BENIN DNS FORUM

DNSathon

LIVRE BLANC

Hackathon sur le DNS

Bénin DNS Forum

Cotonou, Bénin - Avril 2021

Hackathon sur le DNS

Bénin DNS Forum

SOMMAIRE

QUELQUES SPONSORS

ORGANISATION INTERNATIONALE DE la francophonie

Le Bénin DNS Forum, c'est avec le soutien de l'**Organisation Internationale de la Francophonie**

et

JENY SAS FOURNISSEUR D'ACCÈS INTERNET ET REGISTRE DU .BJ

ICANN

Arcep BENIN

Internet Society

Internet Society Chapitre du Benin

ASSI AGENCE DES SERVICES ET SYSTÈMES D'INFORMATION PRÉSIDENCE DE LA RÉPUBLIQUE DU BÉNIN

ADN AGENCE POUR LE DÉVELOPPEMENT DU NUMÉRIQUE PRÉSIDENCE DE LA RÉPUBLIQUE DU BÉNIN

Afilias

LIGHT INNOVATION

SKYRAN-GROUP

CONTEXTE

Le Système des Noms de Domaine, en anglais DNS (Domain Name System) est une ressource critique pour le fonctionnement d'Internet. C'est lui qui permet d'établir la correspondance entre un nom de domaine et une adresse IP. Pour cela, plusieurs serveurs dans une architecture de services distribués dans le monde entier assurent ce rôle. L'architecture est organisée en une hiérarchie et des mécanismes de résilience permettent d'assurer une très haute disponibilité de l'ensemble du système.

En plus d'être critique, le DNS est également un système extrêmement sensible. Pourtant, il est très peu connu et les compétences techniques pratiques sont relativement rares, même dans le milieu professionnel de l'informatique. Le but du présent livre blanc et du hackathon dont il est issu (DNSathon, une activité du Bénin DNS Forum) est de démystifier le DNS, de permettre aux jeunes étudiants et aux ingénieurs déjà en activité professionnelle de mieux comprendre ce système et de se doter de compétences pratiques effectives pour le gérer.

A cet effet, le hackathon est destiné à la réalisation d'un prototype d'infrastructure DNS de bout en bout (Root, Registre, Registrar, Résolveur). Il doit être collaboratif car dans la pratique, la gestion du système est véritablement collaborative et coopérative.

Dénommé DNSathon (hackathon sur le DNS), il réunit plusieurs compétences sur une journée et demie de travail collaboratif, configurations, développement de robots applicatifs, tests, débogages intenses, rédaction de contenu, apprentissage, partage et innovation qui permettent à l'ensemble des participants de contribuer à la mise en place d'un prototype de l'Infrastructure DNS.

L'objectif principal du DNSathon est de construire un prototype d'architecture DNS semblable à celui d'Internet. L'atteinte de cet objectif nécessite de reproduire les principales composantes de l'écosystème DNS que sont :

- Le serveur racine (qui répond aux requêtes relatives aux noms de domaines de premier niveau),
- Le registre (bases de données contenant des informations sur les domaines de premier et de second niveau),
- Le registrar (ou registraire, c'est l'entité auprès de laquelle se rend un client pour enregistrer un nom de domaine),
- Le registrant (utilisateur qui enregistre un nom de domaine).

ORGANISATION STRUCTURELLE DES PARTICIPANTS

Pour la réalisation du hackathon, nous recommandons la création et la mise en place de six équipes de travail (groupes fonctionnels) pouvant être constituées comme suit :

Groupe de travail Réseau Interconnexion et Connectivité

Ce groupe de travail met en place le réseau IP du prototype et assure l'interconnexion entre les différentes composantes (root, registre, registrar, revolver) de l'infrastructure. Il assure également l'adressage IP.

Groupe de travail Gestion de la racine alternative Root

Cette équipe assure la mise en place de deux serveurs DNS racine (.) du prototype tout en assurant la résilience applicative, c'est-à-dire utiliser deux applications différentes pour fournir le service DNS faisant office de Root. Par exemple, utiliser Bind et PowerDNS. Ce Groupe de travail doit également développer une interface permettant d'enregistrer un domaine de premier niveau (Top Level Domain - TLD) dans la zone racine.

Groupe de travail Registre (domaine de premier niveau – TLD)

Le groupe Registre est constitué de deux équipes dont chacune déploie et administre un domaine de premier niveau. Chaque équipe met en place un domaine de premier niveau en installant et configurant le serveur DNS faisant autorité sur ce domaine de premier niveau.

Ce groupe doit également développer deux interfaces : une pour communiquer avec la racine et l'autre pour communiquer avec les registraires.

Groupe de travail Registraire et hébergement de contenus

Il est question ici d'installation d'un service front-end d'enregistrement des noms de domaines de second niveau auprès des domaines de niveau supérieur (premier niveau) créés précédemment. Il s'agit en fait d'une interface web d'enregistrement et/ou une API.

Dans ce groupe, sont également configurés les serveurs NS qui feront autorité sur les domaines de second niveau fraîchement enregistrés et l'installation d'un service à valeur ajoutée d'hébergement mutualisé.

Groupe Gestionnaire de Revolvers DNS

Ce groupe de travail est dédié à la mise en place de deux résolveurs DNS chargés de la résolution des noms de domaines en adresse IP pour le réseau.

Groupe Rédaction du Livre Blanc et communication

Ce groupe est destiné à travailler avec chacun des autres groupes pour la rédaction du Livre Blanc du DNSathon. L'ensemble des processus et opérations techniques entrant dans le cadre de la mise en œuvre du prototype de l'infrastructure DNS doivent être documentés avec le soutien de ce groupe de travail.

PRÉREQUIS

1. Présentation du Raspberry Pi

Sorti officiellement le 29 février 2012, le Raspberry Pi est développé par la Raspberry Pi Foundation de David Braben (programmeur britannique et créateur de jeux vidéo). Il s'agit d'un mini-ordinateur dont la taille est comparable à celle d'une carte de crédit. Il est actuellement décliné en trois modèles se différenciant par leurs composants et leur prix. Si l'idée de départ était de produire un outil à faible coût pour favoriser l'initiation à la programmation, le succès que connaît aujourd'hui le Raspberry Pi va largement au-delà de cet objectif initial, tant les projets ayant vu le jour grâce à ce petit boitier sont nombreux. Actuellement, le prix du Raspberry Pi varie de 40 à 80 mille francs CFA selon que l'on désire acheter la carte toute nue ou le kit complet comprenant le boitier, le cordon d'alimentation, la carte mémoire et d'autres accessoires. Pratique à manipuler de par sa taille et son faible poids, peu cher et facile à utiliser tout en préservant une très bonne performance de l'ensemble, le Raspberry Pi est très polyvalent et constitue un excellent choix pour déployer des services informatiques dans un petit réseau.

