Opdracht 2: Domain Name Service

Enterprise Linux 14-15

Bachelor toegepaste informatica, HoGent Bedrijf en Organisatie

Inhoudsopgave

Deadline: W7

1 Leerdoelen

- Een DNS-server kunnen opzetten en testen
  - De BIND configuratiebestanden, i.h.b. 'zone files' begrijpen
- Een netwerkservice kunnen opzetten met Ansible
  - Jinja templates kunnen gebruiken: variabelen, conditionals, lussen
  - Door Jinja gegenereerde configuratiebestanden valideren

2 Studiemateriaal

- Stephen Wadeley. 2014. "Red Hat Enterprise Linux 7 Networking Guide"
- Paul Albitz and Cricket Liu. 2001. "DNS and BIND", Fourth Edition. O'Reilly.
- Paul Mockapetris. 1987. RFC 1034: "Domain names: Concepts and Facilities". IETF

3 Opdrachtomschrijving

DNS is essentieel voor de correcte werking van een domein, en redelijk wat (volgens sommigen alle) netwerkproblemen zijn terug te leiden tot DNS. Er zijn verschillende implementaties van DNS, maar veruit de meest gebruikte is BIND.

We gaan een DNS-server opzetten voor ons domein, maar dan moeten we eerst een aantal keuzes maken. Meer bepaald is het nu het moment om een lijstje te maken van de servers in ons netwerk, hun hostnamen en het IP-adres dat we elk gaan toekennen.

Hieronder vind je een voorstel. Het domein wordt hier `linuxlab.net` genoemd en we onderscheiden twee netwerken. Het eerste is de 'DMZ' met publieke IP-adressen, 192.0.2.0/24. Dit adresblok is het zgn. "TEST-NET", dat beschouwd wordt als publiek, maar in principe enkel bestemd voor gebruik in documentatie (zie RFC 5737). Het tweede netwerk is het private, 172.16.0.0/16. Hostnamen in de DMZ krijgen prefix `pu` (voor 'publiek'), die in het private netwerk `pr`.

Je mag zelf je eigen domeinnaam, hostnamen en IP-adressen toekennen. Geef in dat geval een duidelijk overzicht in je verslag. Vermijd het eerste adres in een netwerk, Vagrant geeft hier waarschuwingen over en dit adres wordt in een VirtualBox host-only netwerk typisch toegekend aan het hostsysteem.

Hostnaam	Alias	IP-adres	Functie
`pu001`	`ns1`	192.0.2.2	DNS Master server
`pu002`	`ns2`	192.0.2.3	DNS Slave server
`pu010`	`www`	192.0.2.10	Webserver
`pu020`	`mail`, `smtp`, `imap`	192.0.2.20	Mailserver
`pr001`	`dhcp`	172.16.0.2	DHCP
`pr002`	`moni`, `nagios`	172.16.0.3	Monitoring
`pr010`	`intra`, `intranet`	172.16.0.10	Interne Webserver ("intranet")
`pr011`	`file`	172.16.0.11	Fileserver

Merk op dat de "namen" die de functie van de servers aangeven (`www`, `mail`, enz) niet de hostnamen zijn maar aliassen.

We beginnen met het opzetten van een master server, een slave server is extra (en resulteert in een "upgrade" van je score, zie Evaluatie)

4 Uitwerking

Om je op weg te helpen, volgt hier een plan van aanpak. **Denk er aan**: stap voor stap, test elke tussenstap (bv. via een BATS testscript) en als dit lukt, doe een `git commit` + `push`.

