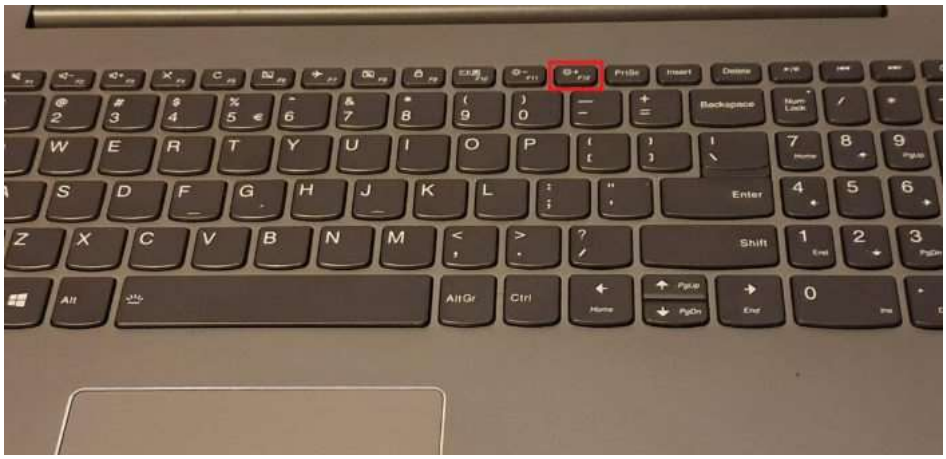

Using PXE to deploy a DNS server

0

BY FREDERIQUE.RETSEMA@AMIS.NL ON FEBRUARY 15, 2020 DEVOPS, PLATFORM TECHNOLOGY

Last week, I published a blog about implementing a PXE server [1]. Today I'll show how kickstart/anaconda files can be used to deploy a server. I will use the example of a DNS-server to explain what can be changed where in kickstart files and what you can do when the kickstart file isn't enough for your purposes. The full kickstart file and an example menu for the deployment can be found both at the end of this blog and on my github repository [2].

1. Kickstart files

The best place to change settings is in the kickstart file itself: many checks will be done before the installation starts. The VM that we deployed in the previous blog was configured based on DHCP, it had the following line in it:

```
network –bootproto=dhcp –device=eth0 –ipv6=auto –activate
```

For a DNS, we should use a static DNS address, so we will change this into:

```
network –bootproto=static –ip=192.168.2.3 –gateway=192.168.2.254 –nameserver=192.168.2.254 –
netmask=255.255.255.0 –ipv6=auto –device=eth0 –activate
```

The network directive is also the place to configure the name of the server:

```
network –hostname=dns
```

There is also room for specifying additional packages to the (in this case: minimal) base install. kexec-tools was already in the kickstart file, it is needed for the install itself. I added bind (for the DNS-server) and bind-utils (for utilities like dig, nslookup etc) myself:

```
%packages
@^minimal-environment
bind
```

ABOUT AUTHOR

frederique.retsema@amis.

Frederique Retsema is active in IT sir 1993. Senior Consultant and develop on diverse areas including SQL and Java. She likes work with automation tools like Bamboo, Jenkins, Ansible, Terraform and CloudFormation.

View all posts

```
bind-utils
kexec-tools
%end
```

2. The post-install part of kickstart

There is a part in the kickstart file that by default isn't there. I'm talking about the post install part, you can use this part to add or change files after the installation, but before the reboot. This part starts with %post and ends with %end.

For the DNS-server, there should be changes in the configuration file /etc/named.conf. After the installation of the DNS-server, the DNS-server will by default respond only to connections from its own server (localhost). To be useful for other servers in the network this should be changed to the network the DNS server is running on. Two lines must be changed: the line where we listen on port 53 must be changed from 127.0.0.1 to the network address of the DNS-server, in my case 192.168.2.3 ("listen-on port 53 { 192.168.2.3;};"), and the line where we allow queries from our network must be changed from localhost to 192.168.2.0/24 (every node in our network: "allow-query { 192.168.2.0/24;};").

The configuration file should also contain our own (local) domain mydomain.org, where we can add some DNS-entries. After the line

```
zone "." IN {
    type "hints";
};
```

the following text should be added:

```
zone "mydomain.org" IN {
    type master;
    file "mydomain.org";
   allow-update { none; } ;
};
```

The "allow-update { none;};" will take care that the domain cannot be changed by other nodes, the only way to change the content of this zone is to change the configuration file mydomain.org.

After that, the file mydomain.org will be created in the /var/named directory, with the following content:

```
$TTL 86400
$ORIGIN mydomain.org.

@ IN SOA dns.mydomain.org. info.mydomain.org (
    20200214
    3600
    1800
    604800
    86400)
@ IN NS dns.mydomain.org.
dns      IN A 192.168.2.3
server1 IN A 192.168.2.11
server2 IN A 192.168.2.12
```

Now, the only thing left to do is to change the firewall. When we would configure a DNS server by hand, we would simply use the following commands on the command line:

```
firewall-cmd –add-port=53/tcp
firewall-cmd –add-port=53/tcp –permanent
firewall-cmd –add-port=53/udp
firewall-cmd –add-port=53/udp –permanent
```

The problem is, however, that when we would use this in the kickstart file, the changes would be made on the installation environment, not on the destination host. So we have to come up with a trick to do these changes directly after the reboot of the server.

