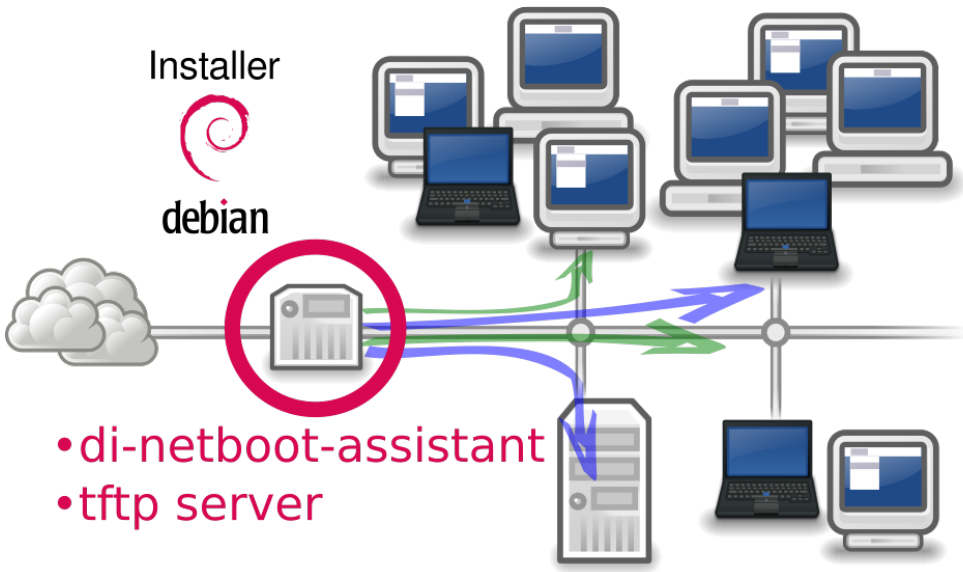
Debian ausrollen und konfigurieren

PXE, di-netboot-assistant, preseeding and ansible

Andreas B. Mundt
andi@debian.org

GNU/Linux Day

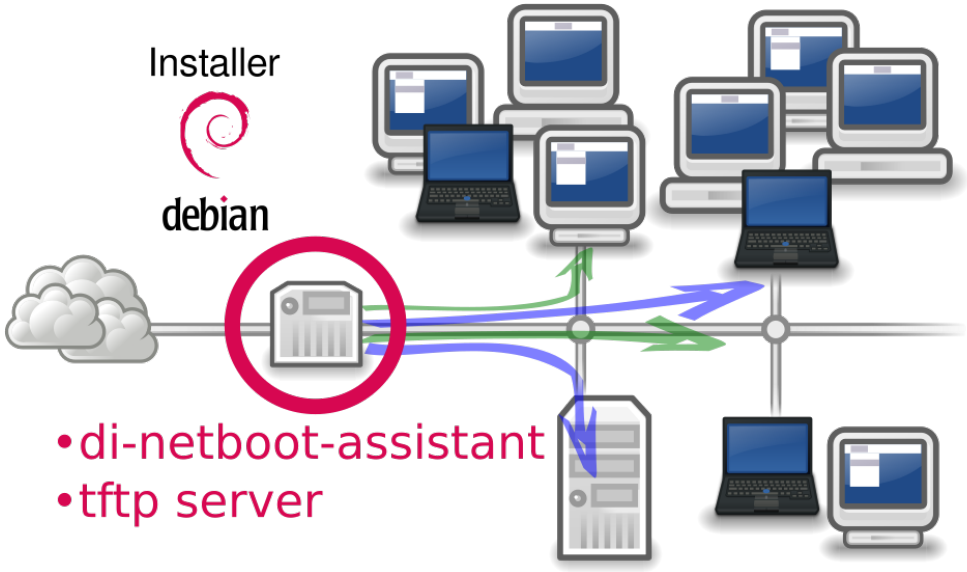
19.10.2019



Overview

1. TFTP/PXE Boot Install
2. `di-netboot-assistant`
3. Customization
   - Preseeding the Installation
   - Ansible Configuration Management
   - Combining debian-installer/preseeding and `ansible`
   - Playbook Examples
4. Summary and Questions
5. Workshop

TFTP/PXE Boot Install



- Configure client to boot over the network with PXE
- Configure DHCP server to send TFTP server address
- Client fetches pxelinux.0 (legacy BIOS) or bootnetx64.efi (UEFI)
- Client fetches kernel and initramfs from TFTP server . . .

$\Longrightarrow$ Install!

