

**Installation et configuration de DNS**

Projet m2l.org

DUMAS Lucie

Table des matières

[LE DNS 3](#_Toc146610564)

[Qu’est-ce qu’un DNS ? 3](#_Toc146610565)

[Quelle est la différence entre un serveur d’autorité et un résolveur DNS ? 3](#_Toc146610566)

[Le DNS Master 4](#_Toc146610567)

[Le DNS Slave 8](#_Toc146610568)

[Vérification d’état et débogage de services 11](#_Toc146610569)

# LE DNS

## Qu’est-ce qu’un DNS ?

Le DNS est un serveur permettant d’associer une adresse logique, appelée un nom de domaine (par exemple www.google.com) à une adresse IP (par exemple 8.8.8.8). Ces adresses sont uniques. Il existe plusieurs types de DNS : les serveurs d’autorité et les résolveurs DNS

## Quelle est la différence entre un serveur d’autorité et un résolveur DNS ?

Un serveur DNS d’autorité est un serveur DNS dont le rôle est de gérer et répondre aux requêtes DNS concernant un domaine particulier. Il est responsable de stocker les enregistrements DNS pour ce domaine et de répondre aux requêtes DNS concernant ce domaine. Les serveurs DNS d'autorité sont la source d'autorité pour les domaines qu'ils gèrent.

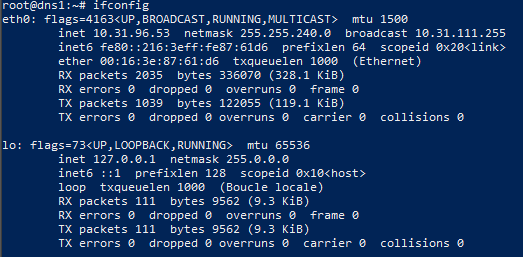
Un résolveur DNS est un serveur DNS dont le rôle est de faire des requêtes DNS au nom des clients (comme les navigateurs Web ou les applications) pour résoudre des noms de domaine en adresses IP. Le résolveur interroge d'autres serveurs DNS, y compris les serveurs d'autorité, pour obtenir les réponses DNS correctes.

Ces deux types de serveur ont donc des rôles très différents mais sont pourtant complémentaires

# Le DNS Master

Pour créer notre conteneur DNS Master, nous allons dupliquer notre conteneur template :

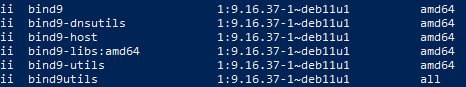
lxc-copy -n template -N dns1

 Nous nous connectons à notre conteneur et changeons son adresse IP dans le fichier /etc/network/interfaces. Le DNS écoutant par défaut le port 53, son adresse IP sera 10.31.96.53.

Nous installons ensuite les paquets nécessaires à l’outil bind, responsable de la gestion de DNS :

apt update && apt upgrade

apt install bind9 bind9utils dnsutils





Nous devons ensuite créer les 3 fichiers suivants :

/etc/bind/named.conf.local (déclaration des zones directes et inverses) :

# Création d’une zone

zone ‘m2l.org’ IN {

# Définition de ce DNS comme master pour cette zone

type master ;

# Lien vers le fichier regroupant les informations nécessaires

file ‘/etc/bind/db.m2l.org’ ;

} ;

/etc/bind/named.conf.options (description des options du serveur DNS) :

options {

# Définition du chemin absolu du serveur

directory ‘/var/cache/bind’;

# Accepter les requêtes pour toutes les machines

allow-query { any; };

# Le serveur fournit des réponses récursives si demandées par les client

recursion yes;

# Le résolveur ne tente pas de valider les réponses des zones DNSSEC

dnssec-validation no;

# Transmission des requêtes à 8.8.8.8 ou 8.8.4.4 si le DNS ne sait pas résoudre les adresses

forwarders { 8.8.8.8; 8.8.4.4; };

forward only;

};

/etc/bind/db.m2l.org (description de la zone ‘m2l.org’) :

@ IN SOA ns1.m2l.org. root.m2l.org. (

2020122601;

43200;

3600;

3600000;

172800 );

# Adresse du serveur web

@ IN A 10.31.96.80;

# Nom des DNS

@ IN NS ns1.m2l.org.;

@ IN NS ns2.m2l.org.;

# Adresse des DNS

ns1 IN A 10.31.96.80;

ns2 IN A 10.31.54;

# Adresse des machines

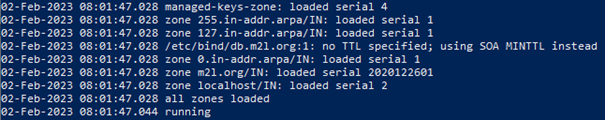
www IN A 10.31.96.80;

# Alias

console IN CNAME www;

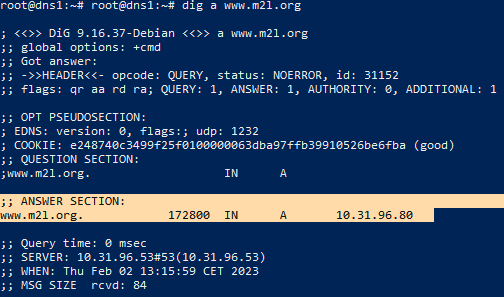
Pour créer des fichiers de log séparés pour bind, nous pouvons nous référer à la documentation : Vérification d’état et débogage de services.

