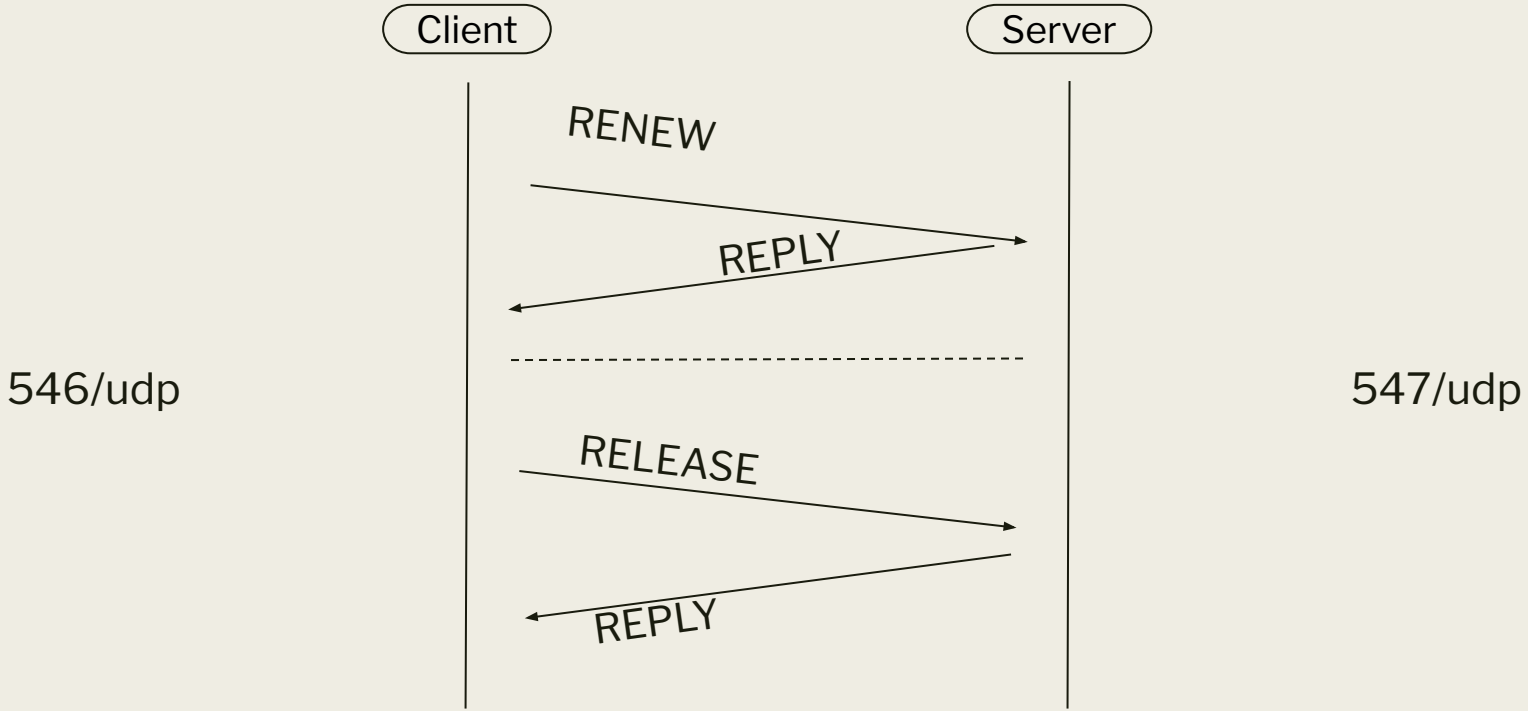
- NDP first get fe80::FLARP
- SLAAC, RA managed