PXE boot: legacy BIOS

Client:

```
Booting from ROM...
iPXE (PCI 00:03.0) starting execution...ok
iPXE initialising devices...ok



iPXE 1.0.0+git-20190125.36a4c85-1 -- Open Source Network Boot Firmware -- http:/
/ipxe.org
Features: DNS HTTP iSCSI NFS TFTP AoE ELF MBOOT PXE bzImage Menu PXEXT

net0: 52:54:00:9e:e7:25 using virtio-net on 0000:00:03.0 (open)
  [Link:up, TX:0 TXE:0 RX:0 RXE:0]
Configuring (net0 52:54:00:9e:e7:25)................... ok
net0: 192.168.0.129/255.255.255.0 gw 192.168.0.10
net0: fe80::5054:ff:fe9e:e725/64
Next server: 192.168.0.10
Filename: d-i/n-a/pxelinux.0
tftp://192.168.0.10/d-i/n-a/pxelinux.0... ok
pxelinux.0 : 42425 bytes [PXE-NBP]

PXELINUX 6.04 PXE 20181203 Copyright (C) 1994-2015 H. Peter Anvin et al
```

PXE boot: legacy BIOS

DHCP/TFTP Server:

```
$ journald -f
dnsmasq-dhcp: DHCPDISCOVER(enp2s0) 52:54:00:9e:e7:25
dnsmasq-dhcp: DHCPOFFER(enp2s0) 192.168.0.129 52:54:00:9e:e7:25
dnsmasq-dhcp: DHCPDISCOVER(enp2s0) 52:54:00:9e:e7:25
dnsmasq-dhcp: DHCPOFFER(enp2s0) 192.168.0.129 52:54:00:9e:e7:25
dnsmasq-dhcp: DHCPREQUEST(enp2s0) 192.168.0.129 52:54:00:9e:e7:25
dnsmasq-dhcp: DHCPACK(enp2s0) 192.168.0.129 52:54:00:9e:e7:25 debian079
dnsmasq-tftp: sent /var/lib/tftpboot/d-i/n-a/pxelinux.0 to 192.168.0.129
dnsmasq-tftp: sent /var/lib/tftpboot/d-i/n-a/ldlinux.c32 to 192.168.0.129
dnsmasq-tftp: file /var/lib/tftpboot/d-i/n-a/pxelinux.cfg/C0A80081 not found
[...]
dnsmasq-tftp: file /var/lib/tftpboot/d-i/n-a/pxelinux.cfg/C0 not found
dnsmasq-tftp: file /var/lib/tftpboot/d-i/n-a/pxelinux.cfg/C not found
dnsmasq-tftp: sent /var/lib/tftpboot/d-i/n-a/pxelinux.cfg/default to 192.168.0.129
```

PXE boot: UEFI

Client:

```
iPXE 1.0.0+git-20190125.36a4c85-1 -- Open Source Network Boot Firmware -- http:/
/ipxe.org
Features: DNS HTTP iSCSI NFS TFTP AoE EFI Menu

net0: 52:54:00:e9:a4:74 using virtio-net on 0000:01:00.0 (open)
  [Link:up, TX:0 TXE:0 RX:6 RXE:1]
  [RXE: 1 x "Operation not supported (http://ipxe.org/3c3f6383)"]
Configuring (net0 52:54:00:e9:a4:74)................... ok
net0: 192.168.0.84/255.255.255.0 gw 192.168.0.10
net0: fe80::5054:ff:fee9:a474/64
Next server: 192.168.0.10
Filename: d-i/n-a/bootnetx64.efi
tftp://192.168.0.10/d-i/n-a/bootnetx64.efi... ok
bootnetx64.efi : 229376 bytes [EFI]
```

PXE boot: UEFI

DHCP/TFTP Server:

```
$ journald -f
dnsmasq-dhcp: DHCPDISCOVER(enp2s0) 52:54:00:e9:a4:74
dnsmasq-dhcp: DHCPOFFER(enp2s0) 192.168.0.84 52:54:00:e9:a4:74
dnsmasq-dhcp: DHCPDISCOVER(enp2s0) 52:54:00:e9:a4:74
dnsmasq-dhcp: DHCPOFFER(enp2s0) 192.168.0.84 52:54:00:e9:a4:74
dnsmasq-dhcp: DHCPREQUEST(enp2s0) 192.168.0.84 52:54:00:e9:a4:74
dnsmasq-dhcp: DHCPACK(enp2s0) 192.168.0.84 52:54:00:e9:a4:74 debian034
dnsmasq-tftp: sent /var/lib/tftpboot/d-i/n-a/bootnetx64.efi to 192.168.0.84
dnsmasq-tftp: sent /var/lib/tftpboot/d-i/n-a/grub/x86_64-efi/normal.mod to 192.168.
dnsmasq-tftp: sent /var/lib/tftpboot/d-i/n-a/grub/x86_64-efi/extcmd.mod to 192.168.
[...]
dnsmasq-tftp: sent /var/lib/tftpboot/d-i/n-a/grub/grub.cfg to 192.168.0.84
[...]
```