3. Start up script

We can do that by creating a start up script that will only run once, after the first start of the server. The script should run after the firewall is started. We will create a "oneshot service" in systemd, which removes itself after execution. Anaconda should create a file /etc/systemd/system/dnsconf.service with the following content:

```
[Unit]
Description=Configure selinux for named
After=firewalld.service

[Service]
```

```
Type=oneshot
ExecStart=/bin/firewall-cmd --add-port=53/tcp
ExecStart=/bin/firewall-cmd --add-port=53/tcp --permanent
ExecStart=/bin/firewall-cmd --add-port=53/udp
ExecStart=/bin/firewall-cmd --add-port=53/udp --permanent
ExecStart=rm -f /etc/systemd/system/dnsconf.service
ExecStart=rm -f /etc/systemd/system/multi-user.target.wants/dnsconf.service

[Install]
WantedBy=multi-user.target
```

To let this work, there should be a softlink in /etc/systemd/system/multi-user.target.wants. Normally, you wouldn't create this softlink yourself, it is placed there by the command systemctl enable dnsconf. But because systemd is not running on the target system when you create these files, Anaconda will have to put the softlink there for you. The creation of the file /etc/systemd/system/dnsconf.service and the creation of the softlink is done in the %post part of the Anaconda file. The same is true for the DNS service itself:

> ln -s /etc/systemd/system/dnsconf.service /etc/systemd/system/multi-user.target.wants/dnsconf.service
> ln -s /etc/systemd/system/named.service /etc/systemd/system/multi-user.target.wants/named.service

4. Conclusion

The nice thing about this solution is that you can deploy an OS and configure it over the network. Even without logging on to the server, this solution will work as soon as you see the login prompt: you can use dig @192.168.2.3 server1.mydomain.org from another server and you will see the address 192.168.2.11 as a result.

For a small organization with few changes in their DNS configuration, it might be considered not to allow users on this new machine at all (not even an administrator) for security reasons: in that case, a userdel command can be added in the %post installation part of Anaconda to delete the (in this example) frederique user. When there should be changes in the DNS environment, the changes are made on the kickstart file on the PXE server and the DNS server is redeployed with PXE after that.

Footnotes

[1] https://technology.amis.nl/2020/02/08/deploying-centos-8-using-pxe/

[2] https://github.com/FrederiqueRetsema/AMIS-Blog-PXE, in the dns directory

Content of the file /etc/lib/tftpboot/pxelinux.cfg/C0A802:

```
DEFAULT menu.c32
PROMPT 0
TIMEOUT 30
LABEL CentOS 8 dns
MENU centos8dns
KERNEL /networkboot/vmlinuz
APPEND initrd=/networkboot/initrd.ing inst.repo=ftp://192.168.2.131/CentOS8 ks=ftp://192.168.2.131/ce
```

Content of /var/ftp/centos8dns.cfg:

```
#version=RHEL8
ignoredisk --only-use=sda
autopart --type=lvm
# Partition clearing information
clearpart --all
zerombr
# Use graphical install
graphical
repo --name=centos-updates --mirrorlist=http://mirrorlist.centos.org/?release=$releasever&amp;arch=
repo --name=appstream-updates --mirrorlist=http://mirrorlist.centos.org/?release=$releasever&amp;ar
repo --name=extras-updates --mirrorlist=http://mirrorlist.centos.org/?release=$releasever&amp;arch=
# Use FTP installation media
url --url=ftp://192.168.2.154/CentOS8
# Keyboard layouts
keyboard --vckeymap=us --xlayouts='us'
# System language
lang en_US.UTF-8


# Network information
network  --bootproto=static --ip=192.168.2.3 --gateway=192.168.2.254 --nameserver=192.168.2.254 --r
network  --hostname=dns.mydomain.org
#Root password
rootpw --lock
# Run the Setup Agent on first boot
firstboot --enable
# Do not configure the X Window System
skipx
# System services
services --enabled="chronyd"
# System timezone
timezone Europe/Amsterdam --isUtc
user --groups=wheel --name=frederique --password=$6$ISxpNV3kE7gfvlpi$RRWVusy/EiatEuCEIdYhR.R1PdT6KE


%packages
@^minimal-environment
bind
bind-utils
kexec-tools


%end