Une fois les fichiers de log de bind séparés, nous allons vérifier que le DNS fonctionne correctement :



Nous utilisons également la commande dig (Domain Information Groper) qui nous permet d’interroger des serveurs de noms en effectuant une recherche DNS et en nous affichant le résultat :

dig a www.m2l.org



# Le DNS Slave

Pour créer le DNS Slave, nous commençons par copier le DNS Master :

lxc-copy -n dns1 -N dns2

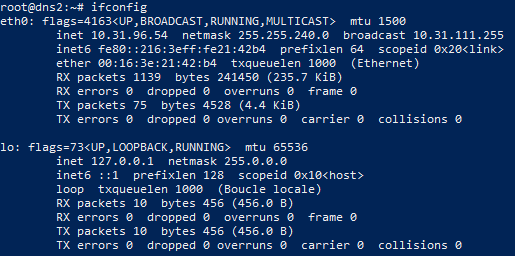
Nous changeons également la configuration du conteneur pour que ce dernier démarre automatiquement au démarrage du serveur. Pour cela, nous rajoutons la ligne suivante au fichier /var/lib/lxc/dns2/config :

lxc.start.auto = 1

Nous nous connectons ensuite au DNS Slave et nous changeons son adresse IP ainsi que son DNS dans le fichier /etc/network/interfaces :

address 10.31.96.54/20

dns-nameservers 10.31.96.53



Nous pouvons maintenant configurer notre DNS Slave. Le but d’un DNS Slave est de demander des informations au DNS Master. Nous devons donc supprimer le fichier /etc/bind/db.m2l.org car le but de notre DNS Slave sera de le demander auprès du DNS Master :

rm /etc/bind/db.m2l.org

Nous pouvons ensuite modifier le fichier /etc/bind/named.conf.local :

# Création d’une zone

zone ‘m2l.org’ IN {

# Définition de ce DNS comme slave pour cette zone

type slave;

# Lien vers le fichier regroupant les informations nécessaires

file ‘/etc/bind/db.m2l.org’;

# Adresse du DNS possédant le fichier /etc/bind/db.m2l.org

masters { 10.31.96.53; };

} ;

Le DNS Slave est maintenant configuré. Pour que le DNS Slave soit reconnu par le DNS Master, nous devons faire des modifications sur ce dernier. Nous commençons par déclarer le second DNS dans le fichier db.m2l.org du DNS Master :

@ IN NS ns2.m2l.org.;

ns2 IN A 10.31.86.54;

Nous devons ensuite autoriser le DNS Slave à récupérer les bases de données des zones dans le fichier named.conf.local (toujours dans le DNS Master) :

zone ‘m2l.org’ IN {

type master;

file ‘/etc/bind/db.m2l.org’;

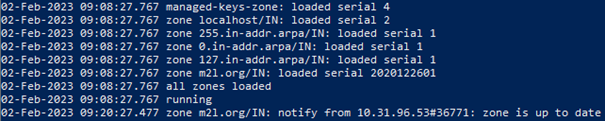
allow-transfert { localhost; 10.31.96.54; };

notify yes;

};

Pour vérifier que les fichiers de zone ne comportent pas d’erreurs de syntaxes, nous allons entrer la commande suivante dans nos conteneurs DNS :

named-checkzone

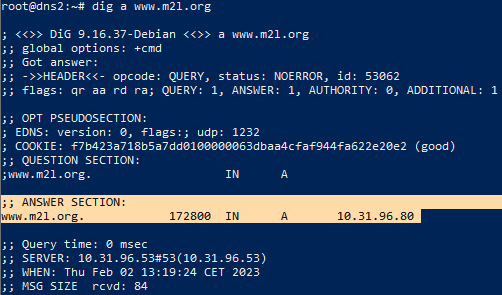


Pour vérifier que les fichiers de configuration named .conf. ne comportent aucune erreur de syntaxe, nous utilisons la commande suivante dans nos conteneurs DNS :

named-checkconf

Nous utilisons également la commande dig afin de nous assurer que le DNS Slave assure la résolution de nom :

dig a www.m2l.org



Une fois que nous nous sommes assurés que les DNS soient fonctionnels, nous allons changer le DNS de toutes nos machines :

nameserver 10.31.96.53

nameserver 10.31.96.54

# Vérification d’état et débogage de services

Pour créer des fichiers de logs dédiés à l’outil bind, nous devons dans un premier temps créer les dossiers qui accueilleront ces derniers :

# Création du dossier /var/log/bind et de tous les dossiers parents si ces derniers n’existent pas

mkdir -p /var/log/bind

# Réattribution de la propriété du dossier à l’utilisateur bind

chown bind /var/log/bind

Nous ajoutons ensuite dans le fichier named.conf.options les catégories suivantes :

logging {

channel transfers {

file ‘/var/log/bind/transfers’ version 3 size 10M;

print-time yes;

severity info;

};

channel notify {

file ‘/var/log/bind/notify’ version 3 size 10M;

print-time yes;

severity info;

};

channel dnssec {

file ‘/var/log/bind/dnssec’ version 3 size 10M;

print-time yes;

severity info;

};

channel query {

file ‘/var/log/bind/query’ version 3 size 10M;

print-time yes;

severity info;

};

channel general {

file ‘/var/log/bind/general’ version 3 size 10M;

print-time yes;

severity info;

};

channel slog {

syslog security;

severity info;

};

category xfer-out { transfers; slog; };

category xfer-in { transfers; slog; };

category notify { notify; };

category lame-servers { general; };

category config { general; };

category default { general; };

category security { general; slog; };

category dnssec { dnssec; };

category queries { query; };

};