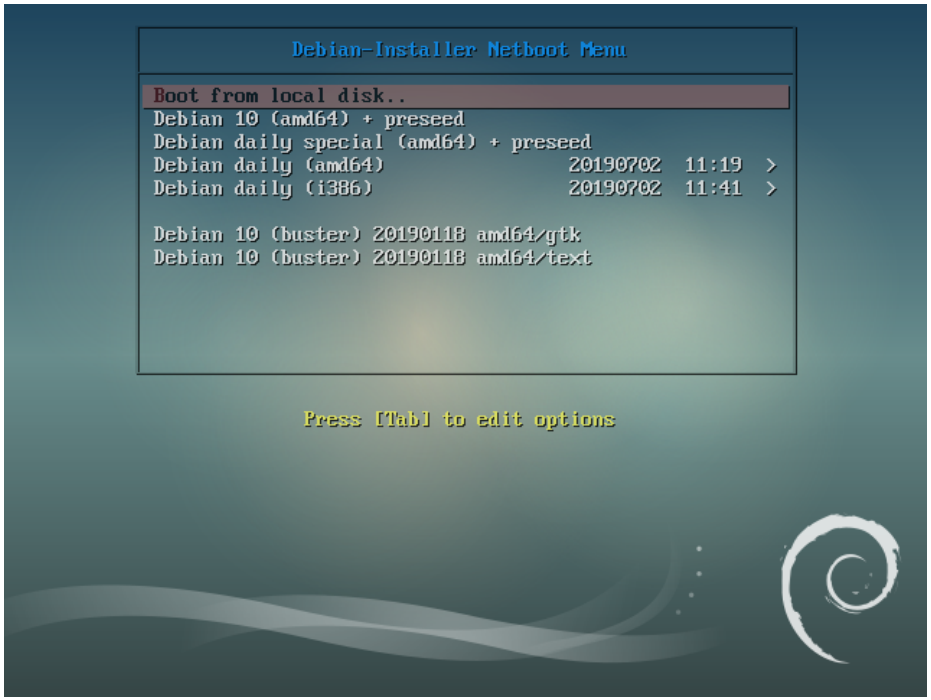
PXE: Legacy BIOS – UEFI

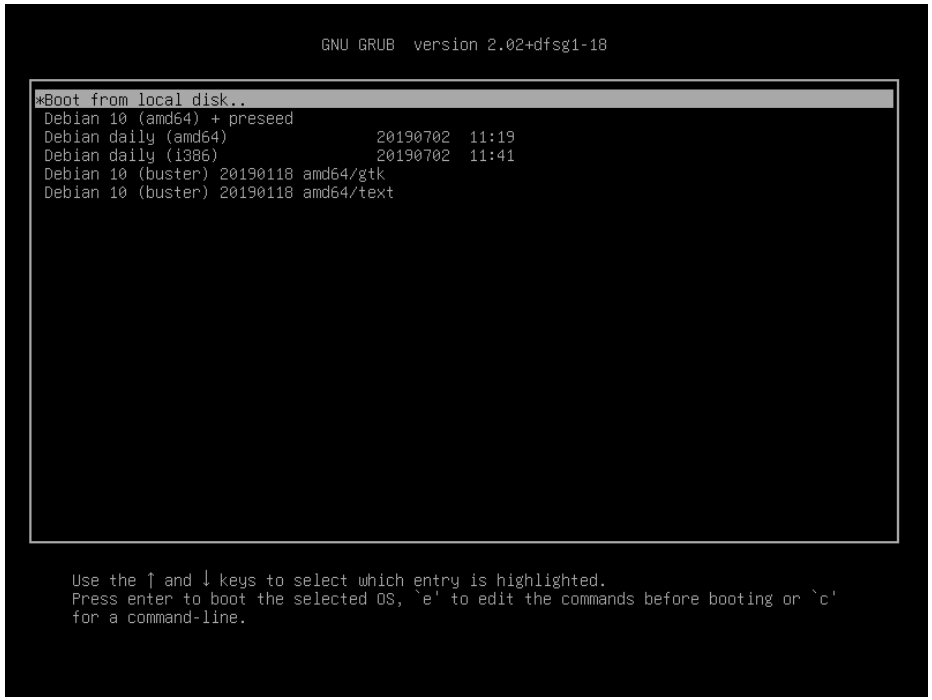
How does the client know which file to fetch?

