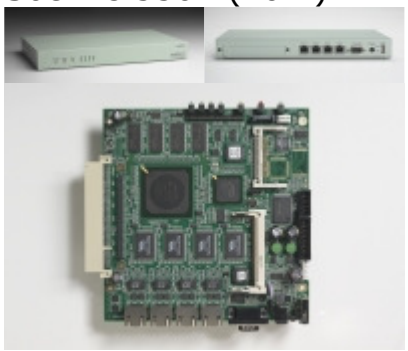
HowTo => OpenBSD => Local Caching DNS + DNSSEC (BIND)

Hardware	=>	Soekris 5501 (10W)



Tools	=>	USB naar Serial Adapter voor Console Putty voor Terminal sessie middels USB Serial Adapter
Operating System	=>	OpenBSD 4.8
Software	=>	BIND

Inleiding:

Door zelf lokaal een caching DNS in te richten zal je netwerkverkeer sneller worden.
Dit komt omdat dan niet voor iedere aanvraag een verbinding naar een externe DNS dient te worden opgezet, dit effect wordt groter met een groter intern netwerk.
Alle interne servers zijn bereikbaar onder hun FQDN (Fully Qualified Domain Name).
Ook de werkplekken die hun IP via DHCP krijgen hebben nu een naam.
DNSSEC is er om risico van cache poisoning en man-in-the-middle attacks te voorkomen.
In eerste instantie laten we dit achterwege tot de basis functioneert, hierna kun je DNSSEC activeren.

Helaas mist geïnstalleerde versie 9.4.2 de benodigde DNSSEC functionaliteit.

Optie: Wacht op nieuwe versie of installeer zelf.
De kans dat er een nieuwere versie komt is gering.
Zelf installeren met ISC versie doet afbreuk aan de zo geroemde security die we van OpenBSD gewend zijn.

BIND werkt voor de rest prima, het is ook het meest geïnstalleerde DNS systeem.
Indien je provider reeds gebruik maakt van DNSSEC is je risico beperkt tot jouw verbinding met hun DNS servers.

De aanbeveling is om UNBOUND te installeren indien je gebruik wenst te maken van DNSSEC.
Zie hiertoe **HowTo OpenBSD Local Caching DNS + DNSSEC (UNBOUND)**

Pre-install:

Zie volgende onderdeel.

Houdt de ingevulde variabelenlijst van de **Bijlage Variabelen** in het inleidende document **HowTo OpenBSD Firewall met Secure Anonymous Access** bij de hand tijdens instalatie/configuratie. Zodra je een variabele ziet, bv `%HOST%` vervang dit dan in zijn geheel door wat je had ingevuld, dus zeker geen %-tekens achterlaten.

Installatie BIND:

De benodigde programmatuur zonder DNSSEC is reeds door de basisinstallatie aanwezig.
Wat gemist wordt is `dnssec-dsfromkey`, dit onderdeel is noodzakelijk voor het ophalen van de Root Key.
We gaan nu de nieuwste sources ophalen bij ISC en zelf compileren.
Hierna plaatsen we het bestand in `/usr/local/sbin` (ergens in search path).

```
root @ 10.0.10.1 - PuTTY [ksh]
# cd /home/install
# wget http://ftp.isc.org/isc/bind9/9.8.0-P1/bind-9.8.0-P1.tar.gz
# tar zxvf bind-9.8.0-P1.tar.gz
# cd bind-9.8.0-P1
# ./configure && make && echo "BUILD OK"
# cd bin/dnssec
# cp -p dnssec-dsfromkey /usr/local/sbin
```

Post-Installatie BIND:

Configuratie gaat middels onderstaande bestanden.
Bewaar eerst het originele bestand.

```
root @ 10.0.10.1 - PuTTY [ksh]
# cp -p /var/named/etc/named.conf /var/named/etc/named.conf.org
```

