MSE | MASTER OF SCIENCE IN ENGINEERING

Hes·so
Haute Ecole Spécialisée de Suisse occidentale
Fachhochschule Westschweiz
University of Applied Sciences and Arts Western Switzerland

Master of Science HES-SO in Engineering

Technologies de l'information et de la communication

Network Security and Architecture

Travail Pratique

Attaques DNS, Sécurisation DNSSec

François Buntschu
francois.buntschu@hefr.ch

Haute école d'ingénierie et d'architecture de Fribourg (HEIA-FR)

HES-SO//Master, 21 octobre 2020, v1.5

Network Security and Architecture

Travail Pratique - Attaques DNS, Sécurisation DNSSec

Table des matières

1 Introduction

Ce travail pratique consiste dans un premier temps à mettre en œuvre une infrastructure DNS et de tester diverses attaques sur celle-ci. Dans un deuxième temps, le protocole DNSSec sera mis en place afin de sécuriser l'infrastructure et divers tests et mesures seront effectués pour en comprendre son fonctionnement.

2 Objectifs

À la fin de ce travail pratique, l'étudiant sera en mesure de :

— Configurer et installer un serveur BIND
— Sécuriser la configuration d'un serveur de nom
— D'auditer et d'effectuer diverses attaques sur le service DNS
— De sécuriser une zone au travers du protocole DNSSec
— D'auditer et de valider le fonctionnement de DNSSec

3 Préparations

Pour la réalisation de ce travail pratique, l'infrastructure suivante sera mise en place :

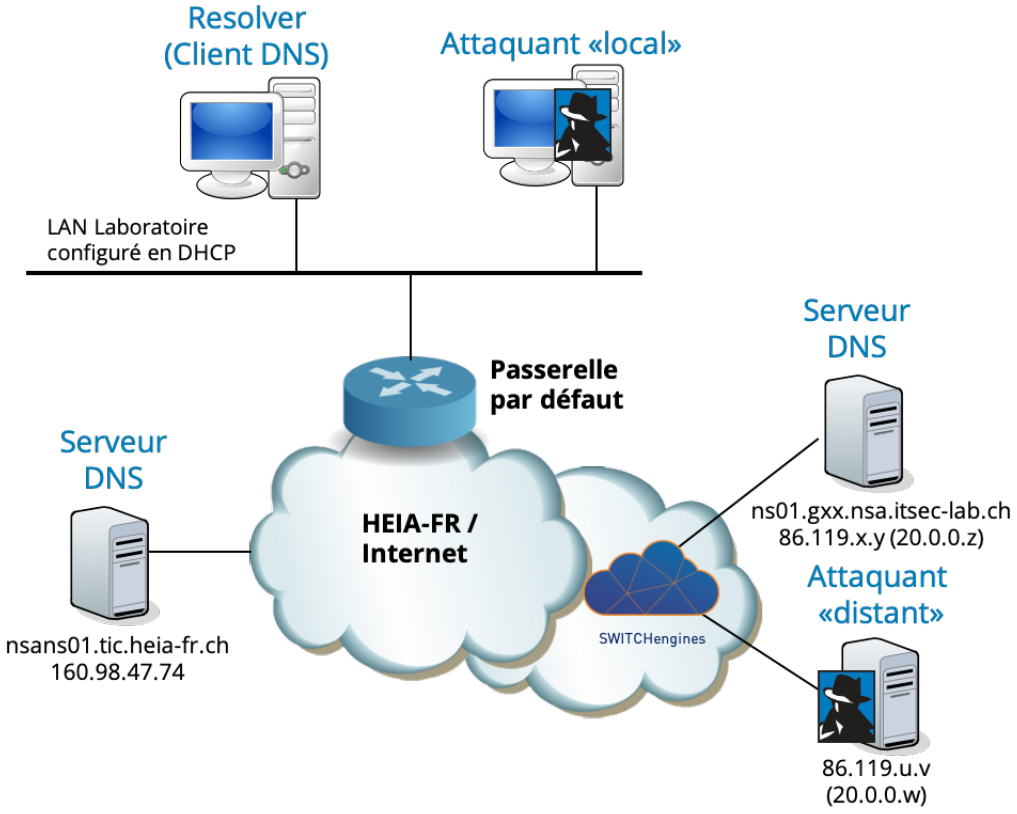

Figure 1 – *Schéma logique*

Les machines installées et gérées par les étudiants sont le client DNS, un serveur DNS hébergé chez SwitchEngines [2] et deux machines pour l'attaquant (locale et distante, fonctionnant avec Kali [12]). Les machines locales utilisent le DHCP du laboratoire pour leur configuration IP. Les adresses IP et les paramètres d'accès sur les machines hébergées chez SwitchEngines sont dans l'annexe A.1 ou sont visibles sur votre compte SwitchEngines.