The DHCP server sends the file path dependent on the client's identification:

- Legacy BIOS
  - PXE Linux (pxelinux.0)
  - isc-dhcp: `option arch = 00:00`
  - dnsmasq: `dhcp-match=set:bios,option:client-arch,0`
- UEFI
  - GRUB (grubx64.efi, bootnetx64.efi)
  - isc-dhcpd: `option arch = 00:07`
  - dnsmasq: `dhcp-match=set:efi-x86_64,option:client-arch,7`

PXE boot menu (examples)





Overview

1. TFTP/PXE Boot Install
2. `di-netboot-assistant`
3. Customization
4. Summary and Questions
5. Workshop

di-netboot-assistant

Debian-Installer netboot assistant package:

- Simplify the preparation of files for TFTP net booting.
- Download and extract files needed and organize them.
- Support PXElinux and grub-efi.
- Prepare a top level menu.
- Add non-free firmware to the installer images.
- Support for preseeding the installer.
- Support for ansible/ansible-pull.

di-netboot-assistant

"News" from the di-netboot-assistant changelog:

- Implement the inclusion of debian-installer packages.
- Implement signature verification for images.
- Implement EFI boot with grub EFI image.
- Add download date and time to PXE-menu entries.
- Implement autopkgtest.
- Updates, improvements and new/more examples.
- Add u-boot pxe menu for arm systems.
- Add example 'installbox'.

. . . what images are available? . . .

Installation:

```
apt install di-netboot-assistant \
     # -t stretch-backports \
     # --no-install-recommends
```

Let's run di-netboot-assistant!

```
$ di-netboot-assistant install
E: No repository name was passed for 'install'.
I: Declared repositories are:
bullseye bullseye-gtk buster buster-gtk daily daily-gtk
jessie jessie-gtk oldstable oldstable-gtk precise stable
stable-gtk stretch stretch-gtk testing testing-gtk trusty
vivid wheezy wheezy-gtk wily xenial yakkety zesty
```

. . . install some images . . .

```
# di-netboot-assistant install stable testing
I: Processing stable/amd64.
I: Downloading 'SHA256SUMS'.
I: Good signature from "Debian Archive Automatic Signing Key
[...]
I: Building menu entries for the netboot-images.
I:  • stable-amd64
I: Using splash screen from 'stable' image.
[...]
I: Building menu entries for the netboot-images.
I:  • stable-amd64
I:  • testing-amd64
I: Building menu entries for debian-installer-*-netboot-* packages.
[...]
```

. . . add firmware . . .

di-netboot-assistant fw-toggle stable

```
I: Processing non-free firmware for stable/amd64.
I: Downloading 'SHA256SUMS'.
[...]
I: Downloading 'firmware.cpio.gz'.
I: Adding non-free firmware.
```