C'est lui qui servira de serveur pour la matérialisation de chaque niveau hiérarchique de l'arborescence DNS tout au long du DNSathon. Au total, nous utiliserons neuf (09) Raspberry Pi.

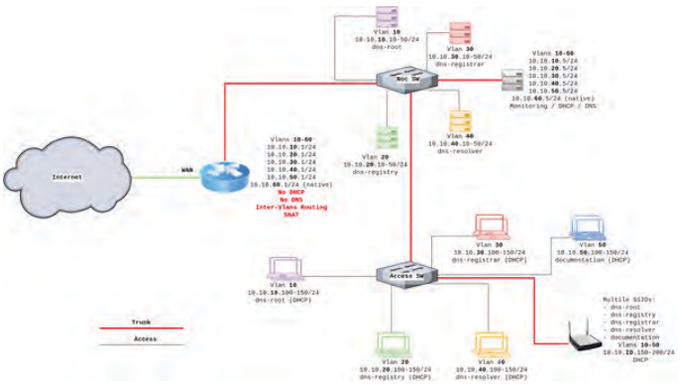
Figure 1 : Raspberry Pi 3 modèle B (carte mère et interfaces à gauche, boitier à droite)

Pour ce DNSathon, les Raspberry Pi utilisés sont de type 3 modèle B avec un système d'exploitation pré installé, Raspbian. Les caractéristiques de ce modèle sont les suivantes :

Désignation	Spécification
Dénomination commerciale	Raspberry Pi 3 Official Desktop Starter Kit
Marque	Raspberry Pi
Numéro du modèle de l'article	RPi3_OffStrKit
Couleur	White
Système d'exploitation	Linux (Raspbian)
Plateforme du matériel informatique	Linux
Marque du processeur	Broadcom
Type de processeur	ARM710
Vitesse du processeur	1200 MHz
Nombre de cœurs	4
Taille de la mémoire vive	1 GB
Interface du disque dur	USB
Disque	16GB Class 10 MicroSD

Tableau 1 : caractéristiques du Raspberry Pi utilisé pour le DNSathon

2. Architecture réseau



Après les préliminaires de répartition des participants en groupe et la distribution du matériel, nous passons à la conception de l'infrastructure réseau.

Les paramètres de configuration utilisés lors du DNSathon sont consignés dans le tableau ci-dessous suivi de la procédure de configuration détaillée pour chaque Raspberry.

3. Choix des solutions applicatives

Le DNSathon consiste en la mise en place d'une infrastructure complète DNS de bout en bout. Pour ce faire, nous allons implémenter :

- notre propre racine DNS (root server),
- nos propres serveurs DNS de premier niveau appelés Registre (Registry en Anglais),
- nos propres bureaux d'enregistrement encore appelés Registrar (en anglais),
- nos propres serveurs DNS de second niveau
- et nos propres resolvers.

Plusieurs solutions existent aujourd'hui pour la mise en place d'un serveur DNS. Pour ce DNSathon, nous avons fait le choix de Bind, PowerDNS et Unbound respectivement pour la mise en place du serveur DNS Racine, des registres (Registre-1 et Registre-2) et des Resolvers (Resolver-1 et Resolver-2) des fournisseurs d'accès (ISP). Le choix d'une diversité de solutions réside dans le fait qu'Internet même est bâti sur une diversité de solutions qui fonctionnent de concert pour délivrer le service de résolution de noms partout sur Internet depuis n'importe où dans le monde. Nous avons donc voulu être le plus près possible de la réalité.

Le choix de PowerDNS réside essentiellement dans le fait qu'il embarque un API HTTP qui sera d'une grande utilité dans la mise en place des bureaux d'enregistrement Registrar-1 et Registrar-2. Plus d'informations sur ces deux utilitaires sont disponibles aux adresses :

https://www.isc.org/downloads/bind/ et https://www.powerdns.com/ .

Quelques autres utilitaires auxquels nous avons fait également recours sont abordés ci- dessous :

01. l'utilitaire en ligne zonefile.org (http://www.zonefile.org/) permet de générer automatiquement le contenu du fichier de zone d'un domaine sur la base d'informations fournies en entrée ;

02. PowerAdmin est un utilitaire d'administration de serveur DNS via interface web écrit en PHP. Il est parfaitement intégrable à PowerDNS et simplifie l'administration du serveur basé sur PowerDNS, celui-ci ne supportant nativement que la ligne de commande. Plus de détails sont disponibles sur http://www.poweradmin.org/;

03. MySQL : utilitaire pour l'administration de bases de données.

04. Apache : utilitaire pour la mise en place d'un serveur web, nécessaire pour utiliser PowerAdmin.

4. Configuration de base des Raspberry Pi

Chaque équipe reçoit un Raspberry, le monte avec ses périphériques (souris, clavier et écran), puis le branche au secteur électrique pour effectuer la configuration de base. Dès que cette configuration est achevée, testée et confirmée, un membre de l'équipe débranche les périphériques du Raspberry Pi et le connecte au commutateur dans son port approprié (grâce à un câble Ethernet) préalablement assigné par le groupe Interconnexion et connectivité. A partir de cet instant, le Raspeberry Pi est un serveur accessible en SSH sur le réseau.

Pour mettre en marche le Raspberry, il est essentiel de disposer d'un écran avec entrée HDMI, d'un clavier ainsi que d'une souris. Il faut ensuite brancher le câble HDMI, la souris et le clavier au Raspberry en utilisant le port HDMI et les ports USB. Une fois branché, le nano ordinateur s'initialise tout seul (le modèle de Raspberry Pi acheté est livré avec le système d'exploitation Raspbian), et affiche directement son écran d'accueil.

Ensuite, il faut ouvrir un terminal en passant par le menu du Raspberry et saisir les commandes consignées dans le tableau ci-dessous pour la configuration.

ACTIONS	COMMANDES CORRESPONDANTES
Configurer le clavier (changer la langue du clavier de l'anglais vers le français)	**$ setxkbmap fr**
Démarrer le service SSH	**$ sudo /etc/init.d ssh start**
Configurer la carte réseau Ethernet pour la connectivité du Raspberry Pi au réseau : les valeurs assignées à Address, Netmask, Network et Gateway dans l'exemple ci-contre sont mises à titre indicatif et devraient varier d'un groupe à un autre en fonction du plan d'adressage retenu pour votre réseau.	**$ sudo nano /etc/network/interfaces** **Rechercher et remplacer la ligne** **$ iface eth0 inet dhcp** **par** **$ iface eth0 inet static** **Address 10.10.10.10** **Net mask 255.255.255.0** **Network 10.10.10.0** **Gateway 10.10.10.1** **Enregistrer les modifications et fermer le fichier**
Redémarrer le service réseau	**$ sudo service networking restart** **ou** **$ sudo /etc/init.d/Networking restart**
Modifier le nom du serveur	**$vi /etc/hostname**
Modifier le mot de passe du user root	**$ sudo passwd root**
Tester la connexion ssh depuis un autre ordinateur dans le réseau	**$ ssh login@IP**

Tableau 2 : Actions et commandes nécessaires à la configuration de base du Raspberry

RESTITUTION DES TRAVAUX PAR ÉQUIPE

Pour des soucis de présentation, plusieurs configurations sont détaillées en annexes. Aussi, l'ensemble des configurations sont disponibles en ligne sur le compte Github : https://github.com/AlfredArouna/DNSathon.