Le serveur DNS sera une machine Linux (VM Ubuntu). Chaque étudiant doit configurer ce serveur DNS comme serveur autoritaire (primaire) pour la zone **g*x*.nsa.itsec-lab.ch**, avec ***x*** étant le numéro du groupe, par exemple **g1.nsa.itsec-lab.ch** pour le groupe 1.

Le serveur secondaire pour cette zone sera le serveur DNS `nsans01.tic.heia-fr.ch` (`160.98.47.74`). Ce serveur DNS de test est à disposition des étudiants pour effectuer des requêtes et valider vos configurations.

Attention : il n'accepte d'effectuer des résolutions récursives que pour les adresses IP de la HEIA-FR (`160.98.0.0/16`) et des transferts de zones que vers votre serveur DNS.

Votre serveur DNS local pourra aussi faire office de serveur secondaire pour une ou plusieurs zones de vos collègues. À vous de vous coordonner et de documenter vos configurations.

4 Configurations & Questions

4.1 Configuration initiale

La première partie de cet exercice pratique consiste à configurer le domaine **gx.nsa.itsec-lab.ch** sur votre serveur DNS. L'annexe A.2 décrit l'installation et la configuration d'un domaine (d'une zone) avec le serveur BIND [15].

> **Note :** Assurez-vous que le client DNS utilise votre serveur DNS pour la résolution de nom et que la plage d'adresse dont fait parti votre client soit autorisée à faire des demandes récursives sur votre serveur (cf. Annexe A.2).

Question P1
Documentez votre installation.

Question P2
Validez le bon fonctionnement de votre infrastructure (*fingerprinting*, transfert de zone, etc.)

Question P3
En théorie, quel(s) mécanisme(s) avez-vous à disposition pour sécuriser le transfert de zone entre votre serveur et le serveur secondaire ?

4.2 Attaques sur le service DNS

Pour rappel, les attaques principales [4] sur le DNS sont :

— Interception de paquets et modification

— *Cache poisoning*

— DDoS

Pour la réalisation de cette partie du travail pratique, vous avez à choix à disposition les outils Scapy [5], `dnsspoof` [13], `arpspoof` [13], `metasploit` [3], `bettercap`[1] et/ou `netwox` [7].

4.2.1 Interception de paquets, DNS Hijacking

Cette attaque consiste à intercepter les requêtes DNS provenant du client et de répondre "avant" le serveur DNS local. L'idée est d'utiliser l'un des outils suivants : `dnsspoof` [13], `ettercap`, `netwox` [7] et/ou `bettercap` [1] [1]. Certains outils ne supportent pas le protocole IPv6, il est conseillé de le désactiver au niveau de l'interface du client.

Si votre serveur DNS répond toujours avant votre machine attaquante, vous pouvez filter les requêtes sortant de celle-ci pour le domaine concerné (dans cet exemple *heia-fr*) en insérant une règle `iptables` sur votre machine Kali :

```
iptables -I FORWARD -p udp --dport 53 -m string --algo bm --string "heia-fr" -j DROP
```

Question P4
Quel est le scénario de votre attaque ?

Question P5
Quels sont les messages échangés ? Validez votre attaque par des mesures (état des tables, capture wireshark, etc.).

Question P6
Quel(s) est (sont) la(les) contre-mesure(s) théorique de cette attaque ?

4.2.2 DNS Server Cache poisoning

L'attaque précédente se concentre sur la machine de l'utilisateur. Afin d'obtenir un effet sur le plus long terme, chaque fois que la machine de l'utilisateur envoie une requête DNS, l'attaquant doit répondre par une fausse réponse (*spoofed*), ce qui est contraignant. Un moyen plus efficace est de s'attaquer au serveur DNS.

