PR02 : SIO SISR – Mise en place d'une passerelle et d'un Firewall sous Debian 11





Sommaires

Contexte :
Objectifs :
Cahier des charges :
Solution :4
Schéma ASI :4
Prérequis :5
Configuration du routeur et du Nat :5
Configuration du serveur et mise en place des services :7
Configuration réseaux :7
Configuration du serveur FTP :7
Installation de Proftpd :7
Configuration du Serveur Web :8
Installation d'Apache2 :8
Configuration des deux intranets :8
Configuration des Virtual hosts :9
Mise en place du protocole HTTPS :11
Configuration du DNS :15
Installation de BIND9 :16
Déclaration et création des zones :16
Configuration réseaux :19
Pour le poste Administrateur :20
Pour le poste Formation :20
Filtrage du trafic par Iptables :20
Phase d'expérimentation et de test du parc informatique :22
Test de connexion avec notre Serveur :22
Accès Internet :24
Conclusion :

Contexte :

- Notre centre de formation regroupant plusieurs enseignes dont MBWay et DigitalSchool, met à disposition des élèves un serveur Web hébergeant un intranet pour chacune d'elle : il s'agit d'un serveur web mutualisé.

- Dans l'architecture initiale, les sites web de chaque enseigne étaient hébergés sur un serveur dans le LAN Administratif.

- Suite à quelques tentatives d'intrusion dans les serveurs locaux du réseau administratif, il a été décidé de sécuriser celui-ci en le limitant strictement aux employés.

- Dans le cadre d'un stage, vous avez été chargé par votre centre de formation de mettre en place une maquette, au moindre coût, pour montrer la faisabilité de la solution.

Objectifs :

- 1) Maquetter le nouveau réseau et filtrer les flux.
- 2) Créer un sous-réseau nommé DMZ pour héberger les services partagés par le personnel et les stagiaires
- 3) A termes, ce réseau DMZ devrait être accessible depuis Internet
- 4) Le serveur Web héberge un site pour chaque établissement. Pour sécuriser les transactions les sites ne doivent être accessibles qu'en https soit https://www.mbway.lan ou https://www.digitalschool.lan. Les sites web sont accessibles à TOUS.
- 5) Mettre en place un service DNS
- 6) Mettre en place des règles de pare-feu afin de sécuriser l'installation ainsi qu'un protocole SSH.
- 7) Effectuer une démonstration montrant la sécurité et la fiabilité de notre projet, répondant au cahier des charges.

Cahier des charges :

- Permettre l'accès au serveur Web dans la DMZ pour tous, LAN Administratif et Formation. Le serveur Web hébergera aussi les service DNS.
- Permettre l'accès à internet pour tous en utilisant le Routeur Debian (R) comme passerelle. Ce routeur fera office de Firewall pour filtrer les accès à la DMZ.
- Permettre l'accès au service FTP à un seul poste, celui de l'administrateur situé dans

le LAN Administratif

- Les postes de l'espace Formation ne pourront pas accéder au service FTP.
- Permettre un accès SSH à un seul poste, celui de l'administrateur situé dans le LAN Administratif
- Les autres périphériques du réseau Administratif et ceux du réseau Formation ne pourront pas accéder en SSH au serveur Web.
- Mettre en place les tests de validation des règles ci-dessus.
- Fournir une documentation expliquant et validant chacune des demandes du cahier des charges

Solution :

Pour répondre au problème du campus, j'ai donc mis en place une DMZ (Demilitarized Zone) hébergeant mon serveur WEB (Apache2) avec les 2 intranets, un serveur Proftpd pour le transfert de fichiers et une application serveur SSH avec OpenSSH.

J'ai également mis en place un routeur Debian hébergeant mon service DNS et mes règles de Firewall, ainsi que 2 machines clients. Une machine linux pour le réseau formation et une machine Linux pour l'administrateur.

Schéma ASI :



Prérequis :

Tout d'abord nous allons devoir installer puis paramétrer une machine virtuelle qui nous servira de Serveur WEB (Apache2), DNS (bind9), FTP (Proftpd) et une application serveur SSH avec OpenSSH.

Ensuite on va configurer deux VM sous Debian : une pour le réseau administrateur et une autre pour le réseau formation.

Les cartes réseau seront configurer en mode « Accès par pont »

Nous utiliserons et configurerons un routeur sous Debian 11 avec 4 interfaces réseau, qui nous servira de pare-feu grâce aux règles iptables que nous allons déployer.