1. Groupe de travail Réseau Interconnexion et Connectivité

NOM DE L'ÉQUIPE	Réseau Interconnexion et Connectivité
THEMATIQUE/OBJET DE TRAVAIL	Mettre en place l'infrastructure physique, les différents sous-réseaux et leur interconnexion pour permettre la communication entre tous les serveurs (Root, TLD, Resolver, ...) et les équipes du DNSathon qui effectuent les configurations.
ETAPE SUIVIS/ PROCESSUS	Suivant l'architecture proposée, nous avons suivi les étapes suivantes : 1) Configuration du routeur Cisco 1800 (détails en annexe « configuration détaillée du routeur ») 2) Configuration switch 2.1- Switch Core (détails en annexe « configuration détaillée du Core Switch ») 2.2- Access switch (détails en annexe « configuration détaillée du Access Switch »)
RÉSULTATS	➔ Tous les groupes de travail sont interconnectés correctement avec un accès à internet disponible. ➔ Tous les groupes arrivent à faire des « ping » vers les RaspBerry Pi et parviennent à se connecter en SSH à ces derniers.
ÉQUIPEMENTS UTILISÉS	1 router, 2 switchs, 9 Raspberry Pi, plusieurs câbles réseaux
RÉSUMÉ DES RÉSULTATS	En résumé le travail n'a pas été facile à cause des problèmes rencontrés sur certains équipements mais cela a été pour nous les participants une bonne occasion de travailler en groupe et de gagner beaucoup d'autres notions en matière de configuration.
DIFFICULTÉS RENCONTRÉES (BIEN VOULOIR MENTIONNER LA DIFFICULTÉ ET LA SOLUTION TROUVÉE)	**Problèmes : 1**- manque de câble pour l'interconnexion des utilisateurs ; 2- certains utilisateurs ne parviennent pas à joindre une partie du réseau. **Solutions : 1**- fourniture de câble par les participants ; 2- investiguer et faire des tests jusqu'à résolution du problème.

2. Groupe de travail gestion de la racine alternative Root

La Racine du DNS est un serveur qui répond aux requêtes concernant les domaines de premier niveau et qui les redirige vers le serveur DNS de premier niveau approprié. La mise en place du serveur DNS racine est une étape primordiale de ce DNSathon parce que la racine est le serveur qui reçoit les requêtes des resolvers et donne l'adresse IP du serveur DNS de domaine de premier niveau (TLD) de la ressource demandée par un client et relayée par le resolver. Dans le cadre de ce DNSathon, c'est donc ce nouveau « Root » qui redirigera les requêtes DNS vers les adresses IP des

registres .cotonou et .benin.

NOM DE L'ÉQUIPE	RACINE ROOT
THEMATIQUE/OBJET DE TRAVAIL	Mettre en place une Racine alternative avec son propre fichier Root hints dédié.

ETAPE SUIVIS/ PROCESSUS	→ **Modifier le nom du Raspberri Pi**
	$ sudo vi /etc/hostname
	→ **Assigner une adresse IP statique au RASPBERRY**
	$ sudo nano /etc/network/interfaces
	Address 10.10.10.10
	Net mask 255.255.255.0
	Network 10.10.10.0
	Gateway 10.10.10.1
	→ **Appliquer les nouveaux paramètres réseaux**
	$ sudo /etc/init.d/Networking restart
	→ **Procéder à l'installation de bind9 :**
	$ sudo apt-get install bind9 bind9utils dnsutils
	→ **Créer le fichier de zone racine db.root (détails en annexe « contenu du fichier de zone racine db.root »).**
	root@root:/etc/bind# vi db.root
	→ **Créer le fichier de zone inverse racine db.10.10.10 (détails en annexe « contenu du fichier de zone inverse racine db.10.10.10 »).**
	root@root:/etc/bind# vi db.10.10.10
	→ **Déclarer à Bind les zones créées afin qu'elles soient considérées (détails en annexe « contenu du fichier listant les zones par défaut sur le serveur (named.conf.default-zones) »).**
	root@root:/etc/bind# vi named.conf.default-zones
	→ **Créer le fichier de root hints customisé (détails en annexe « contenu du fichier root hints customisé »).**
	root@root:/etc/bind# vi db.root.orig

RÉSULTATS	→ Les utilisateurs sont créés sur le système, → Bind9 est installé. → Les fichiers suivants sont configurés avec succès : fichier de zone racine, fichier de zone inverse racine, fichier des zones par défaut, fichier de root hints. → La configuration de Bind est contrôlée avec succès. → Le service Bind est redémarré avec succès.
ÉQUIPEMENTS UTILISÉS	→ RASPBERRY PI 3 → PC/ORDINATEUR → INTERNET/CONNECTION
RÉSUMÉ DES RÉSULTATS	La réalisation du serveur racine DNS était primordiale parce que la racine est un serveur DNS qui répond aux requêtes sur les noms de domaine de premier niveau.
DIFFICULTÉS RENCONTRÉES (BIEN VOULOIR MENTIONNER LA DIFFICULTÉ ET LA SOLUTION TROUVÉE)	**Difficulté :** Potentiel risque d'erreur de configurations **Solution :** concentration et utilisation des commandes de vérification de configuration de Bind telle que : → named-checkconf /etc/named.conf → named-checkconf -t /var/named/chroot /etc/named.conf Il ne devrait avoir aucun message à l'écran après l'exécution due la commande. $ named-checkzone . /etc/bind/db.root Si tout est correct, on devrait avoir un résultat semblable au message suivant à l'écran. zone ./IN: loaded serial 2018101201 OK