Lorsqu'un serveur DNS reçoit une requête, et que le nom recherché ne se trouve pas dans le domaine servi par ce serveur, celui-ci va lancer la résolution récursive.

Dans ce laboratoire, le serveur DNS répondra aux requêtes concernant les machines dans le domaine **gx.nsa.itsec-lab.ch** et effectuera une résolution récursive pour les autres domaines (par ex. `www.google.com`).

Pour rappel, avant de démarrer la résolution récursive, le serveur va consulter son propre cache. S'il y trouve une entrée, il va répondre directement. Dans le cas contraire, il va effectuer la résolution, répondre au client et stocker l'information dans son cache.

Ainsi, si l'attaquant réussit à répondre à la place des autres serveurs, le serveur DNS local va mettre une information erronée dans son cache pour une certaine durée. Les prochaines requêtes des clients concernant la résolution de ce nom de machine vont recevoir une information erronée provenant du cache du serveur.

L'empoisonnement va être actif le temps que l'information dans le cache expire

Question P7
Représentez le diagramme de principe de cette attaque.

Vous pouvez utiliser pour cette attaque la suite d'outils `netwox`, l'outil no 105. Avant de lancer l'attaque, assurez-vous que le cache du serveur est vide en utilisant la commande suivante :

```
# sudo rndc flush
```

1. Si ces outils ne sont pas présents sur votre distribution Kali, installez-les avec la commande `apt install dsniff netwox ettercap-text-only bettercap`

Assurez-vous aussi que le cache DNS du client est effacé avec les commandes suivantes sur Ubuntu :

```
# systemd-resolve --flush-caches
```

Et sur Windows :

```
c:> ipconfig \flushdns
```

La différence entre cette attaque et la précédente est le destinataire de la réponse erronée, dans ce cas, c'est le serveur DNS local qui est visé.

Pour faire un *dump* du cache et voir son contenu, utilisez les commandes suivantes :

```
# sudo rndc dumpdb -cache
# sudo cat /var/cache/bind/dump.db
```

Question P8
Quel sera le paramétrage de l'outil `netwox`?

Question P9
Démontrer par la mesure l'état du cache ou par d'autres moyens que votre attaque a réussis (avec Wireshark).

4.3 Configuration du DNSSec

Une fois l'infrastructure en place, soit le serveur DNS, on vous demande de sécuriser la zone **gx.nsa.itsec-lab.ch** en la signant au moyen du DNSSec.
Veuillez vous référer à l'annexe A.3 pour les détails de la configuration.

Question P10
Validez que votre serveur vous retourne les clés de zone.

Question P11
Validez que vos différents RR sont signés (utilisez les commandes `drill` et `dig`).

Question P12
Que devez-vous encore configurer pour que la chaine de confiance (*Chain of trust*) complète soit en place?

Question P13
Validez votre configuration depuis le site `http://www.dnsviz.net` et au travers de l'outil *drill*, comme décrit dans l'annexe A.3.

Question P14
Effacez le cache de votre serveur et lancer une demande de résolution d'un domaine sécurisé par DNSSec, comme par exemple `www.switch.ch`. Mesurez les échanges entre votre serveur local et l'Internet. Représenter par un diagramme les échanges effectués et documenter les particularités de ceux-ci.

Question P15
Le DNSSec permet d'intégrer de nouvelles fonctionnalités dans la sécurisation des infrastructures. Décrivez brièvement quelles applications les RFC 6698 [11] et RFC 6844 [10] veulent sécuriser au travers du DNSSec et leurs principes de base

A Annexes

A.1 Serveurs distants

Cette annexe contient les adresses IP des machines virtuelles hébergées chez SwitchEngines [2], ainsi que les identifiants pour s'y connecter.

Groupe	Serveur	Adresse IP
1	Serveur DNS	86.119.31.151
	Hacker	86.119.31.126
2	Serveur DNS	86.119.31.49
	Hacker	86.119.31.154
3	Serveur DNS	86.119.31.38
	Hacker	86.119.31.142
4	Serveur DNS	86.119.31.147
	Hacker	86.119.31.196
5	Serveur DNS	86.119.31.183
	Hacker	86.119.31.228
6	Serveur DNS	86.119.31.173
	Hacker	86.119.31.206
7	Serveur DNS	86.119.31.236
	Hacker	86.119.31.194
8	Serveur DNS	86.119.31.218
	Hacker	86.119.31.176
9	Serveur DNS	86.119.31.224
	Hacker	86.119.31.201
10	Serveur DNS	86.119.31.152
	Hacker	86.119.31.56
11	Serveur DNS	86.119.31.212
	Hacker	86.119.31.233
12	Serveur DNS	86.119.30.143
	Hacker	86.119.31.115