```

```
%addon com_redhat_kdump --enable --reserve-mb='auto'


%end


%post
    sed -i 's/127.0.0.1/192.168.2.3/g' /etc/named.conf
    sed -i '/allow-query/c \ \ \ \ \ \ \ allow-query { 192.168.2.0/24;};' /etc/named.conf
    ZONELINENO=`grep -rne "zone \".\" IN {" /etc/named.conf | awk -F":" '{print $1}'`
    let ZONELINENO=ZONELINENO+3


    echo "" &gt; /tmp/mydomain.org
    echo "zone \"mydomain.org\" IN {" &gt;&gt; /tmp/mydomain.org
    echo "        type master;" &gt;&gt; /tmp/mydomain.org
    echo "        file \"mydomain.org\";" &gt;&gt; /tmp/mydomain.org
    echo "        allow-update{ none; };" &gt;&gt; /tmp/mydomain.org
    echo "};" &gt;&gt; /tmp/mydomain.org


    sed -i "${ZONELINENO} r/tmp/mydomain.org" /etc/named.conf


    echo "\$TTL 86400" &gt; /var/named/mydomain.org
    echo "\$ORIGIN mydomain.org." &gt;&gt; /var/named/mydomain.org
    echo "" &gt;&gt; /var/named/mydomain.org
    echo "@ IN SOA dns.mydomain.org. info.mydomain.org. (" &gt;&gt; /var/named/mydomain.org
    echo "    2020021200" &gt;&gt; /var/named/mydomain.org
    echo "    3600" &gt;&gt; /var/named/mydomain.org
    echo "    1800" &gt;&gt; /var/named/mydomain.org
    echo "    604800" &gt;&gt; /var/named/mydomain.org
    echo "    86400)" &gt;&gt; /var/named/mydomain.org
    echo "@ IN NS dns.mydomain.org." &gt;&gt; /var/named/mydomain.org
    echo "dns     IN A 192.168.2.3" &gt;&gt; /var/named/mydomain.org
    echo "server1 IN A 192.168.2.11" &gt;&gt; /var/named/mydomain.org
    echo "server2 IN A 192.168.2.12" &gt;&gt; /var/named/mydomain.org


    echo "[Unit]" &gt; /etc/systemd/system/dnsconf.service
    echo "Description=Configure selinux for named" &gt;&gt; /etc/systemd/system/dnsconf.service
    echo "After=firewalld.service" &gt;&gt; /etc/systemd/system/dnsconf.service
    echo "" &gt;&gt; /etc/systemd/system/dnsconf.service
    echo "[Service]" &gt;&gt; /etc/systemd/system/dnsconf.service
    echo "Type=oneshot" &gt;&gt; /etc/systemd/system/dnsconf.service
    echo "ExecStart=/bin/firewall-cmd --add-port=53/tcp" &gt;&gt; /etc/systemd/system/dnsconf.servi
    echo "ExecStart=/bin/firewall-cmd --add-port=53/tcp --permanent" &gt;&gt; /etc/systemd/system/d
    echo "ExecStart=/bin/firewall-cmd --add-port=53/udp" &gt;&gt; /etc/systemd/system/dnsconf.servi
    echo "ExecStart=/bin/firewall-cmd --add-port=53/udp --permanent" &gt;&gt; /etc/systemd/system/d
    echo "ExecStart=rm -f /etc/systemd/system/dnsconf.service" &gt;&gt; /etc/systemd/system/dnsconf
    echo "ExecStart=rm -f /etc/systemd/system/multi-user.target.wants/dnsconf.service" &gt;&gt; /et
    echo "" &gt;&gt; /etc/systemd/system/dnsconf.service
    echo "[Install]" &gt;&gt; /etc/systemd/system/dnsconf.service
    echo "WantedBy=multi-user.target" &gt;&gt; /etc/systemd/system/dnsconf.service


    ln -s /etc/systemd/system/dnsconf.service /etc/systemd/system/multi-user.target.wants/dnsconf.s
    ln -s /etc/systemd/system/named.service /etc/systemd/system/multi-user.target.wants/named.servi
    sync
%end


%anaconda
pwpolicy root --minlen=6 --minquality=1 --notstrict --nochanges --notempty
pwpolicy user --minlen=6 --minquality=1 --notstrict --nochanges --emptyok
pwpolicy luks --minlen=6 --minquality=1 --notstrict --nochanges --notempty
%end
```

Related Posts:

Oracle Compute Cloud – Uploading

Getting started with git behind a company

Deploying CentOS 8 using PXE

Public Cloud consequences with

Relocating base directory Enterprise

ABOUT AUTHOR

FREDERIQUE.RETSEMA@AMIS.NL

Frederique Retsema is active in IT since 1993. Senior Consultant and developer on diverse areas including SQL and Java. She likes to work with automation tools like Bamboo, Jenkins, Ansible, Terraform and CloudFormation.

RELATED POSTS

FEBRUARY 8, 2020 1 DECEMBER 18, 2019 2 NOVEMBER 2, 2019 0

Deploying CentOS 8 using PXE

Linux Shell Script – passing arguments to functions and returning and capturing results

Switch off Screen Lock on Ubuntu Bionic Desktop

Leave a Reply

Enter your comment here...