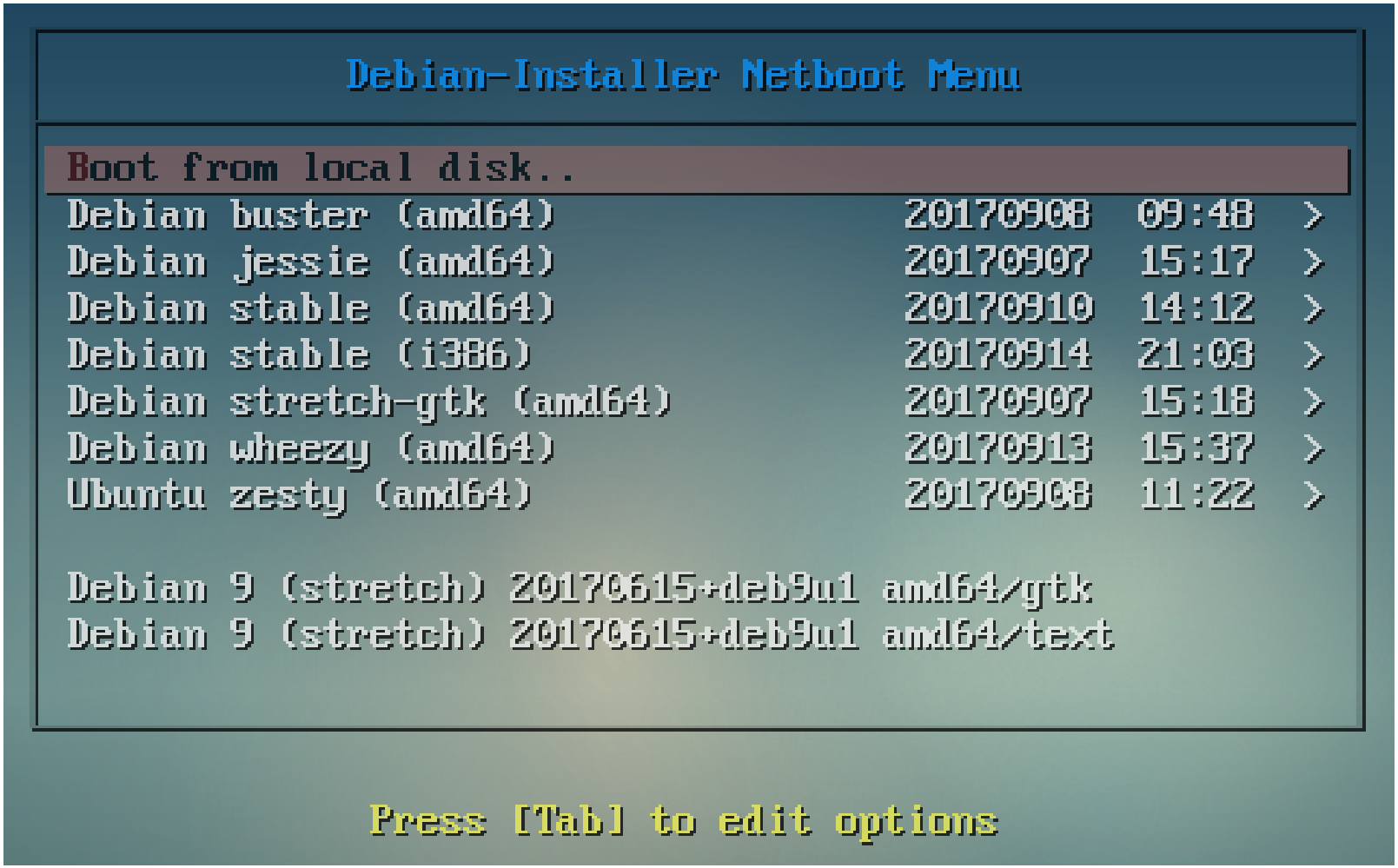
ls -l /var/lib/tftp/d-i/n-a/stable/amd64/

```
[...]
-rw-r--r-- 1 root root 114469518 Jul  5 08:24 initrd.gz
-rw-r--r-- 1 root root 114469518 Jul  5 08:24 initrd.gz.fw
-rw-r--r-- 1 root root  30620123 Jul  1 20:09 initrd.gz.orig
[...]
```

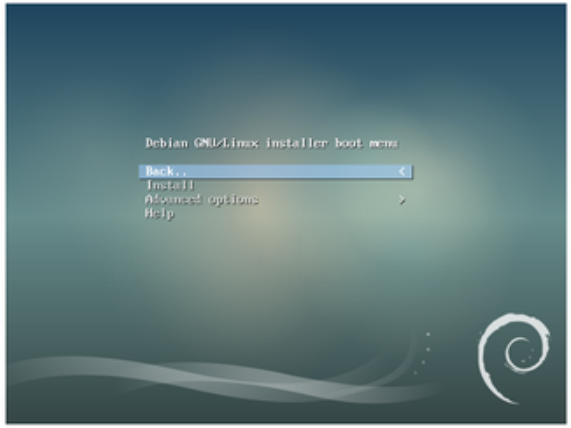
di-netboot-assistant fw-toggle stable

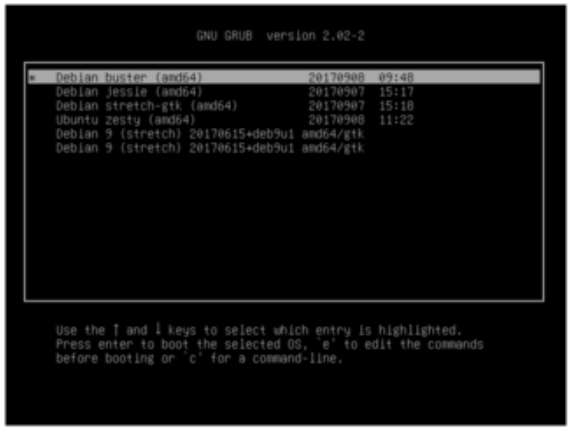
```
I: Processing non-free firmware for stable/amd64.
I: Removing non-free firmware.
```

. . . boot the client . . .