546/udp



547/udp

DHCP IPv6 Happy Path



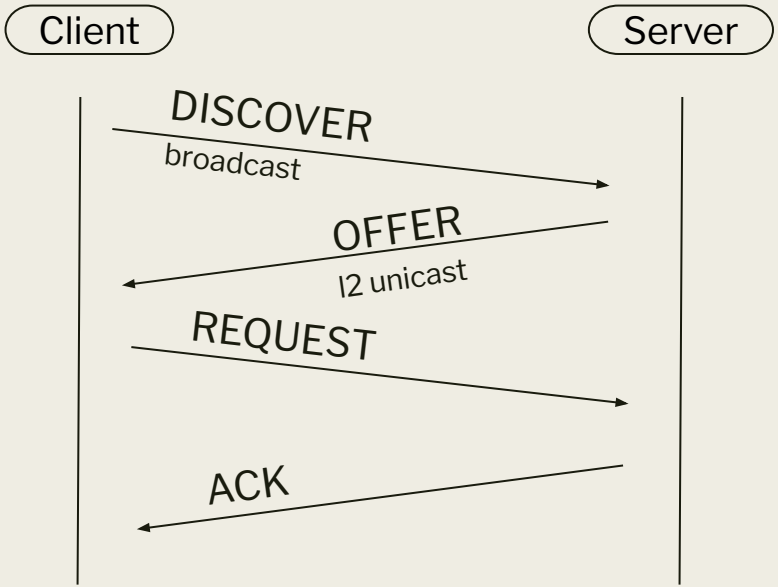
DHCP IPv4 Happy Path

RFC1541

RFC951 (bootp)

Transaction IDs

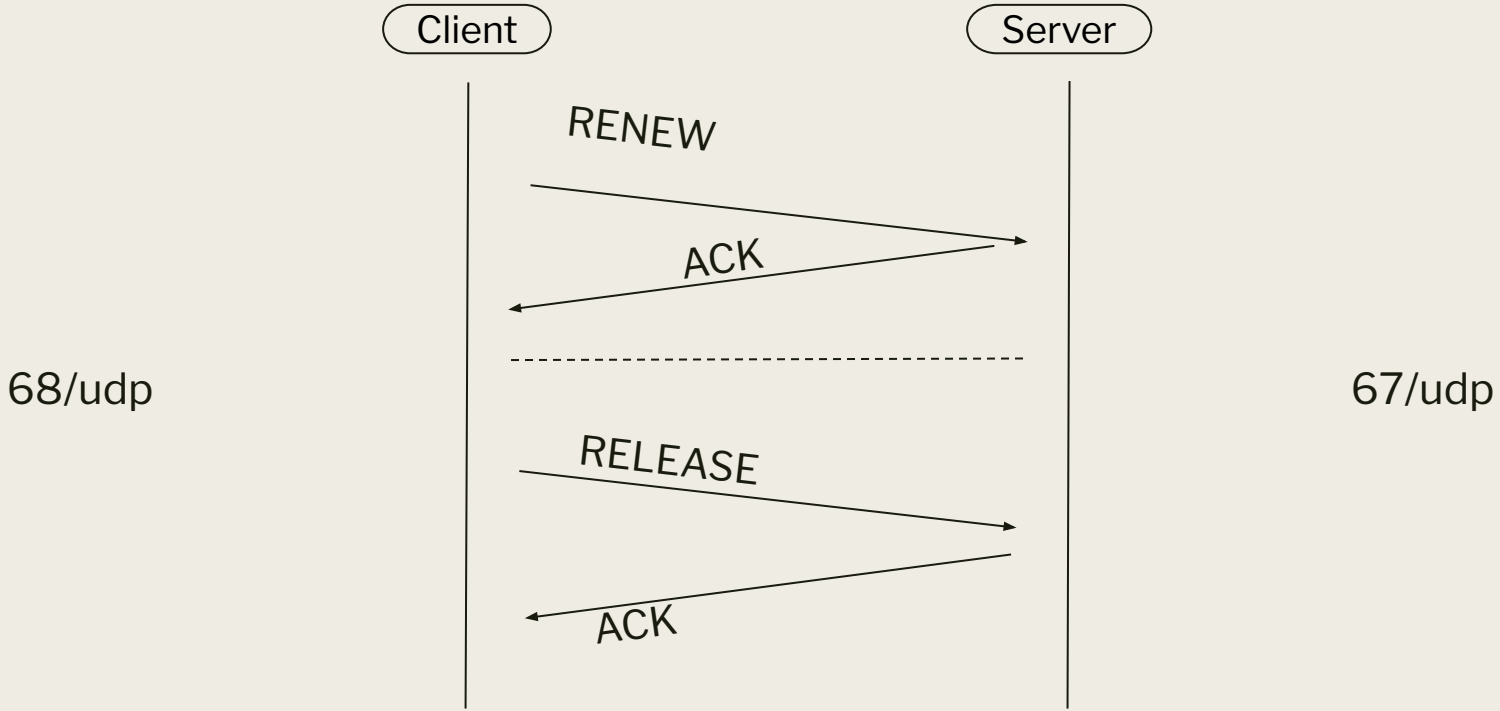
DUID



68/udp

67/udp

DHCP IPv4 Happy Path



SOLICIT/DISCOVER: I want to be configured for what?