Algemene configuratie.

```
root @ 10.0.10.1 - PuTTY [vi /var/named/etc/named.conf]
#########################################################################
#######
###                                named.conf
###
#########################################################################
#######

options {
        allow-query { 127.0.0.1; 10.0.10/24; };         # Eigen
range invullen
        allow-recursion { 127.0.0.1; 10.0.10/24; };     # Eigen
range invullen
        allow-transfer { none; };
#        dnssec-enable yes;                             # Zodra Key
opgeslagen
#        dnssec-validation yes;                         # Zodra Key
opgeslagen
       #forward first;
        forwarders { %NS1%; %NS2%; };     # Provider DNS servers
        query-source address 127.0.0.1 port *;
# Mogelijk alleen localhost en LAN/DMZ middels redirect in
pf.conf!!!
        listen-on { 127.0.0.1; 10.0.10.1; };            # Eigen LAN
interface
```

```
        listen-on-v6 { none; };
        version "named";
};

### Signed Root Zone Key voor DNSSEC
#
# Gebruik separaat script voor ophalen en verifieren Root Zone.
# Activeer include regel als je de Key hebt opgeslagen.
# Activeer dnssec regels bij bovenstaande Options.
#include "/etc/root_trusted_key";

### Enable RNDC Commands
#
#controls {
#  inet 127.0.0.1 allow { localhost; };
#};

### Disable RNDC Commands
#
controls { };

logging {
        category lame-servers { null; };
};

zone "." {
        type hint;
        file "etc/root.hint";
};

zone "localhost" {
        type master;
        file "standard/localhost";
        allow-transfer { localhost; };
};

zone "127.in-addr.arpa" {
        type master;
```

```
        file "standard/loopback";
        allow-transfer { localhost; };
};

zone "%DOMEIN%" {                                       # Eigen
domein
        type master;
        file "master/db.%DOMEIN%";                      # Eigen
domein
                                allow-update { none; };
                        };

                        zone "10.0.10.in-addr.arpa" {
# Eigen range invullen
                                type master;
                                file "master/db.10.0.10";
# Eigen range invullen
                                allow-update { none; };
                        };
```

Forward Resolving File
Commentaar met # geeft error, dus hier alleen voor verduidelijking.

```
host1.%DOMEIN%.          86400   IN   A    10.0.10.200
host2.%DOMEIN%.          86400   IN   A    10.0.10.201
```

root @ 10.0.10.1 - PuTTY [vi /var/named/master/db.%DOMEIN%]

```
%DOMEIN%.                86400   IN   SOA  wodan.%DOMEIN%.
root.%HOST%.%DOMEIN%. ( 1 10800 3600 6044800 86400 )
                         86400   IN   NS   %HOST%.%DOMEIN%.

%HOST%.%DOMEIN%.         86400   IN   A    10.0.10.1
dhcp1.%DOMEIN%.          86400   IN   A    10.0.10.10
dhcp2.%DOMEIN%.          86400   IN   A    10.0.10.11
dhcp3.%DOMEIN%.          86400   IN   A    10.0.10.12
```

```
dhcp4.%DOMEIN%.          86400   IN   A    10.0.10.13
dhcp5.%DOMEIN%.          86400   IN   A    10.0.10.14
```

Reverse Lookup File

```
200.10.0.10.in-addr.arpa.     86400   IN   PTR   host1.%DOMEIN%.
201.10.0.10.in-addr.arpa.     86400   IN   PTR   host2.%DOMEIN%.
```

root @ 10.0.10.1 - PuTTY [vi /var/named/master/db.10.0.10]

```
10.0.10.in-addr.arpa.         86400   IN   SOA   %HOST%.%DOMEIN%.
root.%HOST%.%DOMEIN%. ( 1 10800 3600 6044800 86400 )
                              86400   IN   NS    %HOST%.%DOMEIN%.

1.10.0.10.in-addr.arpa.       86400   IN   PTR   %HOST%.%DOMEIN%.
10.10.0.10.in-addr.arpa.      86400   IN   PTR   dhcp1.%DOMEIN%.
11.10.0.10.in-addr.arpa.      86400   IN   PTR   dhcp2.%DOMEIN%.
12.10.0.10.in-addr.arpa.      86400   IN   PTR   dhcp3.%DOMEIN%.
13.10.0.10.in-addr.arpa.      86400   IN   PTR   dhcp4.%DOMEIN%.
14.10.0.10.in-addr.arpa.      86400   IN   PTR   dhcp5.%DOMEIN%.
```