3. Groupe Registre de TLD

NOM DE L'ÉQUIPE	Registre de TLD 1	Registre de TLD 2
THEMATIQUE/OBJET DE TRAVAIL	Mettre en place un TLD en installant 2 serveurs DNS faisant autorité, et à développer ensuite deux interfaces : une pour communiquer avec la racine root et l'autre pour communiquer avec les registrar.	
	TLD1 : .cotonou	TLD2 : .benin
ETAPE SUIVIS/ PROCESSUS	Les étapes suivies pour la mise en place d'un TLD sont les suivantes: ➔ Installation des modules nécessaires : php, mysql, powerdns, poweradmin (détails en annexe « Installation des modules nécessaires pour la mise en en place d'un tld »). 01. # apt-get update -y 02. # apt-get install apache2 -y #install apache 03. # apt-get install php -y 04. # apt-get install mysql-server mysql-client 05. # apt-get install pdns-server pdns-backend-mysql ➔ Configuration nécessaire 01. Création de la base de donnée pour powerdns (détails en annexe « contenu du fichier de zone racine db.root »). 02. Configuration de Poweradmin 03. Ajout des zones avec Poweradmin	
RÉSULTATS	La mise en place d'un TLD installé, configuré et opérationnel	
ÉQUIPEMENTS UTILISÉS	Raspberry pi 3 Model B, câble réseau, ordinateurs	
RÉSUMÉ DES RÉSULTATS	A la fin des configurations, les tld .cotonou et .benin ont été bien mis en place avec la possibilité d'ajouter des zones pour l'enregistrement.	
DIFFICULTÉS RENCONTRÉES (BIEN VOULOIR MENTIONNER LA DIFFICULTÉ ET LA SOLUTION TROUVÉE)	➔ instabilité de la connexion durant le travail ➔ souci d'encodage de la base de données lors de la configuration de powerdns	

4. Groupe Registrar et hébergement de contenus

NOM DE L'ÉQUIPE	Groupe Registrar et hébergement de contenus
THEMATIQUE/OBJET DE TRAVAIL	Mettre en place un service d'enregistrement des noms de domaines pour les TLD .cotonou et .benin grâce à une interface web d'enregistrement et/ou une API. Le groupe mettra également en place les serveurs NS qui feront autorité sur les domaines enregistrés et mettra en place un service à valeur ajoutée d'hébergement mutualisé.
ETAPE SUIVIS/ PROCESSUS	**Développement** Nous avons utilisé un template. Ce template a été géré grâce à HTML5, CSS3 avec le Framework Bootstrap. Javascript, jquery et Php ont permis de gérer la partie dynamique. Nous avons ensuite procédé à l'installation et configuration du serveur: (apache2, Php 7.1). Pour plus d'information sur les procédure d'installation: http://www.heidislab.com/tutorials/installing-php-7-1-on-raspbian-stretch-raspberry-pi-zero-w Nous avons également installé phpmyadmin et mysql server en suivant ce lien https://www.stewright.me/2012/09/tutorial-install-phpmyadmin-on-your-raspberry-pi/

ÉQUIPEMENTS UTILISÉS	Raspberry Pi et des ordinateurs
RÉSUMÉ DES RÉSULTATS	Mise en place d'un prototype de bureau d'enregistrement de nom de domaine (Registrar)
DIFFICULTÉS RENCONTRÉES (BIEN VOULOIR MENTIONNER LA DIFFICULTÉ ET LA SOLUTION TROUVÉE)	

5. Groupe Gestionnaire de Revolvers DNS

Ce groupe de travail met en place deux resolvers DNS public (en utilisant Unbound) qui devront faire de la validation DNSSEC et répondre en IPv6. Le resolver DNS sera configuré pour activer la fonction de caching qui permet au resolver de garder en mémoire pendant un temps les réponses qu'il a obtenues. De nouvelles interrogations ne seront donc pas envoyées vers des serveurs extérieurs lorsque le resolver recevra une requête dont il possède déjà la réponse. La fonction de cache DNS permet ainsi d'accélérer les réponses au niveau d'un resolver.

NOM DE L'ÉQUIPE	**Groupe Gestionnaire de Revolver DNS**
THEMATIQUE/OBJET DE TRAVAIL	Mise en place du resolver DNS
ETAPE SUIVIS/ PROCESSUS	→ Installer unbound → Editer le fichier de configuration → Renseigner le contenu du fichier /var/lib/root.hints → Redémarrer le service unbound → Tester la configuration effectuée avec la commande dig
RÉSULTATS	Résolution des correspondants noms de domaine/adresse IP provenant des clients voulant accéder à des ressources des extensions .cotonou et .benin de nos deux registres.
ÉQUIPEMENTS UTILISÉS	→ Raspberry → Laptop → Câbles Ethernet → Rallonges → Switch → Template → Resolver DNS (Unbound) → Serveur web Apache
RÉSUMÉ DES RÉSULTATS	le serveur resolver résout et met en cache les réponses pour notre hiérarchie DNS.
DIFFICULTÉS RENCONTRÉES (BIEN VOULOIR MENTIONNER LA DIFFICULTÉ ET LA SOLUTION TROUVÉE)	La majeure difficulté se trouve au niveau du test de vérification de la configuration avec la commande dig. Nous pouvons noter l'attente du fichier root.hints de la part du groupe se chargeant de ce fichier. Il y a également eu d'autres difficultés mineures mais qu'on a facilement pu résoudre avec Google

6. Groupe Rédaction du Livre Blanc et communication

NOM DE L'ÉQUIPE	**Groupe Gestionnaire de Revolver DNS**
THEMATIQUE/OBJET DE TRAVAIL	Communication et Rédaction du rapport final
ETAPE SUIVIS/ PROCESSUS	→ faire la communication sur les réseaux sociaux pendant l'atelier → Présenter le rapport final
RÉSULTATS	Téléphone, réseaux sociaux, réseau wifi, appareil photo, PC
ÉQUIPEMENTS UTILISÉS	→ Raspberry → Laptop → Câbles Ethernet → Rallonges → Switch → Template → Resolver DNS (Unbound) → Serveur web Apache
RÉSUMÉ DES RÉSULTATS	→ Réaliser des photos et faire du contenu approprié à chaque fois qu'on avance en les partageant sur Twitter et Facebook. → Concevoir un rapport détaillé sur l'atelier et sur chaque équipe puis suivre les étapes et les résultats de chaque équipe en les consignant à chaque fois.
DIFFICULTÉS RENCONTRÉES (BIEN VOULOIR MENTIONNER LA DIFFICULTÉ ET LA SOLUTION TROUVÉE)	Les équipes étaient très occupées, trouver du temps pour remplir les formulaires. Par conséquent, nous avons eu du mal à récolter les formulaires remplis. Nous avons dû patienter et surtout y ajouter de la compréhension pour que tout se déroule merveilleusement bien.