. . . enjoy the sub menus:





Example 'installbox' (README.installbox)

Preparations for the installbox:

```
## Install packages, stop dnsmasq and prevent it from starting at boot:
# apt install di-netboot-assistant dnsmasq nftables \
#                              debian-installer-netboot-amd64
# systemctl stop dnsmasq
# systemctl disable dnsmasq
# cp /usr/share/doc/di-netboot-assistant/examples/dnsmasq.conf.simple \
#                                          /etc/dnsmasq.d/netboot-installer

## Prepare location for debian-installer-netboot-amd64 images:
# mkdir -p /var/lib/tftpboot/d-i/n-pkg/
```

Example 'installbox' (README.installbox)

Temporary convert to an installbox:

```
LAN_IF="${LAN_IF:-enp2s0}"

## Make packaged installer files available:
mount -o bind,ro /usr/lib/debian-installer/ /var/lib/tftpboot/d-i/n-pkg/
di-netboot-assistant install daily

## Configure the netboot interface:
ip address add 192.168.0.10/24 dev $LAN_IF
ip link set $LAN_IF up

## Enable forwarding:
sysctl -w net.ipv4.ip_forward=1

## Configure masquerading:
nft add table nat
nft add chain nat prerouting { type nat hook prerouting priority 0 \; }
nft add chain nat postrouting { type nat hook postrouting priority 100 \; }
nft add rule nat postrouting masquerade

## Provide DNS, DHCP and a TFTP server for the netboot clients:
systemctl start dnsmasq
```

Customization

1. TFTP/PXE Boot Install
2. `di-netboot-assistant`
3. Customization
   - Preseeding the Installation
   - Ansible Configuration Management
   - Combining debian-installer/preseeding and `ansible`
   - Playbook Examples
4. Summary and Questions
5. Workshop

Preseeding the Installation

preseeding = answer installer questions in advance

How? $\Longrightarrow$ provide preconfiguration/preseed-file

Methods:

- initrd
- file
- network
  - http
  - tftp
- boot parameters

Preseeding with di-netboot-assistant

Customized boot options (pxelinux.HEAD/grub.cfg.HEAD):

/etc/di-netboot-assistant/pxelinux.HEAD

```
DEFAULT bootlocal
TIMEOUT 10
# ##################################################################
# Install stable automatically.
# For details consult
#                 '/usr/share/doc/di-netboot-assistant/README.preseed'.
#
LABEL autoinstall
   MENU LABEL Debian stable (amd64) + preseed
   kernel ::/d-i/n-a/stable/amd64/linux
   append initrd=::/d-i/n-a/stable/amd64/initrd.gz \
            auto=true priority=critical url=tftp://installbox \
                                        playbook=ANSIBLEPLAYBOOK ---
#
# ##################################################################
```

Preparing the Preseed File

- Example preseed file:
  `/usr/share/doc/di-netboot-assistant/examples/preseed.cfg`
- Use TFTP server:

Prepare `$TFTP_ROOT/d-i/buster/`:

```
# mkdir -p /var/lib/tftpboot/d-i/buster/
# cp /usr/share/doc/di-netboot-assistant/examples/preseed.cfg \
                                        /var/lib/tftpboot/d-i/buster/
```

- Modify the preseed file to your needs:
  - localization
  - partitioning
  - ssh public key
- Make sure the TFTP server is found (DNS or use IP address)

Preseeding a complete system is rather complicated and hard to maintain.

Ansible Configuration Management

1. TFTP/PXE Boot Install
2. `di-netboot-assistant`
3. Customization
   - Preseeding the Installation
   - Ansible Configuration Management
   - Combining debian-installer/preseeding and `ansible`
   - Playbook Examples
4. Summary and Questions
5. Workshop

Configuration Management: Ansible Basics

hosts, group of hosts;
roles: tasks, files, templates, handlers;
playbooks

`ansible-playbook`

- private/public key ssh login into client as root or sudo user
- run playbooks (configuration space) on the client (push config)

`ansible-pull`

- client fetches playbooks (configuration space) from the network
- implement the relevant playbooks on the client (pull config)

Combining debian-installer/preseeding and `ansible`

Idea:

- Only preseed the minimal necessary options.
- Further set up the system from ansible playbooks.