Configuration du routeur et du Nat :

On va d'abord aller dans le fichier /etc/network/interfaces afin de paramétrer notre carte réseaux avec la commande sudo nano /etc/network/interfaces.

On redémarrera les interfaces avec la commande sudo systemctl restart networking.service pour que les changements prennent effet.

Il ne faut pas oublier d'activer l'accès par pont.

On aura pour le routeur les interfaces ci- dessous :

auto ens33 iface ens33 inet dhcp	Ens33 sert à avoir l'accès internet.
<pre>post-up iptables-restore < /etc/iptables_rules.save </pre>	Ens34 sera la carte pour le lan Administrateur.
iface ens34 inet static address 192.168.10.254 netmask 255.255.255.0	Ens35 sera la carte pour le lan Formation.
auto ens35 iface ens35 inet static address 192.168.20.254 netmask 255 255 0	Ens36 sera la carte pour la DMZ.
auto ens36 iface ens36 inet static address 192.168.30.254 netmask 255.255.255.0 root@debianRouter:~#	La ligne post-up nous sert pour le Nat et les règles iptables, nous le reverrons plus loin dans le dossier.

Maintenant que les interfaces réseaux sont configurer on va pouvoir configurer notre Vm en mode routeur et installer le NAT.

Mise en place du mode routeur :

- On va aller dans le fichier /etc/sysctl.conf et on décommente la ligne net.ipv4.ip_forward=1

Ð	utilisateur@debianRouter: ~
GNU nano 5.4	/etc/sysctl.conf
<pre># prevent some spoofing at #net.ipv4.conf.default.rp_ #net.ipv4.conf.all.rp_filt</pre>	ttacks _filter=1 ter=1
<pre># Uncomment the next line # See http://lwn.net/Artic # Note: This may impact IF #net.ipv4.tcp_syncookies=:</pre>	to enable TCP/IP SYN cooki cles/277146/ Pv6 TCP sessions too L
<pre># Uncomment the next line net.ipv4.ip_forward=1</pre>	to enable packet forwarding
<pre># Uncomment the next line</pre>	to enable packet forwardin

On configure ensuite le Nat avec la commande :

 iptables -t nat -A POSTROUTING -o ens33 -j MASQUERADE avec ens33 qui est la carte qui sort sur internet.

On vérifie que la commande et bonne avec iptables -L -t nat et on vérifie que l'on a notre règle dans la chaine postrouting

utilisateur@debianRouter:/\$ sudo iptables -L -t nat Chain PREROUTING (policy ACCEPT) target prot opt source destination Chain INPUT (policy ACCEPT) target prot opt source destination Chain OUTPUT (policy ACCEPT) target prot opt source destination Chain POSTROUTING (policy ACCEPT) target prot opt source destination MASQUERADE all -- anywhere anywhere utilisateur@debianRouter:/\$

On automatise le montage des règles au démarrage : - iptables-save > /etc/iptables_rules.save

Et on rajoute cette ligne dans notre fichier networking :

auto ens33 iface ens33 inet dhcp post-up iptables-restore < /etc/iptables_rules.save

Configuration du serveur et mise en place des services :

Configuration réseaux :

On va d'abord aller dans le fichier /etc/network/interfaces afin de paramétrer notre carte réseaux avec la commande sudo nano /etc/network/interfaces.

On redémarrera les interfaces avec la commande sudo systemctl restart networking.service pour que les changements prennent effet.

Il ne faut pas oublier d'activer l'accès par pont.

Pour notre serveur on lui définit comme adresse IP : 192.168.30.10

E	• utilisateur@debianServeur: ~	۹	≡	×
(# 1 # 3	GNU nano 5.4 /etc/network/interfaces This file describes the network interfaces available on your syste and how to activate them. For more information, see interfaces(5).	m		
sou	urce /etc/network/interfaces.d/*			
# 1 aut ifa	The loopback network interface to lo ace lo inet loopback			
au† ifa	to ens33 ace ens33 inet static address 192.168.30.10 netmask 255.255.255.0 gateway 192.168.30.254			
^G ^X	[Lecture de 14 lignes] Aide ^O Écrire ^W Chercher ^K Couper ^T Exécuter ^C Quitter ^R Lire fich.^\ Remplacer ^U Coller ^J Justifier ^	Empl Alle	aceme r lig	nt

Maintenant que l'interface réseau est configurée on va pouvoir installer les services nécessaires.