CONCLUSION

Internet évolue à une vitesse croissante et offre de plus en plus de services. Le Système des Noms de Domaine, une ressource hautement critique pour le fonctionnement de ce vaste réseau, mérite qu'une meilleure attention lui soit porté car tout ce que nous faisons sur Internet commence par le DNS et finit par le DNS. C'est pourquoi, au Bénin DNS Forum, nous avons opté pour le renforcement des capacités à travers plusieurs initiatives comme le DNSathon et la rédaction de ce Livre Blanc afin de permettre à plusieurs communautés de reproduire cette initiative et d'aller encore bien au delà à travers l'implémentation de DNSSEC, WHOIS, RDAP, DoH/DoT, et bien d'autres protocoles liés à l'écosystème du DNS.

ANNEXES

L'ensemble des configurations présentées en annexe sont également disponibles en ligne sur le compte Github :
https://github.com/AlfredArouna/DNSathon

1. Configuration détaillée du routeur

```
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
hostname RHACKATON
!
no ip domain lookup
no ipv6 cef
multilink bundle-name authenticated
crypto pki token default removal timeout 0
license udi pid CISCO1841 sn FCZ1404C0UC
redundancy
!
interface FastEthernet0/0
 description "TO INTERNET"
 ip address 164.160.143.68 255.255.255.192
 ip nat outside
 ip virtual-reassembly in
 duplex auto
 speed auto
!
interface FastEthernet0/1
 description "TO LAN"
 no ip address
 ip nat inside
 ip virtual-reassembly in
 duplex auto
 speed auto
!
interface FastEthernet0/1.10
 description "Gateway for VLAN 10"
 encapsulation dot1Q 10
 ip address 10.10.10.1 255.255.255.0
 ip nat inside
 ip virtual-reassembly in
!
interface FastEthernet0/1.20
 description "Gateway for VLAN 20"
 encapsulation dot1Q 20
 ip address 10.10.20.1 255.255.255.0
 ip nat inside
 ip virtual-reassembly in
!
interface FastEthernet0/1.30
 description "Gateway for VLAN 30"
 encapsulation dot1Q 30
 ip address 10.10.30.1 255.255.255.0
 ip nat inside
 ip virtual-reassembly in
!
interface FastEthernet0/1.40
 description "Gateway for VLAN 40"
 encapsulation dot1Q 40
 ip address 10.10.40.1 255.255.255.0
 ip nat inside
 ip virtual-reassembly in
!
interface FastEthernet0/1.50
 description "Gateway for VLAN 50"
 encapsulation dot1Q 50
 ip address 10.10.50.1 255.255.255.0
 ip nat inside
 ip virtual-reassembly in
!
interface FastEthernet0/1.60
 description "Gateway for VLAN 60"
 encapsulation dot1Q 60 native
 ip address 10.10.60.1 255.255.255.0
 ip nat inside
 ip virtual-reassembly in
!
ip default-gateway 164.160.143.126
ip forward-protocol nd
!
ip nat pool SNAT-10 10.10.10.0 10.10.10.255 netmask
255.255.255.0
ip nat pool SNAT-20 10.10.20.0 10.10.20.255 netmask
255.255.255.0
ip nat pool SNAT-30 10.10.30.0 10.10.30.255 netmask
255.255.255.0
ip nat pool SNAT-40 10.10.40.0 10.10.40.255 netmask
255.255.255.0
ip nat pool SNAT-50 10.10.50.0 10.10.50.255 netmask
255.255.255.0
ip nat pool SNAT-60 10.10.60.0 10.10.60.255 netmask
255.255.255.0
ip nat inside source list 10 interface FastEthernet0/0 overload
ip nat inside source list 20 interface FastEthernet0/0 overload
ip nat inside source list 30 interface FastEthernet0/0 overload
ip nat inside source list 40 interface FastEthernet0/0 overload
ip nat inside source list 50 interface FastEthernet0/0 overload
ip nat inside source list 60 interface FastEthernet0/0 overload
ip route 0.0.0.0 0.0.0.0 FastEthernet0/0
!
logging esm config
access-list 10 permit 10.10.10.0 0.0.0.255
access-list 20 permit 10.10.20.0 0.0.0.255
access-list 30 permit 10.10.30.0 0.0.0.255
access-list 40 permit 10.10.40.0 0.0.0.255
access-list 50 permit 10.10.50.0 0.0.0.255
access-list 60 permit 10.10.60.0 0.0.0.255
!
snmp-server community public RW
snmp-server contact Dnsathon
snmp-server enable traps snmp linkdown linkup coldstart
warmstart
snmp-server host 10.10.60.5 version 2c public
!
control-plane
!
line con 0
 password DNSathon
line aux 0
line vty 0 4
 password DNSathon
 login
 transport input all
line vty 5 15
```

```
 password DNSathon
 login
 transport input all
!
scheduler allocate 20000 1000
end
```

2. Configuration détaillée du Core Switch

```
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname switch-core
!
enable password DNSathon
!
system mtu routing 1500
!
spanning-tree mode pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
!
interface FastEthernet0/1
 switchport trunk native vlan 60
 switchport mode trunk
!
interface FastEthernet0/2
 switchport access vlan 10
 switchport mode access
!
interface FastEthernet0/3
 description root-1
 switchport access vlan 10
 switchport mode access
!
interface FastEthernet0/4
 description root-2
 switchport access vlan 10
 switchport mode access
!
interface FastEthernet0/5
 description registry-1
 switchport access vlan 20
 switchport mode access
!
interface FastEthernet0/6
 description registry-2
 switchport access vlan 20
 switchport mode access
!
interface FastEthernet0/7
 description registrar-1
 switchport access vlan 30
 switchport mode access
!
interface FastEthernet0/8
 description registrar-2
 switchport access vlan 30
 switchport mode access
!
interface FastEthernet0/9
 description resolver-1
 switchport access vlan 40
 switchport mode access
!
interface FastEthernet0/10
 description resolver-2
 switchport access vlan 40
 switchport mode access
!
interface FastEthernet0/11
 description comm-1
 switchport access vlan 50
 switchport mode access
!
interface FastEthernet0/12
 description comm-2
 switchport access vlan 50
 switchport mode access
!
interface FastEthernet0/13
 description Cascade_SW_Access
 switchport trunk native vlan 60
 switchport mode trunk
!
interface FastEthernet0/23
 switchport trunk native vlan 60
 switchport mode trunk
!
interface FastEthernet0/24
 switchport trunk native vlan 60
 switchport mode trunk
!
interface Vlan1
 no ip address
!
interface Vlan10
 ip address 10.10.10.2 255.255.255.0
!
interface Vlan20
 ip address 10.10.20.2 255.255.255.0
!
interface Vlan30
 ip address 10.10.30.2 255.255.255.0
!
interface Vlan40
 ip address 10.10.40.2 255.255.255.0
!
interface Vlan50
 ip address 10.10.50.2 255.255.255.0
!
interface Vlan60
 ip address 10.10.60.20 255.255.255.0
!
```

```
ip http server
ip http secure-server
snmp-server community public RW
snmp-server contact Dnsathon
snmp-server enable traps snmp linkdown linkup coldstart warmstart
snmp-server host 10.10.60.5 version 2c public
!
!
line con 0
line vty 0 4
 password DNSathon
 login
line vty 5 15
 password DNSathon
 login
!
end
```