Advantages:

- Preseeding is done once for all.
- Configuration management system's advantages (roles, logging, ...).

An even better idea: Use ansible-pull during the installation.

- Same source of configuration during and after installation.
- Installation to the final system state is done in one step.

Combining debian-installer/preseeding and `ansible`

Idea:

- Only preseed the minimal necessary options.
- Further set up the system from ansible playbooks.

Advantages:

- Preseeding is done once for all.
- Configuration management system's advantages (roles, logging, ...).

An even better idea: Use ansible-pull during the installation.

- Same source of configuration during and after installation.
- Installation to the final system state is done in one step.

Implementation with ansible-pull

Preseed file:

```
### Individual additional packages to install
d-i pkgsel/include string firmware-linux ansible git
[...]
### This command is run just before the install finishes:
d-i preseed/late_command string \
      mkdir -p /target/home/ansible/.ssh && \
      echo "ssh-rsa YOURSSHPUBLICKEYHERE ansible@installbox" \
                >> /target/home/ansible/.ssh/authorized_keys ; \
      in-target chown -R ansible:ansible /home/ansible/.ssh/ ; \
      in-target chmod -R og= /home/ansible/.ssh/ ; \
      in-target ansible-pull --verbose --purge \
      -i localhost, --url=git://installbox/.git $playbook
```

Export ansible playbook on installbox:

```
git daemon --verbose --export-all
       --base-path=/dir/of/playbook -- /dir/of/playbook
```

di-netboot-assistant playbook menu

Boot options (pxelinux.HEAD/grub.cfg.HEAD):

/etc/di-netboot-assistant/pxelinux.HEAD

```
MENU LABEL Debian stable (amd64) + preseed + WEBSERVER
kernel ::/d-i/n-a/stable/amd64/linux
append initrd=::/d-i/n-a/stable/amd64/initrd.gz \
        auto=true priority=critical url=tftp://installbox \
                                    playbook=webserver ---

MENU LABEL Debian testing (amd64) + preseed + DESKTOP
kernel ::/d-i/n-a/stable/amd64/linux
append initrd=::/d-i/n-a/stable/amd64/initrd.gz \
        auto=true priority=critical url=tftp://installbox \
                                    playbook=desktop ---

...
```

Playbook Examples

A few ansible playbook examples are provided here:

`https://salsa.debian.org/andi/debian-lan-ansible`

```
git clone https://salsa.debian.org/andi/debian-lan-ansible.git
```

Contributions like patches, suggestions, pull requests and/or further profiles are highly appreciated!

Playbook Examples

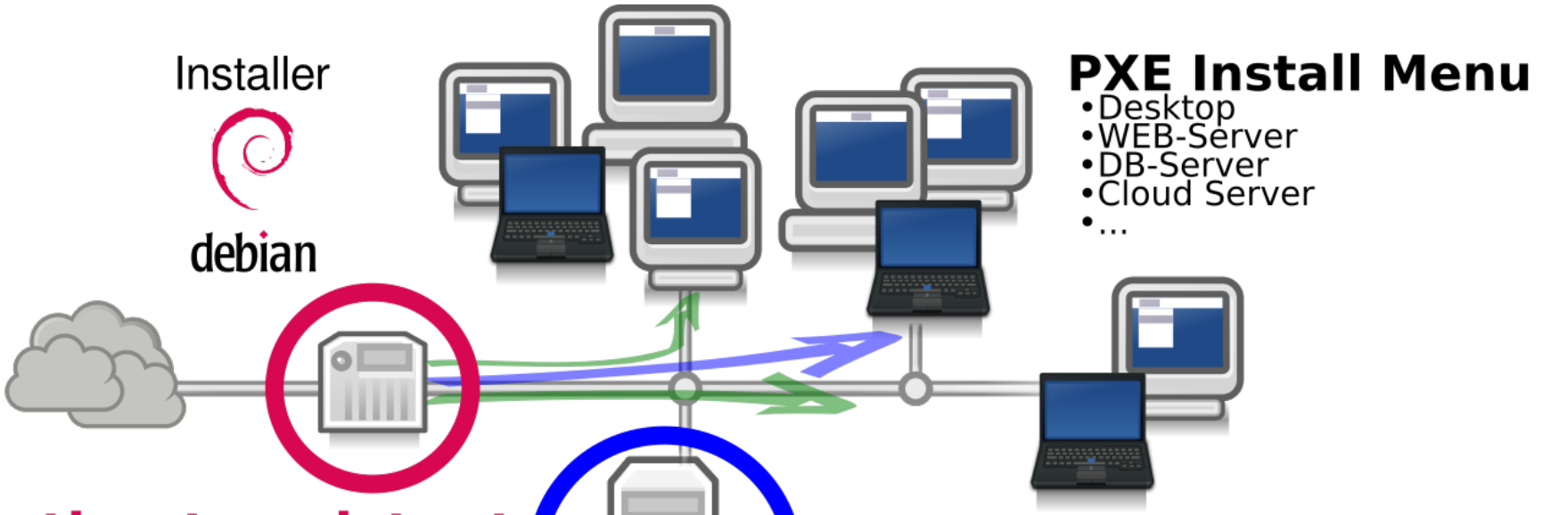
- Installbox:
  - set up as gateway to some external network (WAN)
  - providing TFTP installation on the LAN interface
  - automatic installs: preseeding and ansible playbooks
  - squid package cache