1. Maak de directorystructuur voor je Ansible-rol aan, bv: `mkdir -p ansible/roles/bind/{handlers,tasks,templates}`.
2. Installeer BIND, zet de service aan, configureer de firewall
3. Definieer de variabelen die je zal nodig hebben (ihb. de hostnamen, geassocieerde IP-adressen en aliassen). Zie verderop voor een voorbeeld (`host_vars/pu001`). Begin eventueel eerst met enkel de publieke servers (in de DMZ).
4. Kopieer `/etc/named.conf` van je VM naar je hostsysteem in de `templates` directory van je rol.
5. Zorg er voor dat `/etc/named.conf` aangepast is aan jouw situatie, in het bijzonder de open poorten (`listen-on` optie in het configuratiebestand).
    - Zorg dat bij het kopiëren naar de server de juiste eigenaar en permissies zijn ingesteld!
    - Valideer het configuratiebestand a.h.v. het commando `named-checkconf` (opgeroepen via de optie `validate=` van de Ansible-module `template`)
6. Maak een template aan voor een "forward lookup zone file". Verderop vind je een voorbeeld van hoe zo'n BIND zone-bestand er kan uitzien. "Hard-coded" waarden worden in je template zoveel mogelijk vervangen door variabelen (Jinja-notatie `{{ var }}`).
7. Installeer het bestand op de server, zorg voor de juiste permissies en valideer (met `named-checkzone`). Zorg er voor dat het bestand wordt geladen vanuit `/etc/named.conf`!
8. Maak een template aan voor een "reverse lookup zone file". Je vindt opnieuw verderop een voorbeeld.
9. Installeer het bestand, zorg voor de juiste permissies en valideer
10. Voeg de private servers toe aan de `host_vars` van de DNS-server en breid de functionaliteit van je Ansible rol uit voor het ondersteunen van meerdere reverse lookup zones.

4.1 Voorbeeld `host_vars/pu001`

```yaml
# host_vars/pu001
# vi: ft=yaml
---
bind_listen_ipv4:
  - "any"
bind_listen_ipv6:
  - "any"
bind_allow_query:
  - "192.0.2.0/24"
  - "172.16.0.0/16"

bind_recursion: "no"

bind_zone_name: "linuxlab.net"
bind_zone_networks:
  - ip: "192.0.2"
    reverse: "2.0.192"
  - ip: "172.16"
    reverse: "16.172"

bind_zone_name_servers:
  - "pu001"

bind_zone_mail_servers:
  - name: "mail"
    preference: "10"

bind_zone_hosts:
  - name: pu001
    ip: 192.0.2.2
    aliases:
      - ns1
  - name: pu002
    ip: 192.0.2.3
    aliases:
      - ns2
  - name: pu010
    ip: 192.0.2.10
    aliases:
      - www
  - name: pu020
    ip: 192.0.2.20
    aliases:
      - mail
      - smtp
      - imap

  - name: pr001
    ip: 172.16.0.2
    aliases:
      - dhcp
```

```
  - name: pr002
    ip: 172.16.0.3
    aliases:
      - moni
      - nagios
  - name: pr010
    ip: 172.16.0.10
    aliases:
      - intra
      - intranet
  - name: pr011
    ip: 172.16.0.11
    aliases:
      - file
```

4.2 Voorbeeld zone-bestand `/var/named/linuxlab.net`

Belangrijke punten bij het opmaken van een zonebestand:

- DNS is op zich niet complex. Het komt neer op een databank met enkele tabellen (= zonebestanden) die in (een strak) tekstformaat zijn opgemaakt. Er zijn verschillende types van records, o.a.
  - A voor mapping van hostnaam naar IP-adres
  - CNAME voor een alias
  - PTR voor mapping van IP-adres naar hostnaam (in een “reverse lookup” zonebestand)
  - enz. (lees de documentatie!)
- Velden worden gescheiden door witruimte. Hieronder is alles mooi uitgelijnd, maar dit is niet verplicht.
- Hostnamen die volledig uitgeschreven zijn (fully qualified domain name/FQDN) moeten afgesloten worden met een punt, bv. `pu001.linuxlab.net.`
- Namen die niet met een punt afgesloten worden, worden aangevuld met de waarde van `$ORIGIN` die aan het begin van een zonebestand gegeven wordt. Bv. `pu002` wordt dan `pu002.linuxlab.net.`. Als je een hostnaam volledig uitschrijft en je vergeet het punt, dan zal de domeinnaam dus verkeerd geïnterpreteerd worden (`pu002.linuxlab.net` wordt immers `pu002.linuxlab.net.linuxlab.net.`).
- De `@` in een zonebestand wordt geïnterpreteerd als de waarde van `$ORIGIN`.

```
; Zone file for linuxlab.net
$ORIGIN linuxlab.net.
$TTL 1W
;         primary NS          email address admin
@ IN SOA  pu001.linuxlab.net. hostmaster.linuxlab.net. (
  14101813  ; serial
  1D        ; refresh
  1H        ; retry
  1W        ; expire
  1D )      ; negative caching TTL

                    IN  NS     pu001.linuxlab.net.

@                   IN  MX     10  mail.linuxlab.net.

pu001               IN  A      192.0.2.2
ns1                 IN  CNAME  pu001
```

```
pu002                IN  A      192.0.2.3
ns2                  IN  CNAME  pu002
pu010                IN  A      192.0.2.10
www                  IN  CNAME  pu010
pu020                IN  A      192.0.2.20
mail                 IN  CNAME  pu020
smtp                 IN  CNAME  pu020
imap                 IN  CNAME  pu020
pr001                IN  A      172.16.0.2
dhcp                 IN  CNAME  pr001
pr002                IN  A      172.16.0.3
moni                 IN  CNAME  pr002
nagios               IN  CNAME  pr002
pr010                IN  A      172.16.0.10
intra                IN  CNAME  pr010
intranet             IN  CNAME  pr010
pr011                IN  A      172.16.0.11
file                 IN  CNAME  pr011
```