Table 1 – *Groupe et adresses IP*

Serveur	Username	Password
Serveur DNS	user	H31a-fr$
Hacker	user	H31a-fr$

Table 2 – *Identifiants pour l'accès aux serveurs distants*

A.2 Configuration d'un serveur avec DNS

Cette annexe décrit la procédure pour l'installation d'un serveur de nom (BIND) et la configuration d'un domaine. Dans cet exemple de configuration, le serveur sera primaire pour la zone **msensa.ch** et pour la zone inverse du bloc d'adresses **160.98.34.0/24**. Plusieurs sources [6][8][9][14] ont été utilisées pour créer ce mode d'emploi.

A.2.1 Conditions initiales, exemple

- Version BIND : 9.16.1
- OS : Ubuntu 20.04 Server, 64 bits
- Domaine configuré : `msensa.ch`
- Adresse IP du serveur utilisé : `160.98.34.99`

A.2.2 Installation des packages

```
$ sudo apt-get install bind9
```

L'installation de BIND avec `apt-get` installe automatiquement le fichier `/etc/bind/rndc.key` pour permettre à `rndc` (*Remote Name Daemon Control*) de s'y connecter pour le piloter.

A.2.3 Configuration du DNS

Serveur master
Pour utiliser *rndc* qui permet la gestion du BIND, il faut ajouter une ligne à fin du fichier de configuration `/etc/bind/named.conf` :

```
include "/etc/bind/rndc.key";
```

Il faut éventuellement générer des clés (`http://tecadmin.net/configure-rndc-for-bind9/`). Afin d'améliorer la sécurité de BIND, nous allons indiquer les directives pour interdire les requêtes récursives provenant d'autres serveurs et cacher le numéro de version de BIND dans le fichier de configuration `/etc/bind/named.conf.options`, dans le bloc `options`.

```
allow-recursion { localhost; 160.98.0.0/16; };
version "it's secret";
dump-file "/var/cache/bind/dump.db";
masterfile-format text;
```

Redémarrage du serveur BIND :

```
$ sudo /etc/init.d/bind9 restart
```

Affichage du log :

```
$ sudo tail -f /var/log/syslog
```

Création de la structure des répertoires :

```
$ sudo mkdir /etc/bind/zones
$ sudo mkdir /etc/bind/zones/master
$ sudo mkdir /etc/bind/zones/slave
```

Modification du fichier `/etc/bind/named.conf.local`, en y ajoutant la zone gérée par le serveur (master) :

```
      ”msensa.ch.” IN {
            type master;
            file ”/etc/bind/zones/master/db.msensa.ch.zone”;
            allow-update { none; };
            allow-transfer { 160.98.47.74;};
            notify yes;
    };
```

Création du fichier `db.msensa.ch.zone` (dans le répertoire `/etc/bind/zones/master`) :

```
$ORIGIN msensa.ch.
$TTL 86400
@   IN   SOA   msensa.ch. root.msensa.ch. (
                                          2018103001
                                          3600
                                          900
                                          604800
                                          86400 )

; Descriptions of names servers of this domain (primary and secondary)
                    IN   NS     ns1.msensa.ch.
                    IN   NS     nsans01.tic.heia-fr.ch.
                    IN   MX 10  160.98.34.100

; List of known hosts in this domain
ns1                 IN   A      160.98.34.99
mail                IN   A      160.98.34.100
www                 IN   CNAME  ns1
```

La configuration de la zone pour le *Reverse DNS* n'est pas nécessaire pour la réalisation de cet exercice pratique. Elle est uniquement présente à titre indicatif.
Modification du fichier `/etc/bind/named.conf.local`, en y ajoutant la zone gérée par le serveur (master) :

```
zone ”34.98.160.in-addr.arpa.” IN {
        type master;
        file ”/etc/bind/zones/master/db.34.98.160.in-addr.zone”;
        allow-update { none; };
        allow-transfer { 160.98.47.74;};
        notify yes;
};
```