3. Configuration détaillée du Access Switch

```
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname switch-core
!
enable password cisco
!
system mtu routing 1500
!
spanning-tree mode pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
!
interface FastEthernet0/1
 switchport access vlan 10
 switchport mode access
interface FastEthernet0/2
 switchport access vlan 10
 switchport mode access
interface FastEthernet0/3
 switchport access vlan 10
 switchport mode access
interface FastEthernet0/4
 switchport access vlan 10
 switchport mode access
interface FastEthernet0/5
 switchport access vlan 20
 switchport mode access
interface FastEthernet0/6
 switchport access vlan 20
 switchport mode access
interface FastEthernet0/7
 switchport access vlan 20
 switchport mode access
interface FastEthernet0/8
 switchport access vlan 20
 switchport mode access
interface FastEthernet0/9
 switchport access vlan 30
 switchport mode access
interface FastEthernet0/10
 switchport access vlan 30
 switchport mode access
interface FastEthernet0/11
 switchport access vlan 30
 switchport mode access
interface FastEthernet0/12
 switchport access vlan 30
 switchport mode access
interface FastEthernet0/13
 description Cascade_SW_Core
 switchport trunk native vlan 60
 switchport mode trunk
interface FastEthernet0/14
 switchport access vlan 40
 switchport mode access
interface FastEthernet0/15
 switchport access vlan 40
 switchport mode access
interface FastEthernet0/16
 switchport access vlan 40
 switchport mode access
interface FastEthernet0/17
 switchport access vlan 40
 switchport mode access
interface FastEthernet0/18
 switchport access vlan 50
 switchport mode access
interface FastEthernet0/19
 switchport access vlan 50
 switchport mode access
interface FastEthernet0/20
 switchport access vlan 50
 switchport mode access
interface FastEthernet0/21
 switchport access vlan 50
 switchport mode access
interface FastEthernet0/22
 switchport access vlan 60
 switchport mode access
interface FastEthernet0/23
 switchport access vlan 60
 switchport mode access
interface FastEthernet0/24
 switchport access vlan 60
 switchport mode access
!
interface Vlan10
 ip address 10.10.10.3 255.255.255.0
 no ip route-cache
 shutdown
interface Vlan20
 ip address 10.10.20.3 255.255.255.0
 no ip route-cache
 shutdown
```

```
interface Vlan30
 ip address 10.10.30.3 255.255.255.0
 no ip route-cache
 shutdown
interface Vlan40
 ip address 10.10.40.3 255.255.255.0
 no ip route-cache
 shutdown
interface Vlan50
 ip address 10.10.50.3 255.255.255.0
 no ip route-cache
 shutdown
interface Vlan60
 ip address 10.10.60.25 255.255.255.0
 no ip route-cache
!
ip http server
ip http secure-server
snmp-server community public RW
snmp-server contact Dnsathon
snmp-server enable traps snmp linkdown linkup coldstart warmstart
snmp-server host 10.10.60.5 version 2c public
!
!
line con 0
 password DNSathon
 login
line vty 0 4
 password DNSathon
 login
line vty 5 15
 password DNSathon
 login
!
```

4. Contenu du fichier de zone racine db.root

```
;
; BIND zone file for "."
;
$TTL    604800
.       IN      SOA     root. admin.root. (
                        2018101201      ; Serial
                                604800      ; Refresh
                                86400       ; Retry
                                2419200     ; Expire
                                604800 )    ; Negative Cache TTL
;
.       IN    NS        root.
.       IN    NS     racine.
root.   IN    A      10.10.10.10
racine. IN    A      10.10.10.10
.       IN    A      10.10.10.10
```

5. Contenu du fichier de zone inverse racine db.10.10.10

```
;
; BIND reverse file for 10.10.10.0/24
;
$TTL    604800
@       IN      SOA     root. admin.root. (
                              2018101201          ; Serial
                                   604800         ; Refresh
                                    86400         ; Retry
                                   2419200        ; Expire
                                   604800 )   ; Negative Cache TTL
;
@   IN   NS     root.
@   IN   NS     racine.
10  IN   PTR    root.
10  IN   PTR    racine.
```

6. Contenu du fichier listant les zones par défaut sur le serveur (named.conf.default-zones)

```
// prime the server with knowledge of the root servers
zone "." {
        type master;
        file "/etc/bind/db.root";
};
// be authoritative for the localhost forward and reverse zones, and
for
// broadcast zones as per RFC 1912

zone "localhost" {
        type master;
        file "/etc/bind/db.local";
};
zone "127.in-addr.arpa" {
        type master;
        file "/etc/bind/db.127";
};
zone "0.in-addr.arpa" {
        type master;
        file "/etc/bind/db.0";
};
zone "255.in-addr.arpa" {
        type master;
        file "/etc/bind/db.255";
};
zone "10.10.10.in-addr.arpa" {
        type master;
        file "/etc/bind/db.10.10.10";
};
```

7. Contenu du fichier root hints customisé

```
;       This file holds the information on root name servers needed to
;       initialize cache of Internet domain name servers
;       (e.g. reference this file in the "cache  .  <file>"
;       configuration file of BIND domain name servers).
;
;       This file is made available by InterNIC
;       under anonymous FTP as
;           file                /domain/named.cache
;           on server           FTP.INTERNIC.NET
;       -OR-                    RS.INTERNIC.NET
;
;       last update:    February 17, 2016
;       related version of root zone:   2016021701
;
; formerly NS.INTERNIC.NET
;
.            3600000      NS    root.
root.        3600000      A     10.10.10.10
;
.            3600000    NS              racine.
racine.      3600000                    A               10.10.10.10
; End of file
```

8. Installation des modules nécessaires pour la mise en en place d'un tld

9. Configuration du serveur autoritaire avec PDNS

Installation du paquet MySQL server et client avec la commande :

sudo apt-get install mysql-server mysql-client

Se connecter au serveur MySQL, ensuite saisir le mot de passe root du serveur MySQL avec la commande :

mysql -u root –p

Créer une base de données powerdns. Avec la commande :

CREATE DATABASE powerdns;

Créer un utilisateur de la base de données powerdns qui sera utilisé pour se connecter à la base de données. Avec la commande :

GRANT ALL ON powerdns.* TO 'power_admin'@'localhost' IDENTIFIED BY 'power_admin_password'; GRANT ALL ON powerdns.* TO 'power_admin'@'localhost.localdomain' IDENTIFIED BY 'power_admin_password'

FLUSH PRIVILEGES;