MAC, DUID, VENDOR, ARCH

OPTIONS:

router, dns, search
next server, boot file name (boot-file-url DHCPv6)
mtu, ntp, sip, …

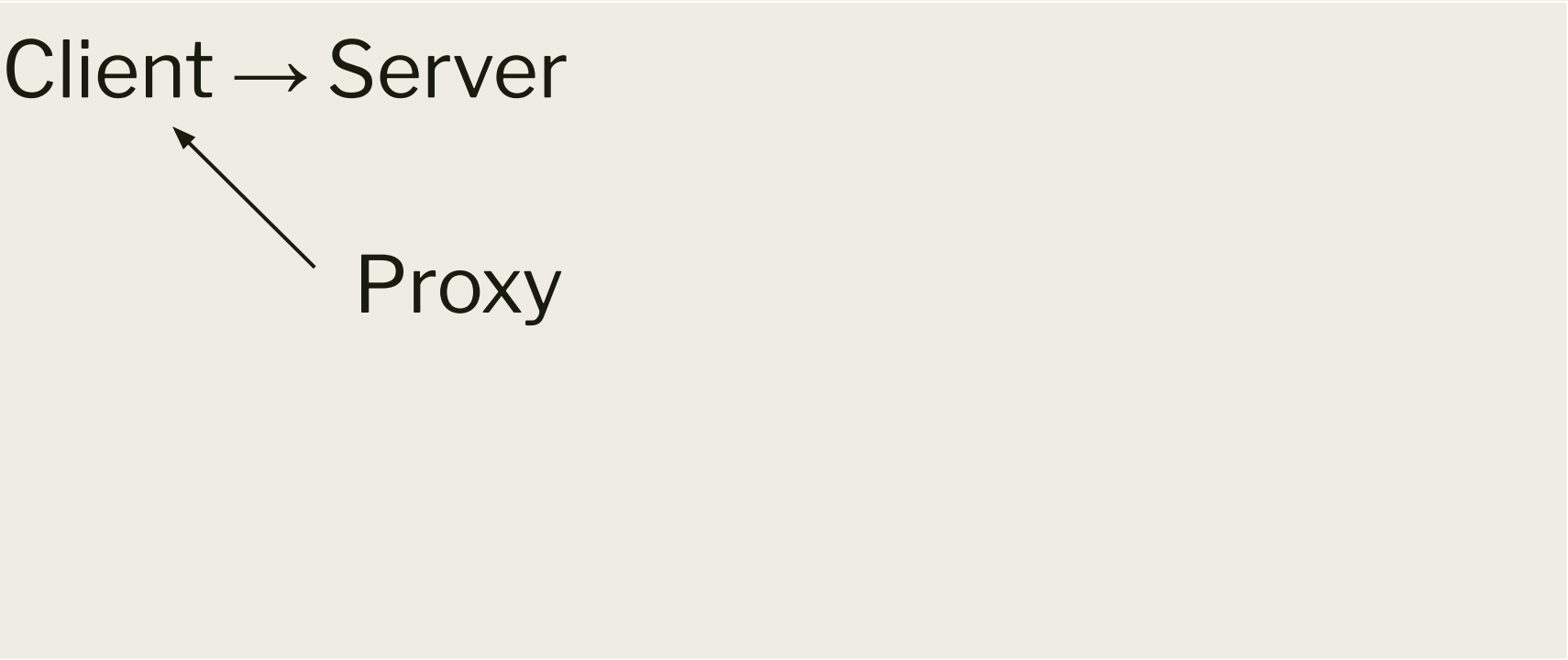
DHCP servers answers requested options if avail

DHCP - Liebe zu dritt?