Création du fichier `db.34.98.160.in-addr.zone` pour le reverse DNS (dans le répertoire `/etc/bind/zones/master`) :

```
$TTL 86400
@   IN   SOA   msensa.ch. root.msensa.ch. (
                                          2018103001
                                          3600
                                          900
                                          604800
                                          86400 )

; Descriptions of names servers of this domain
                    IN   NS      ns1.msensa.ch.
                    IN   NS      nsans01.tic.heia-fr.ch.

; List of known hosts in this domain

99                  IN   PTR     ns1.msensa.ch.
100                 IN   PTR     mail.msensa.ch.
```

Redémarrage du serveur pour prendre en compte ces fichiers de configuration :

```
$ sudo rndc reload
```

A.3 Configuration et activation du DNSSec

Cette annexe décrit la procédure pour l'activation du DNSSec sur un domaine. Il s'agit de la suite de la configuration décrite dans l'annexe A.2.

Activation du DNSSec

Ajoutez l'option suivante dans le fichier `named.conf.options` :

```
dnssec-enable yes;
```

Génération des clefs KSK et ZSK

- **ZSK** : *Zone Signing Key* - permet la signature des 'Ressource Records dans les fichiers de zone.
- **KSK** : *Key Signing Key* - permet la signature de la ZSK. Il s'agit généralement d'une clef de plus grande taille.

Dans cette configuration, les fichiers de zone sont dans les répertoires : `/etc/bind/zones/*` et les fichiers de clés seront stockés dans des répertoires différents pour plus de visibilité :

- `/etc/bind/ZSK` : toutes les clés de zones
- `/etc/bind/KSK` : toutes les clés de clés

Note : idéalement vous devez stocker vos clés sur une machine *offline*, signer vos zones sur cette machine et transférer ces zones sur votre serveur en production. Cependant, pour les serveurs qui ont des zones qui sont manipulées chaque jour, ceci amènerait trop de travail.

Génération de la Zone Signing Key (ZSK) pour msensa.ch :

```
$ sudo mkdir /etc/bind/ZSK
$ sudo cd /etc/bind/ZSK
$ sudo dnssec-keygen -r /dev/urandom -a RSASHA256 -b 1024 -n ZONE msensa.ch
```

Deux fichiers seront gérés utilisant le format : `K<zone>.<id>.key` (par ex. `Kmsensa.ch.+008+16188.key`) et `K<zone>.<id>.private` (par ex. `Kmsensa.ch.+008.+16188.private`). Le premier fichier contient la clé publique et le fichier `.private` contient la clé privée.

Génération de la Key Signing Key (KSK) pour msensa.ch :

```
$ sudo mkdir /etc/bind/KSK
$ sudo cd /etc/bind/KSK
$ sudo dnssec-keygen -r /dev/urandom -a RSASHA256 -b 4096 -n ZONE -f KSK msensa.ch
```

Cette commande va générer deux fichiers selon la même syntaxe utilisée pour la ZSK.
Note : chaque fois que vous regénérez une KSK ou ZSK, un nouveau nombre aléatoire (ID) sera assigné au fichier.

Ajout de la clé publique sur la zone

Après la génération de la ZSK et KSK, nous devons ajouter ces deux clés publiques sur la zone. Dans cet exemple, cela signifie :

```
$ sudo cat /etc/bind/ZSK/Kmsensa.ch.*.key >> \
      /etc/bind/zones/master/db.msensa.ch.zone
$ sudo cat /etc/bind/KSK/Kmsensa.ch.*.key >> \
      /etc/bind/zones/master/db.msensa.ch.zone
```

Nous avons utilisé ici `Kmsensa.ch.*.key` parce qu'on ne peut pas prédire l'ID. Cependant, si vous regénérez vos clés vous aurez plusieurs fichiers → attention de supprimer les "vieilles" clés avant d'ajouter les clés publiques la zone.
Si vous regardez le fichier de configuration de la zone, vous verrez 2 entrées `DNSKEY` qui ont été ajoutées.

```
msensa.ch. IN DNSKEY 256 3 8 "random chars"
msensa.ch. IN DNSKEY 257 3 8 "random chars"
```