  use case: orchestration of automatic installs in the LAN
- Kiosk:
  - auto login user
  - reasonable defaults
  - temporary home directory on tmpfs, reset at boot

  use case: school's computer, hackerspace, ...
- Cloudbox:
  - setup of a home cloud server
  - dynamic DNS name
  - ...

  use case: nextcloud server in the basement

Summary



di-netboot-assistant

- TFTP server
- preseed.conf

Ansible in git repository

- playbooks
- roles

Summary and Questions

1. TFTP/PXE Boot Install
2. `di-netboot-assistant`
3. Customization
   - Preseeding the Installation
   - Ansible Configuration Management
   - Combining debian-installer/preseeding and `ansible`
   - Playbook Examples
4. Summary and Questions
5. Workshop

Summary and Questions

1. TFTP/PXE Boot Install
2. `di-netboot-assistant`
3. Customization
   - Preseeding the Installation
   - Ansible Configuration Management
   - Combining debian-installer/preseeding and `ansible`
   - Playbook Examples
4. Summary and Questions
5. Workshop

Thank you very much!

Further Reading and Resources

- `di-netboot-assistant` package:

  `https://packages.debian.org/di-netboot-assistant`

  `https://salsa.debian.org/installer-team/netboot-assistant`
- Debian Documentation “Preseeding”:

  `https://www.debian.org/releases/stretch/amd64/apb.html.en`
- Debian Wiki:

  `https://wiki.debian.org/DebianInstaller/NetbootAssistant`

  `https://wiki.debian.org/DebianLAN`
- Debian-LAN presentations:

  `https://people.debian.org/~andi/`
- Debian-LAN-Ansible:

  `https://salsa.debian.org/andi/debian-lan-ansible`

Illustrations remixed from: `https://openclipart.org/`

Workshop with qemu built-in TFTP server

```
## PREPARE
#sudo apt --no-install-recommends -t stretch-backports install di-netboot-assistant
sudo apt --no-install-recommends install di-netboot-assistant
sudo di-netboot-assistant install buster
TFTP_ROOT="/var/lib/tftpboot/"
qemu-system-x86_64 -enable-kvm -m 1024 -boot n -net nic \
            -net user,tftp=$TFTP_ROOT,bootfile=/d-i/n-a/pxelinux.0
## PRESEED
cd $TFTP_ROOT/d-i
sudo mkdir buster
sudo cp /usr/share/doc/di-netboot-assistant/examples/preseed.cfg buster/
sudo -e buster/preseed.cfg
sudo -e /etc/di-netboot-assistant/pxelinux.HEAD
sed -i "s/installbox/10.0.2.2/g" /etc/di-netboot-assistant/pxelinux.HEAD
sudo di-netboot-assistant rebuild-menu
qemu-system-x86_64 -enable-kvm -m 1024 -boot n -net nic \
            -net user,tftp=$TFTP_ROOT,bootfile=/d-i/n-a/pxelinux.0
## ANSIBLE
# a-pull preseeding: --url=https://salsa.debian.org/andi/debian-lan-ansible.git
qemu-img create -f qcow2 /tmp/vm.qcow2 8G
qemu-system-x86_64 -enable-kvm -m 1024 -boot n -net nic \
   -net user,tftp=$TFTP_ROOT,bootfile=/d-i/n-a/pxelinux.0 -hda /tmp/vm.qcow2
## READY
qemu-system-x86_64 -enable-kvm -m 1024 -net nic -net user -hda /tmp/vm.qcow2
```