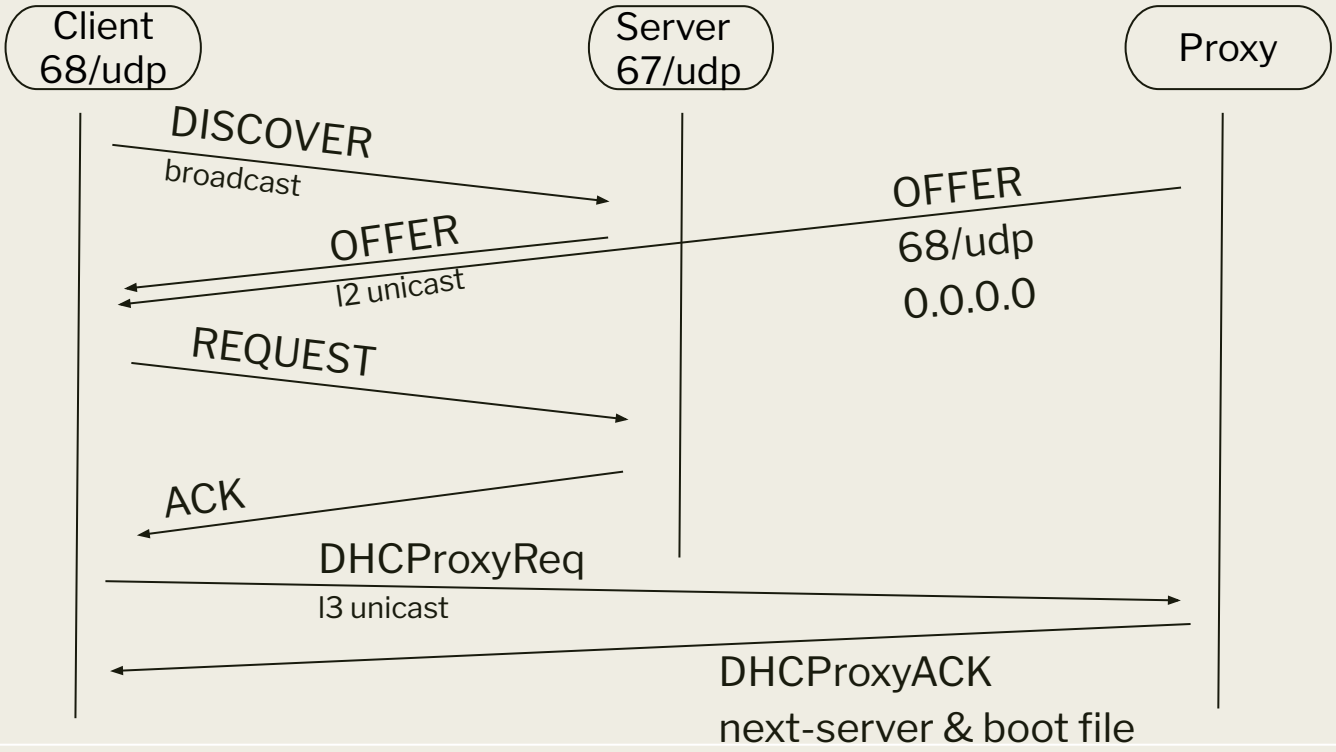
Client → Relay →Server

- Client to Relay L2 (multicast, broadcast)
  - *Relay forwards to Server L3 (unicast)*
  - *Server answers to Relay*
- Relay to Client

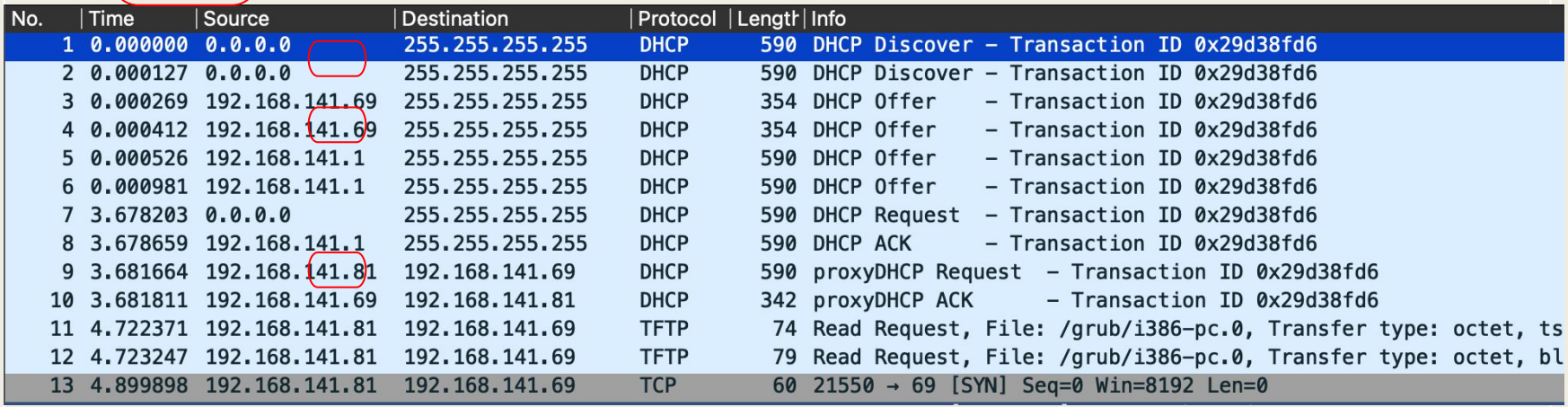
DHCP - Liebe zu dritt?