La première clé `DNSKEY`, avec le numéro "256", est la ZSK et la seconde clé avec le numéro "257" est la KSK.
N'oubliez pas d'incrémenter le numéro de série de la zone afin de mettre à jour le(s) serveur(s) secondaire(s). Il est ensuite nécessaire de redémarrer le serveur pour tenir compte de ces clés.

```
$ sudo rndc reload
```

Contrôle de la zone msensa.ch avec les clés

```
$ sudo dig @localhost DNSKEY msensa.ch -t dnskey
```

Signature de la zone
Pour signer la zone, il faut utiliser la commande suivante, qui met en relation le fichier de zone, la ZSK et la KSK :

```
$ sudo mkdir /etc/bind/SET
$ sudo cd /etc/bind/SET
$ sudo dnssec-signzone -o msensa.ch -k /etc/bind/KSK/Kmsensa.ch.+008+42436.key \
   /etc/bind/zones/master/db.msensa.ch.zone /etc/bind/ZSK/Kmsensa.ch.+008+16188.key
```

- Le premier paramètre, "`-o`", indique la zone à signer (dans notre cas, `msensa.ch`).
- Le deuxième paramètre , "`-k`", pointe vers le fichier contenant la KSK.
- Le troisième paramètre est la localisation du fichier de définition de la zone, suivi de la localisation du fichier contenant la ZSK.

Note :

- les IDs des fichiers de clés varient en fonction de la configuration.
- ne pas oublier d'incrémenter le numéro de série de la zone avant de la signer

Un fichier sera généré dans `/etc/bind/SET`, il se nomme `dsset-msensa.ch`. Nous plaçons ce fichier dans un répertoire distinct (SET), afin d'avoir une meilleure visibilité sur `/etc/bind/zones`.
On remarquera aussi que dans le répertoire `/etc/named/zones/master`, en plus du fichier de zone `db.msensa.ch.zone`, il y a un fichier supplémentaire `db.msensa.ch.zone.signed` qui contient la version signée de la zone. Il est plus gros que le fichier original, car tous les RR (*Resource Records*) sont maintenant signées.
Nous pouvons maintenant modifier le fichier `named.conf.local` pour pointer vers la zone `.signed` :

```
zone "msensa.ch." IN {
  ...
  file "/etc/bind/zones/master/db.msensa.ch.zone.signed";
};
```

Redémarrage du serveur pour tenir compte du fichier signé

```
$ sudo rndc reload
```

Contrôle de la zone msensa.ch avec DNSSec :

```
$ sudo dig @localhost DNSKEY msensa.ch +dnssec
```

Vous noterez que chaque enregistrement est accompagné d'un RR de type `RRSIG`, qui contient la signature du champ demandé. Après avoir signé notre zone, vous devez transmettre votre KSK publique à votre local Registrar (comme par exemple SWITCH pour les .ch), ils pourront ainsi ajouter un enregistrement de type DS au TLD.

Key rotation, zone maintenance

- Une fois la zone signées les champs RRSIG's ont une durée de vie de 30 jours. Après ces 30 jours, les signatures expirent et la zone n'est plus "valide". La seule manière de permettant de faire un "reset" de ce temporisateur de 30 jours est de resigner la zone.
- Chaque fois qu'une entrée est ajoutée, modifiée ou supprimée de la zone, il faut re-signer la zone en regénérant la version ".signed"
- Vous devez regénérer vos clés KSK et ZSK dans le temps. Il est recommandé de regénérer la KSK chaque année et la ZSK tous les 3 mois.

Activation du contrôle des signatures DNSSEC

Afin que votre serveur valide les résolutions de nom avec le DNSSEC, il est important d'activer ce contrôle en modifiant le fichier `/etc/bind/named.conf.options` en y ajoutant les deux options suivantes :

```
...
        dnssec-validation auto;
        dnssec-enable yes;
...
```

En mettant la validation en mode `auto`, votre serveur récupérera toutes les clés des différentes zones. Si vous spécifiez `yes` au lieu d'`auto`, il faudra spécifier dans ce même fichier la `trusted-key` correspondant à la clé publique de la racine.