Configuration du serveur FTP :

Nous devons installer au préalable ProFTPD sur notre VM SERVEUR Debian11

Installation de Proftpd :

- sudo apt-get update

sudo apt install proftpd

Une fois installé nous pouvons choisir de créer des utilisateurs mais nous allons nous connecter en mode utilisateur pour nos tests.

FTP n'est pas un protocole sécurisé. Pour éviter la transmission d'informations en clair, il est nécessaire de crypter les données en transit. Nous utiliserons donc principalement le protocole SFTP.

Pour cela, nous devrons installer OpenSSH sur les serveurs ainsi que sur les postes clients via la commande :

- sudo apt install openssh-server

```
utilisateur@debianClient:~$ sudo apt-cache policy openssh-server
openssh-server:
    Installé : 1:8.4p1-5+deb11u3
    Candidat : 1:8.4p1-5+deb11u3
    Table de version :
    *** 1:8.4p1-5+deb11u3 500
        500 http://deb.debian.org/debian bullseye/main amd64 Packages
        500 http://deb.debian.org/debian-security bullseye-security/main am
d64 Packages
        100 /var/lib/dpkg/status
```

Configuration du Serveur Web :

On va devoir configurer Apache 2 avec nos 2 intranets et mettre en place le protocole HTTPS

Dans un premier temps faire sudo apt-get update afin d'avoir les dernières versions disponibles (il faut le faire avant toutes installations sous Debian)

Installation d'Apache2 :

- sudo apt install Apache2

```
utilisateur@debianServeur:~$ sudo apache2 -v
Server version: Apache/2.4.56 (Debian)
Server built: 2023-04-02T03:06:01
```

On constate donc qu'Apache2 est correctement installé.

Configuration des deux intranets :

On va créer deux dossiers MBWay et DigitalSchool :

Aller dans le dossier cd /var/www/html est créé deux dossiers avec la commande mkdir :

```
utilisateur@debianServeur:~$ cd /var/www/html/
utilisateur@debianServeur:/var/www/html$ ls -l
total 20
drwxr-xr-x 2 root root 4096 15 avril 12:46 digitalschool
-rw-r--r-- 1 root root 10701 7 avril 21:19 index.html
drwxr-xr-x 2 root root 4096 13 avril 18:18 mbway
utilisateur@debianServeur:/var/www/html$
```

Pour chaque dossier nous allons créer un fichier index.html que nous allons ensuite configurer :

```
utilisateur@debianServeur:/var/www/html$ cd mbway
utilisateur@debianServeur:/var/www/html/mbway$ ls -l
total 4
-rw-r--r- 1 root root 123 13 avril 18:18 index.html
utilisateur@debianServeur:/var/www/html/mbway$ cd ..
utilisateur@debianServeur:/var/www/html$ cd digitalschool
utilisateur@debianServeur:/var/www/html$ cd digitalschool
utilisateur@debianServeur:/var/www/html/digitalschool$ ls -l
total 8
-rw-r--r- 1 root root 131 14 avril 11:13 index.html
```

Fichier index.html pour DigitalSchool :

On va créer le même type pour MBWay.

Nous avons donc créé un dossier où nous importerons nos fichiers html, PHP, JavaScript etc... respectifs à chaque site. Ici nous n'avons déposé qu'un « index.html » pour le moment.

Configuration des Virtual hosts :

Grâce à l'étape précédente nous avons créé nos intranets.

De ce fait pour y accéder nous sommes obligés d'écrire : http://192.168.30.5/mbway/ pour accéder au site d'MBWay par exemple.

Pour corriger cela nous allons donc configurer des Virtual Host pour accéder à nos sites depuis l'adresse http://www.mbway.lan

On va aller dans le dossier /etc/apache2/sites-available/

On a un fichier 000-default.conf avec une configuration par default que l'on va copier (commande cd) et créer deux fichiers un pour MBWay et un autre pour DigitalSchool.

```
utilisateur@debianServeur:~$ cd /etc/apache2/sites-available/
utilisateur@debianServeur:/etc/apache2/sites-available$ ls -l
total 36
-rw-r--r-- 1 root root 1332 2 avril 2023 000-default.conf
-rw-r--r-- 1 root root 6387 14 avril 19:53 default-ssl.conf
-rw-r--r-- 1 root root 1351 13 avril 15:31 digitalschool.conf
-rw-r--r-- 1 root root 6390 16 avril 09:20 digitalschool-ssl.conf
-rw-r--r-- 1 root root 1636 14 avril 19:38 mbway.conf
-rw-r--r-- 1 root root 6374 16 avril 09:19 mbway-ssl.conf
utilisateur@debianServeur:/etc/apache2/sites-available$
```