DHCP IPv4 Proxy



DHCP - Proxy



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	0.0.0.0	255.255.255.255	DHCP	590	DHCP Discover - Transaction ID 0x29d38fd6
2	0.000127	0.0.0.0	255.255.255.255	DHCP	590	DHCP Discover - Transaction ID 0x29d38fd6
3	0.000269	192.168.141.69	255.255.255.255	DHCP	354	DHCP Offer - Transaction ID 0x29d38fd6
4	0.000412	192.168.141.69	255.255.255.255	DHCP	354	DHCP Offer - Transaction ID 0x29d38fd6
5	0.000526	192.168.141.1	255.255.255.255	DHCP	590	DHCP Offer - Transaction ID 0x29d38fd6
6	0.000981	192.168.141.1	255.255.255.255	DHCP	590	DHCP Offer - Transaction ID 0x29d38fd6
7	3.678203	0.0.0.0	255.255.255.255	DHCP	590	DHCP Request - Transaction ID 0x29d38fd6
8	3.678659	192.168.141.1	255.255.255.255	DHCP	590	DHCP ACK - Transaction ID 0x29d38fd6
9	3.681664	192.168.141.81	192.168.141.69	DHCP	590	proxyDHCP Request - Transaction ID 0x29d38fd6
10	3.681811	192.168.141.69	192.168.141.81	DHCP	342	proxyDHCP ACK - Transaction ID 0x29d38fd6
11	4.722371	192.168.141.81	192.168.141.69	TFTP	74	Read Request, File: /grub/i386-pc.0, Transfer type: octet, ts
12	4.723247	192.168.141.81	192.168.141.69	TFTP	79	Read Request, File: /grub/i386-pc.0, Transfer type: octet, bl
13	4.899898	192.168.141.81	192.168.141.69	TCP	60	21550 → 69 [SYN] Seq=0 Win=8192 Len=0

DHCP - proxy ?

```
dnsmasq-dhcp: PXE(lan) 08:00:27:d3:8f:d6 proxy
dnsmasq-dhcp: PXE(lan) 08:00:27:d3:8f:d6 proxy
dnsmasq-dhcp: PXE(lan) 192.168.141.81 08:00:27:d3:8f:d6 /grub/i386-pc.0
dnsmasq-tftp: error 0 TFTP Aborted received from 192.168.141.81
dnsmasq-tftp: failed sending /tftp/grub/i386-pc.0 to 192.168.141.81
dnsmasq-tftp: sent /tftp/grub/i386-pc.0 to 192.168.141.81
nginx: GET /grub/i386-pc/command.lst HTTP/1.1 from 192.168.141.81 - 200 - - GRUB 2.02+dfsg1-20+deb10u4
```

Arch:

- i386-pc
- x86_64
- Raspberry Pis
- Grub/IPXE/UBoot

Use Case:

- NetBoot to go
- NetBoot

dnsmasq

ISC DHCPD (deprecated)

Rescue Boot ohne Rescue CD

?

Panic!

```
# fsck /dev/sda2
fsck from util-linux 2.41
e2fsck 1.47.2 (1-Jan-2025)
/dev/sda2 is mounted.
e2fsck: Cannot continue, aborting.


# umount /dev/sda2
umount: / target is busy.
```

Einsame Insel, no rescue CD at hand, keine Floppy oder Stick?

ideas?

Blockers?

Work-arounds?

ideas?

Blockers?

- Filesystem usage
- Open files
- Work directories

Work-arounds?

- close all files! easy ;)

PID1 - init

- running !
  - *not running → kernel panic*
- signal handling - at all but
  - *SIGCHLD*
  - *INT ← Ctrl-Alt-Delete*
  - *PWR*
  - *(SIGWINCH)*

PID1

- init U
- openrc-shutdown -R | --reexec
- systemctl daemon-reexec

PID1 - bind mounts?

mount /bin/bash /sbin/init

PID1 - combine it

- bind mount executable for take over

...

- re-exec

PID1 - combine it

- bind mount executable for take over
- re-exec

This is no security hole!