Affichage des zones signées

BIND enregistre les zones *slave* signées en format *texte* comme nous l'avons spécifié dans le fichier *named.conf.options*. Si rien n'est précisé, le format sera de type *raw*. C'est le format dans lequel les données de zones sont stockées en mémoire et c'est ainsi plus rapide de charger/enregistrer cette zone sans en modifier le format.
Pour afficher le contenu d'une zone *slave* signée, il suffit d'utiliser la commande suivante :

```
$ sudo named-checkzone -D -f text itsec-lab.ch db.itsec-lab.ch.zone.signed
```

A.3.1 Test de la zone

L'outil drill est très pratique pour le test de la configuration DNSSec. Pour ceci, il suffit de l'installer sur la machine Linux :

```
$ sudo apt-get install ldnsutils
```

Afin qu'il puisse utiliser les clés des serveurs racines pour effectuer le contrôle de la chaîne de confiance, il faut installer celles-ci :

```
$ sudo dig +nocomments +nostats +nocmd +noquestion -t DNSKEY . > trusted-key.key
```

Pour tester une zone en particulier, la commande est la suivante :

```
$ sudo dig +trusted-key=~/trusted-key.key +topdown +sigchase +multiline <domain>
```

ou avec la commande dig :

```
$ sudo dig +topdown +sigchase +multiline +trusted-key=trusted-key.key -t A www.ripe.net.
```

ou avec la commande drill :

```
$ sudo drill -k trusted-key.key -V 5 -TD www.ietf.org
```

A.4 Configuration et activation d'un site web

Cette annexe décrit le paramétrage du logiciel **apache2** [1] afin que votre machine attaquante puisse simuler de faux site web. Dans cette annexe, nous allons instancier un serveur web pour qu'il retourne une fausse page web pour l'URL `www.heia-fr.ch`.

A.4.1 Installation

La première étape consiste à installer apache2 sur la machine Kali, si ce logiciel n'est pas encore présent et de le démarrer avec la configuration initiale.

```
$ sudo apt-get install apache2
$ sudo service apache2 start
```

En ouvrant un navigateur et en tapant l'url avec l'IP de votre machine, vous devriez obtenir la page suivante :

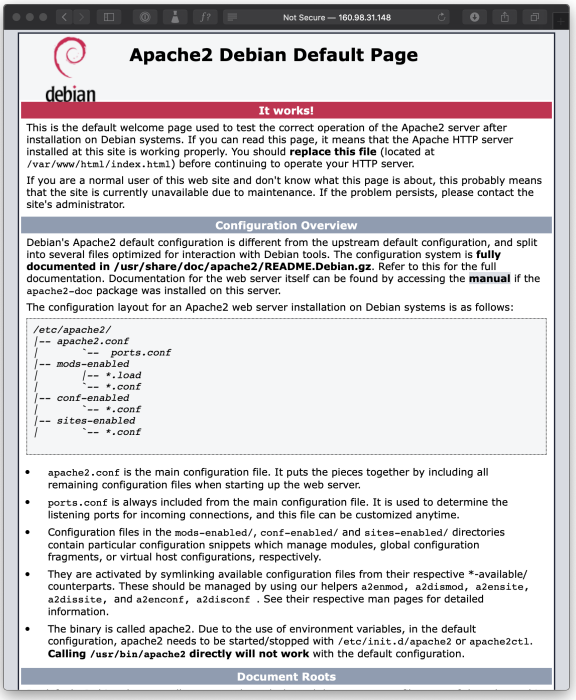

Figure 5 – *Site web par défaut*

A.4.2 Virtual Hosts

Pour instancier des *virtuals hosts* [2] et ainsi héberger plusieurs "faux" sites sur la machine de notre attaquant, il est nécessaire d'effectuer les configuration suivantes :

Copie du fichier de config

```
$ cd /etc/apache2/sites-available
$ sudo cp 000-default.conf heia-fr.conf
```

1. http://httpd.apache.org
2. https://httpd.apache.org/docs/2.4/fr/vhosts/

Edition du fichier de configuration
Editez les deux paramètres suivants dans le nouveau fichier `heia-fr.conf` :

```
<VirtualHost *:80>
ServerName www.heia-fr.ch
DocumentRoot /var/www/html/heia-fr
</VirtualHost>
```

Création d'une "fausse" page
Nous allons stocker la "fausse" page web dans le répertoire mentionné dans la configuration ci-dessus. À vous de produire le contenu de la page ainsi créée (index.html).

```
$ sudo mkdir /var/www/html/heia-fr
$ sudo vi /var/www/html/heia-fr/index.html
```