4.3 Voorbeeld reverse zone bestand `/var/named/2.0.192.in-addr.arpa`

Een “reverse” zonebestand bevat de omgekeerde mappings, dus van IP-adres naar hostnaam. Ook hier moet je rekening houden met enkele eigenaardigheden.

- De notatie van netwerkadressen: ten eerste wordt het host-deel van het netwerkadres niet genoteerd, de getallen in de “dotted quad”-notatie worden omgekeerd en je moet er `in-addr.arpa.` achter schrijven. Met andere woorden, `192.0.2.0/24` wordt als `2.0.192.in-addr-arpa.` geschreven.
- In de mappings (PTR-records) wordt alleen het host-deel van de IP-adressen geschreven, dus 192.0.2.2 wordt “2”.
- In een zone waar je verschillende IP-adresblokken hebt (zoals hier: 192.0.2.0/24 en 172.16.0.0/16) moet je **twee** reverse zonebestanden opmaken.

```
; Reverse zone file for linuxlab.net
$TTL 1W
$ORIGIN 2.0.192.in-addr.arpa.

;         primary NS          email address admin
@ IN SOA  pu001.linuxlab.net. hostmaster.linuxlab.net. (
  14101813  ; serial
  1D        ; refresh
  1H        ; retry
  1W        ; expire
  1D )      ; negative caching TTL

         IN  NS   pu001.linuxlab.net.

2        IN  PTR  pu001.linuxlab.net.
3        IN  PTR  pu002.linuxlab.net.
10       IN  PTR  pu010.linuxlab.net.
20       IN  PTR  pu020.linuxlab.net.
```

5 Evaluatie

Deliverables:

- Labo-verslag met
  - Link naar jullie Bitbucket-repository
  - Toelichting van de gekozen aanpak: welke stappen heb je ondernomen?
  - Gebruikte bronnen voor het uitwerken van de opdracht
- Demo: toon aan dat je server antwoordt op DNS-requests vanop je hostsysteem.

Om de score in de rechterkolom te halen, moet je **alle** taken tot en met de overeenkomstige lijn realiseren.

Taak	Score
Het labo-verslag is aanwezig en volledig	
De server antwoordt op DNS-requests vanop het hostsysteem	voldoende
Deze requests lukken voor de publieke servers: `A`, `CNAME`, `NS`, `MX`	goed
Reverse lookups (PTR) lukken voor de publieke servers	zeer goed
Reverse lookups voor verschillende adresblokken (dus ook private servers)	uitmuntend

Extra's:

Eén van deze bijkomende taken uitvoeren resulteert in een "upgrade" van je score. Voldoende wordt dus goed, ongeacht de vereisten daarvoor, goed wordt zeer goed, enz. Een onvoldoende blijft echter onvoldoende…

- Een `slave` server opzetten.
- Elke deeltaak uitputtend automatisch testen, hetzij met BATS, hetzij met ServerSpec. Met "uitputtend" wordt bedoeld dat je **alle** mogelijke requests naar de DNS-server stuurt, en alle resultaten controleert. Je vraagt m.a.w. de IP-adressen van alle hostnamen op, enz.