Script voor ophalen en controle Trusted Key.
Plaats het tevens in cron voor wekelijkse controle en output naar /var/log/named/dnssec_verify_rootzone.log.

root @ 10.0.10.1 - PuTTY [vi /var/named/etc/dnssec_verify_rootzone.sh]

```
#####################################################################
#######
###                      dnssec_verify_rootzone.sh
###
#####################################################################
#######

# Gebaseerd op script van Calomel.org DNSSEC Root Zone

echo ""
echo "DNSSEC Root Zone Verificatie"
echo ""

# Plek waar de Key wordt opgeslagen
roottrustedkey="/var/named/etc/root_trusted_key"

# Ga naar /tmp en verwijder ouwe bestanden
# Beter nog contole toevoegen of pwd=/tmp
cd /tmp
rm -rf root-*

# Ophalen Root Zone DNSKEY
echo "Ophalen Root Zone DNSKEY middels DNS (root-ds)"
dig +noall +answer DNSKEY . > root-dnskey
dnssec-dsfromkey -f root-dnskey . > root-ds

# Ophalen IANA Key Digest
echo "Ophalen IANA Key Digest middels https (root-anchors)"
wget -q https://data.iana.org/root-anchors/root-anchors.xml
```

```
# Filteren Digest om output duidelijk te zien
rootds=`cat root-ds | grep "8 2" | awk '{print $7 $8}'`
rootanchors=`cat root-anchors.xml | grep \<Digest\> | sed
's/<Digest>//' |sed 's/<\/Digest>//'`

# Print beide Digests ter controle
echo ""
echo "Vergelijk Visueel de Digests:"
echo -n "root-ds     : "; echo $rootds
echo -n "root-anchors: "; echo $rootanchors
echo ""

# Controleer of "Root Zone DNSKEY" en de "IANA Digest" Gelijk zijn
if [ $rootds = $rootanchors ]
 then
  echo "GECONTROLEERD: Correcte Signature. Digests zijn Gelijk."

  # Filter de DNSKEY
  keytype=`cat root-dnskey | grep 257 | awk '{print $5" "$6" "$7 }'`
  trustedkey=`cat root-dnskey | grep 257 | awk '{print substr($0,
index($0,$8)) }'`

  # Plaats de DNSKEY in het door named.conf benodigde Include
Bestand
  echo "managed-keys {" > $roottrustedkey
  echo "   \".\" initial-key $keytype \"$trustedkey \";" >>
$roottrustedkey
  echo "};" >> $roottrustedkey
  echo " "
  echo "Klaar. De key is in $roottrustedkey"

 else
  echo "MISLUKT: FOUTIEVE Signature. GEEN MATCH !!"
  # Indien gewenst kun je hier mail naar beheer toevoegen
  exit
fi
```

Wijzig de te raadplegen Nameserver.

root @ 10.0.10.1 - PuTTY [vi /etc/resolv.conf]

```
nameserver 127.0.0.1
nameserver 10.0.10.1
```

Opstarten van de daemon gaat automatisch voor ip4 door deze toevoeging.

root @ 10.0.10.1 - PuTTY [vi /etc/rc.conf.local]

```
named_flags="-4"                # IP4
```

Links:

http://www.openbsd.org/
http://www.chiark.greenend.org.uk/~sgtatham/putty/

Boeken:

DNS and BIND