Activation du nouveau site web
Pour cela nous devons créer un lien symbolique :

```
$ sudo cd /etc/apache2
$ sudo ln -s ./sites-available/heia-fr.conf ./sites-enabled/heia-fr.conf
```

Redémarrage du server
Ensuite pour prendre en compte ces nouvelles configurations, il est nécessaire de redémarrer le serveur apache :

```
$ sudo service apache2 restart
```

A.4.3 Test du fonctionnement

Pour tester si votre site web est correctement configuré, le plus simple est d'utiliser l'outil Postman [1] qui permet de forger une requête HTTP et de spécifier l'attribut **Host** de l'entête HTTP. Dans l'exemple ci-dessous, le serveur du hacker à l'adresse 160.98.31.148 et nous avons ajouter l'attribut Host avec la valeur "www.heia-fr.ch" dans l'onglet Headers.

Sur la droite de l'interface vous devriez obtenir le contenu de la "fausse" page que vous avez configuré.

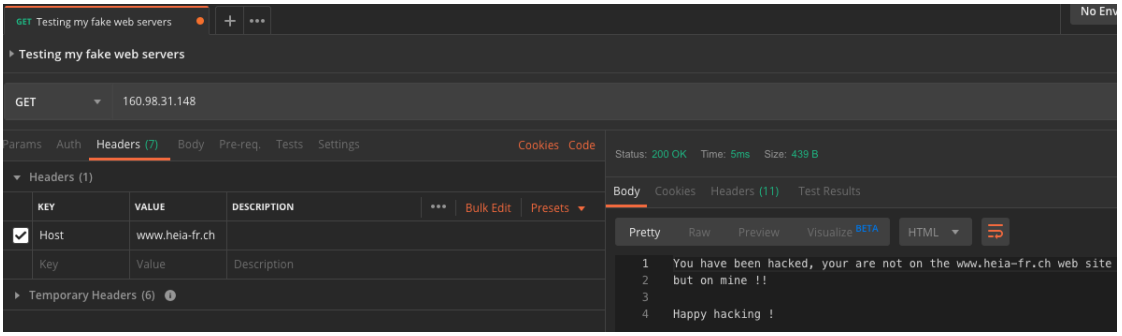

Figure 6 – *Interface du logiciel Postman - Résultat de la requête*

1. https://www.getpostman.com

Références

[1] bettercap hacking tool. https://www.bettercap.org/intro/.

[2] Cloud provider (compute and storage ressource) for swiss researchers and lectures. https://www.switch.ch/fr/engines/.

[3] Collection of tools for audit and pentesting, like arpspoof, dnsspoof. http://www.metasploit.com/, 2019.

[4] D. Atkins and R. Austein. Threat Analysis of the Domain Name System (DNS). RFC 3833, IETF, August 2004.

[5] Philippe Biondi. Scapy. http://www.secdev.org/projects/scapy/, 2013.

[6] Alan Clegg. Deploying dnssec, using bind 9.7. http://www.nanog.org/meetings/nanog50/presentations/Sunday/NANOG50.Talk6.NANOG-50-Clegg.pdf, 2010.

[7] Laurent Constantin. Netwox help to find and solve network problems. http://sourceforge.net/projects/ntwox/, 2014.

[8] Debian. Installation et configuration d'un serveur dns sur debian. http://www.admin-debian.com/scripts-shell/installation-et-configuration-serveur-dns-linux-debian/, 2010.

[9] Mathias Geniar. Implementing and maintaining dnssec on bind9 nameservers. http://mattiasgeniar.be/2010/07/12/implementing-maintaining-dnssec-on-bind9-nameservers/, 2014.

[10] P. Hallam-Baker and R. Stradling. DNS Certification Authority Authorization (CAA) Resource Record. RFC 6844, IETF, January 2013.

[11] P. Hoffman and J. Schlyter. The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol : TLSA. RFC 6698, IETF, August 2012.

[12] Offensive Security. Linux distribution used for penetration testing. http://www.kali.org, 2015.

[13] Dug Song. Collection of tools for audit and pentesting, like arpspoof, dnsspoof. http://monkey.org/~dugsong/dsniff/, 2014.

[14] Spiceworks. Deploy primary and secondary dns server. http://community.spiceworks.com/education/projects, 2014.

[15] ISC Team. Bind dns server from isc. http://www.isc.org.