On va ensuite les configurer :

```
utilisateur@debianServeur:/etc/apache2/sites-available$ cat mbway.conf
 <VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        ServerName www.mbway.lan
        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html/mbway
utilisateur@debianServeur:/etc/apache2/sites-available$ cat digitalschool.conf
<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port .
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) th.
        # value is not decisive as it is used as a last resort host regardless
        # However, you must set it for any further virtual host explicitly.
        ServerName www.digitalschool.lan
        ServerAdmin webmaster@localhost
```

On modifie ServerName par le nom de nos serveurs : <u>www.mbway.lan</u> et www.digitalschool.lan.

On lui indique la route avec DocumentRoot : /var/www/html/mbway et /var/www/html/digitalschool .

DocumentRoot /var/www/html/digitalschool

On va activer nos sites avec la commande a2ensite + le nom de notre site

Maintenant si on ajoute www.mbway.lan en alias à notre serveur dans le fichier /etc/hosts de notre client, vous pouvez voir la page d'accueil de notre site dans notre navigateur.

On vérifie que nos sites fonctionnent :

Bienvenue sur le site de MBWAY

Ceci est la page d'acceuil.

Mise en place du protocole HTTPS :

La connexion à nos sites intranets ne se fait de base qu'en HTTP ce qui n'est absolument pas sécurisé. Pour éviter de faire circuler des informations en clair, nous devons crypter les données qui transitent par le biais du protocole HTTPS.

On a un fichier avec une configuration déjà présente sous Debian :

```
utilisateur@debianServeur:/etc/apache2/sites-available$ ls -l
total 36
-rw-r--r-- 1 root root 1332 2 avril 2023 000-default.conf
-rw-r--r-- 1 root root 6387 14 avril 19:53 default-ssl.conf
-rw-r--r-- 1 root root 1351 13 avril 15:31 digitalschool.conf
-rw-r--r-- 1 root root 6390 16 avril 09:20 digitalschool-ssl.conf
-rw-r--r-- 1 root root 1636 14 avril 19:38 mbway.conf
-rw-r--r-- 1 root root 6374 16 avril 09:19 mbway-ssl.conf
utilisateur@debianServeur:/etc/apache2/sites-available$
```

On va donc copier le fichier default-ssl et créer deux fichiers pour chaque site puis les configurer :

Activ	ités 🕞 Terminal 🔻 16 avri	il 10:09
Ð	utilisateur@debianServeur: /	etc/apache2/sites-available
<ifm< td=""><td>odule mod_ssl.c> <pre></pre></td><td>1001-551.0011</td></ifm<>	odule mod_ssl.c> <pre></pre>	1001-551.0011
	ServerName www.digitalschool.lan	
	DocumentRoot /var/www/html/digitalschool	
	<pre># Available loglevels: trace8,, trace1, debug, inf # error, crit, alert, emerg. # It is also possible to configure the loglevel for pa # modules, e.g. #LogLevel info ssl:warn</pre>	ⁱ o, notice, warn, articular
	ErrorLog \${APACHE_LOG_DIR}/error.log CustomLog \${APACHE_LOG_DIR}/access.log combined	
	# For most configuration files from conf-available/, w # enabled or disabled at a global level, it is possibl # include a line for only one particular virtual host. # following line enables the CGI configuration for thi # after it has been globally disabled with "a2disconf' #Include conf-available/serve-cgi-bin.conf	which are le to For example the s host only '.
	<pre># SSL Engine Switch: # Enable/Disable SSL for this virtual host. SSLEngine on</pre>	
	<pre># A self-signed (snakeoil) certificate can be create # the ssl-cert package. See # /usr/share/doc/apache2/README.Debian.gz for more i # If both key and certificate are stored in the same # SSLCertificateFile directive is needed. SSLCertificateFile /etc/ssl/certs/ssl-cert-snaked SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snaked</pre>	d by installing info. file, only the pil.pem pil.key
Ac	tivités 🕑 Terminal 🔻 16 avril	10:10
1	utilisateur@debianServeur:/e	tc/apache2/sites-available
ut <i< td=""><td>ilisateur@debianServeur:/etc/apache2/sites-available\$ cat mbway-ssl.c fModule mod_ssl.c> <virtualhost _default_:443=""> ServerAdmin webmaster@localhost</virtualhost></td><td>cont</td></i<>	ilisateur@debianServeur:/etc/apache2/sites-available\$ cat mbway-ssl.c fModule mod_ssl.c> <virtualhost _default_:443=""> ServerAdmin webmaster@localhost</virtualhost>	cont
	DocumentRoot /var/www/html/mbway ServerName www.mbway.lan	
	<pre># Available loglevels: trace8,, trace1, debug, info # error, crit, alert, emerg. # It is also possible to configure the loglevel for par # modules, e.g. #LogLevel info ssl:warn</pre>	o, notice, warn, rticular
	ErrorLog \${APACHE_LOG_DIR}/error.log CustomLog \${APACHE_LOG_DIR}/access.log combined	
	# For most configuration files from conf-available/, wh # enabled or disabled at a global level, it is possible # include a line for only one particular virtual host. # following line enables the CGI configuration for this # after it has been globally disabled with "a2disconf". #Include conf-available/serve-cgi-bin.conf	tich are ≥ to For example the 5 host only