Créer les tables nécessaires à l'application PowerDNS:

USE powerdns;

```
CREATE TABLE domains (
id INT AUTO_INCREMENT, name VARCHAR(255) NOT NULL,
master VARCHAR(128) DEFAULT NULL, last_check  INT DEFAULT NULL, type VARCHAR(6) NOT NULL, notified_serial INT UNSIGNED DEFAULT NULL, account  VARCHAR(40) CHARACTER SET 'utf8' DEFAULT NULL, PRIMARY KEY (id)) Engine=InnoDB CHARACTER SET=utf8;

CREATE UNIQUE INDEX name_index ON domains(name);

CREATE TABLE records (id   BIGINT AUTO_INCREMENT,
domain_id   INT DEFAULT NULL, name   VARCHAR(255) DEFAULT NULL, type   VARCHAR(10) DEFAULT NULL, content VARCHAR(64000) DEFAULT NULL, ttl   INT DEFAULT NULL, prio  INT DEFAULT NULL, change_date  INT DEFAULT NULL,   disabled TINYINT(1) DEFAULT  0, ordername  VARCHAR(255) BINARY DEFAULT NULL, auth  TINYINT(1) DEFAULT 1,  PRIMARY KEY (id)) Engine=InnoDB CHARACTER SET=UTF8;

CREATE INDEX nametype_index ON records (name,type);

CREATE INDEX domain_id ON records (domain_id);

CREATE INDEX ordername ON records (ordername);

CREATE TABLE supermasters (ip   VARCHAR (64) NOT NULL, nameserver  VARCHAR (255) NOT NULL, account   VARCHAR(40) CHARACTER SET 'utf8' NOT NULL, PRIMARY KEY (ip, nameserver)) Engine=InnoDB CHARACTER SET =UTF8;

CREATE TABLE comments (id  INT AUTO_INCREMENT, domain_id INT NOT NULL, name  VARCHAR(255) NOT NULL,  type VARCHAR(10) NOT NULL, modified_at  INT NOT NULL, account VARCHAR(40) CHARACTER SET 'utf8' DEFAULT NULL, comment TEXT CHARACTER SET 'utf8' NOT NULL, PRIMARY KEY (id)) Engine=InnoDB CHARACTER SET=UTF8;

CREATE INDEX comments_name_type_idx ON comments (name, type);

CREATE INDEX comments_order_idx ON comments (domain_id, modified_at);

CREATE TABLE domainmetadata (id   INT AUTO_INCREMENT, domain_id  INT NOT NULL, kind   VARCHAR(32), content
```

TEXT, PRIMARY KEY (id)) Engine=InnoDB CHARACTER SET=UTF8;

CREATE INDEX domainmetadata_idx ON domainmetadata (domain_id, kind);

CREATE TABLE cryptokeys (id INT AUTO_INCREMENT, domain_id INT NOT NULL, flags INT NOT NULL, active BOOL, content TEXT, PRIMARY KEY(id)) Engine=InnoDB CHARACTER SET=UTF8;

CREATE INDEX domainidindex ON cryptokeys (domain_id);

CREATE TABLE tsigkeys (id INT AUTO_INCREMENT, name VARCHAR (255), algorithm VARCHAR(50), secret VARCHAR(255), PRIMARY KEY (id)) Engine=InnoDB CHARACTER SET=UTF8;

CREATE UNIQUE INDEX namealgoindex ON tsigkeys(name, algorithm);

Installer PowerDNS

apt-get install -y pdns-server pdns-backend-mysql

Vous serez invité à configurer le backend MySQL. Nous allons exécuter ce processus manuellement dans quelques instants. Utilisez donc les touches fléchées pour sélectionner <Non> , puis appuyez sur ENTRÉE pour terminer l'installation.

Configurer PowerDNS

Nous devons configurer PowerDNS pour utiliser notre nouvelle base de données. Tout d'abord, supprimez les fichiers de configuration existants:

*rm /etc/powerdns/pdns.d/**

Nous pouvons maintenant créer le fichier de configuration MySQL:

nano /etc/powerdns/pdns.d/pdns.local.gmysql.conf

Entrez les données suivantes dans le fichier. N'oubliez pas d'ajouter vos propres paramètres de base de données pour gmysql- dbname , gmysql-user et en particulier gmysql-password.

############################

 # MySQL Configuration file

launch=gmysql

gmysql-host=localhost

gmysql-dbname=powerdns

gmysql-user=power_admin

password=power_admin_password

############################

Redémarrez PowerDNS pour appliquer les modifications:

service pdns restart

Testez PowerDNS

Ces étapes permettent de vérifier que PowerDNS est bien installé et peut se connecter à la base de données. Si vous ne réussissez pas les tests suivants, il y a un problème avec la configuration de votre base de données. Vérifiez si PowerDNS est à l'écoute:

netstat -tap | grep pdns

Vous devriez voir une sortie semblable à:

```
root@tld1:/home/pi# netstat -tap | grep pdns
tcp        0      0 0.0.0.0:domain          0.0.0.0:*               LISTEN      17562/pdns_server
tcp6       0      0 [::]:domain             [::]:*                  LISTEN      17562/pdns_server
root@tld1:/home/pi#
```

Vérifiez si PowerDNS répond correctement

Installation du paquet dnsutils afin d'avoir accès a la commande dig avec la commande :

apt-get install dnsutils

Maintenant on peut vérifier si PowerDNS répond.

```
root@tld1:/home/pi# dig @127.0.0.1

; <<>> DiG 9.10.3-P4-Raspbian <<>> @127.0.0.1
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: REFUSED, id: 10226
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1
;; WARNING: recursion requested but not available

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1680
;; QUESTION SECTION:
;.                              IN      NS

;; Query time: 3 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Fri Oct 12 12:46:52 UTC 2018
;; MSG SIZE  rcvd: 28

root@tld1:/home/pi#
```

Installer Poweradmin

Poweradmin est un outil d'administration DNS basé sur le Web pour PowerDNS. Il prend en charge tous les types de zone (maître , natif et esclave) et offre une prise en charge complète du super master pour l'approvisionnement automatique des zones esclaves, une prise en charge complète de l'IPv6 et plusieurs langues.