Works in containers , too

Caps Policies LibSeccomp eBPF

PID1 - use case

- why are all installers sh...?
- rescue system
- getting own rootfs on other devices with vendor kernel

- if no re-exec gdb

Demo: Substitute PID1

```
# cat > /tmp/init.substitute.sh < EOF

#!/bin/sh
set -eu
exec > /dev/tty3 < /dev/tty3 1>&2
exec /bin/bash -

EOF

# chmod 755 /tmp/init.substitute.sh

# ls -l /proc/1/exe

lrwxrwxrwx 1 root root 0 Mär  2 00:40 /proc/1/exe -> /usr/lib/systemd/systemd

# mount --bind /tmp/init.substitute.sh /usr/lib/systemd/systemd

# systemctl daemon-reexec
```

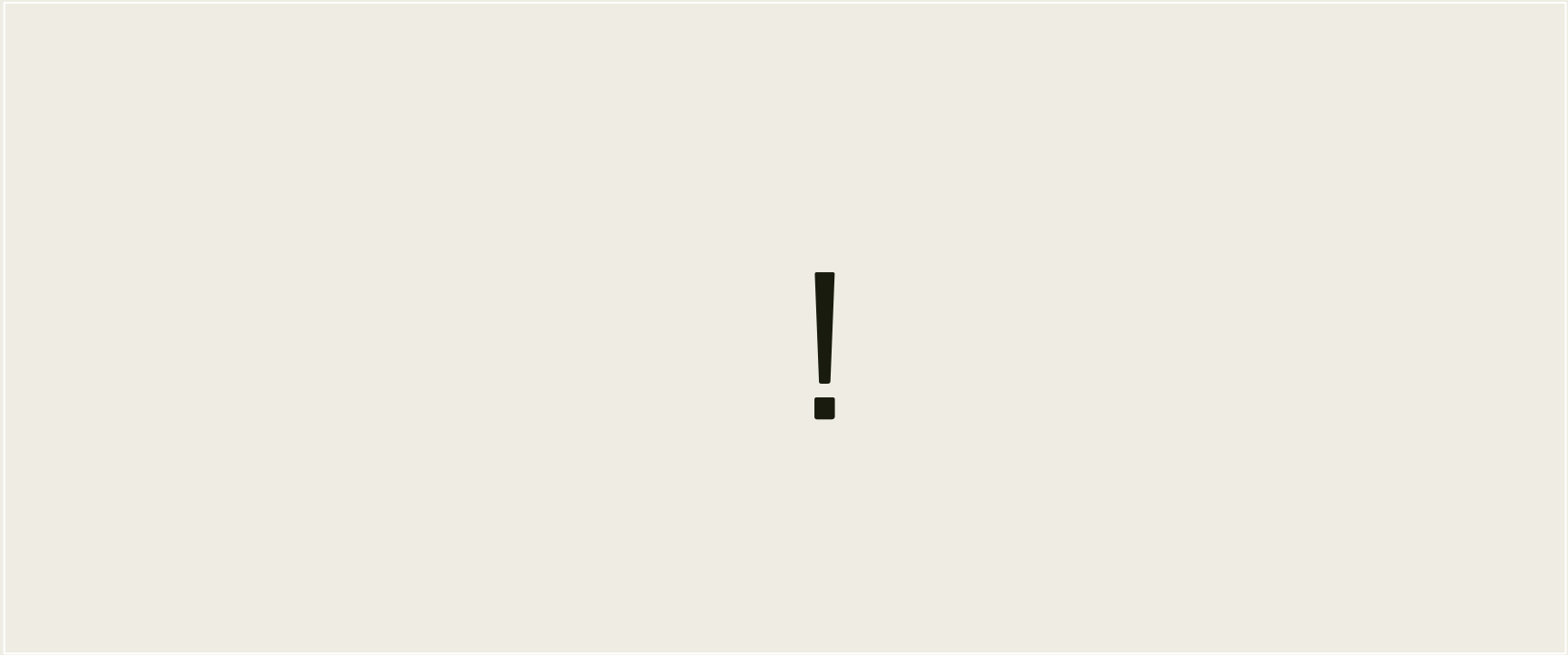
VT3: you find a open bash running as PID

Demo: Rescue Boot - on embedded SBC Linux

Source:

- https://github.com/j0ju/sbc-fw-alchemy/blob/main/recipes/alpine/SBC%2Bopenrc-init%2Boverlay/190_overlay%2Bopenrc-init.sh.d/usr/local/bin/BootToRam

Danke



Q&A