- Installez Apache et les dépendances requises pour Poweradmin:

apt-get install -y apache2 gettext libapache2-mod-php7.0 php7.0 php7.0-common php7.0-curl php7.0-dev php7.0-gd php-pear php7.0-imap php7.0-ming php7.0-mysql php7.0-xmlrpc

- Installez les modules PEAR requis:

pear install DB

pear install pear/MDB2#mysql

- Activer Mcrypt:

phpenmod mcrypt

- Redémarrez Apache pour appliquer les modifications:

service apache2 restart

- Accédez à votre répertoire personnel:

cd ~

- Téléchargez les fichiers compressés Poweradmin:

wget https://github.com/downloads/poweradmin/poweradmin/poweradmin-2.1.6.tgz

- Extraire l'archive:

tar xvzf poweradmin-2.1.6.tgz

- Déplacez le power admin répertoire dans le répertoire Web Apache:

mv poweradmin-2.1.6 /var/www/html/poweradmin

- Créez le fichier de configuration:

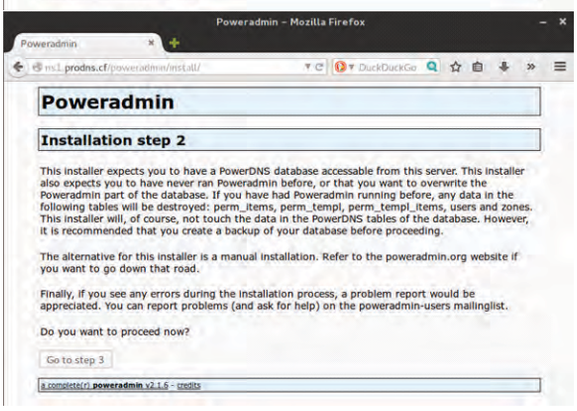
touch /var/www/html/poweradmin/inc/config.inc.php

- Donnez à l'utilisateur Apache la propriété du répertoire:

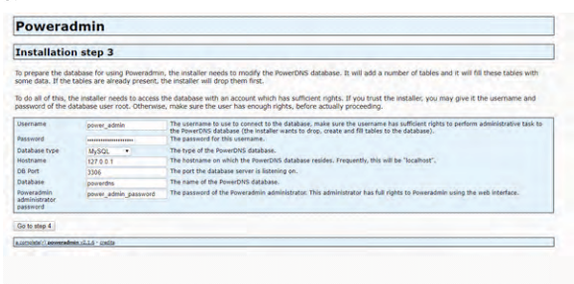
chown -R www-data:www-data /var/www/html/poweradmin/

- Configurer Poweradmin

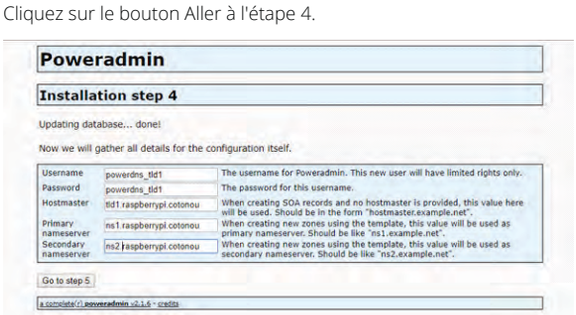
Pour terminer l'installation de Poweradmin, nous allons utiliser l'assistant de configuration basé sur le Web. Ouvrez votre navigateur Web et visitez l'URL ci-dessous, en remplaçant votre propre adresse IP ou votre nom d'hôte de serveur: http://your_server_ip/poweradmin/install/



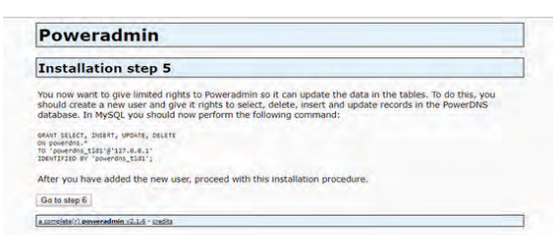
Il existe des informations précieuses sur la page d'étape 2 , en particulier pour les installations multiples de Poweradmin. Cette information ne s'applique pas directement à ce tutoriel. Lorsque vous avez terminé de lire la page, cliquez sur le bouton Aller à l'étape 3.



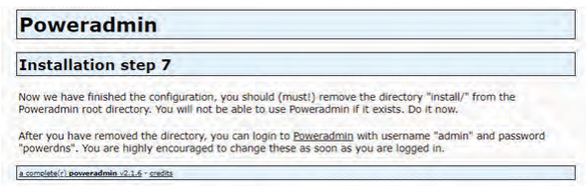
Cliquez sur le bouton Aller à l'étape 4.



Cliquez sur le bouton Aller à l'étape 5



Vérifiez que les informations de la base de données sont correctes. Si vous avez choisi de créer un nouvel utilisateur et un nouveau mot de passe, vous devez vous connecter à votre base de données MySQL et ajouter le nouvel utilisateur en copiant / collant le bloc de code affiché à l'écran, en commençant par GRANT. Cliquez ensuite sur le bouton Aller à l'étape 6 puis à la page 7 pour terminer l'installation.



Vous recevrez le nom d'utilisateur admin et le mot de passe de votre panneau de contrôle Poweradmin.

Nous avons terminé la configuration de Poweradmin. Pour nettoyer, retournez sur votre serveur et supprimez le répertoire d'installation. Poweradmin exige que nous fassions cela avant de pouvoir nous connecter:

rm -rf /var/www/html/poweradmin/install/

Si vous devez modifier les paramètres de Poweradmin une fois l'installation terminée, éditez ce fichier:

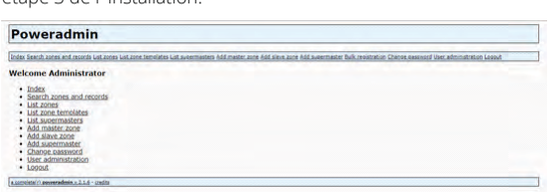
nano /var/www/html/poweradmin/inc/config.inc.php

Créez votre premier enregistrement DNS

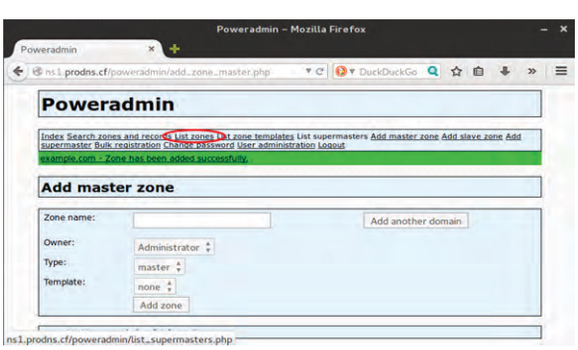
Accédez au panneau de configuration de Poweradmin:

http://your_server_ip/poweradmin/

Connectez-vous à votre panneau de configuration Poweradmin à l'aide des informations d'identification que vous avez définies lors de la configuration. Le nom d'utilisateur est admin et le mot de passe est le mot de passe de l'administrateur Poweradmin à partir de l' étape 3 de l' installation.



Cliquez sur le lien Liste des zones dans le menu supérieur.



Cliquez sur le bouton d'édition de votre fichier de zone, qui ressemble à un petit crayon à gauche de l'entrée de zone.

DNSathon