SSL Engine Switch: # Enable/Disable SSL for this virtual host. SSLEngine on

- # A self-signed (snakeoil) certificate can be created by installing # the ssl-cert package. See # /usr/share/doc/apache2/README.Debian.gz for more info. # If both key and certificate are stored in the same file, only the # SSLCertificateFile directive is needed. SSLCertificateFile /etc/ssl/certs/ssl-cert-snakeoil.pem CCLCertificateFile /etc/ssl/certs/ssl-cert-snakeoil.pem
- SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key

On donne le nom de notre serveur :

ServerName <u>www.mbway.lan</u> et ServerName <u>www.digitalschool.lan</u> ainsi que la route avec DocumentRoot /var/www/html/mbway et /var/www/html/digitalschool .

On va activer a2enmod ssl puis a2ensite default-ssl

On va activer nos sites avec la commande a2ensite + le nom du virtual Host

Puis redemarrer apache2 : service apache2 reload

La recommandation est de créer/acheter un certificat pour chaque site plutôt que d'utiliser la configuration de base d'Apache2

C'est donc ce que nous allons faire :

Dans un premier temps nous allons verifier que le module SSL est bien activée :

- sudo a2enmod ssl

Nous allons ensuite générer nos clés privées et des CSR(certificate signing request) : Pour Mbway : openssl req -newkey rsa:2048 -nodes -keyout /etc/ssl/private/mbway.key -out /etc/ssl/certs/mbway.csr

Et pour Digitalschool : openssl req -newkey rsa:2048 -nodes -keyout /etc/ssl/private/digitalschool.key -out /etc/ssl/certs/ digitalschool.csr

Nous devrons répondre à une suite de question :

Not corpose the pose of t

Nous allons enfin passer a la création de nos certificats auto-signés :

Pour Mbway : sudo openssl req x509 -req -days 365 -in /etc/ssl/certs/mbway.csr -signkey

/etc/ssl/private/mbway.key -out /etc/ssl/certs/mbway.crt

Et pour digitalschool : sudo openssl req x509 -req -days 365 -in /etc/ssl/certs/digitalschool.csr -signkey /etc/ssl/private/digitalschool.key -out /etc/ssl/certs/digitalschool.crt

Nous devons ensuite modifier les lignes :

- SSLCertificateFile
- SSLCertificateKeyFile

Pour les sites Mbway et Digitaschool dans mbway-ssl.conf et digitalschool-ssl.conf

Đ		utilisateur@debian11: ~
GNU nano 5.4		mbway-ssl.conf
IfModule mod_ssl VirtualH<	.c> lost_default:443>	
S	erverAdmin webmaster@localhost	
S	erverName www.mbway.lan ocumentRoot /var/www/html/mbway	
#	Available loglevels: trace8,, trace1, debug, info, notice,	warn,
#	It is also possible to configure the loglevel for particular modules, e.g.	
" E C	<pre>:rrorLog \${APACHE_LOG_DIR}/error.log :ustomLog \${APACHE_LOG_DIR}/access.log combined</pre>	
# # # # #	For most configuration files from conf-available/, which are enabled or disabled at a global level, it is possible to include a line for only one particular virtual host. For examp following line enables the CGI configuration for this host only after it has been globally disabled with "a2disconf". Include conf-available/serve-cgi-bin.conf	le the Y
# # S	 SSL Engine Switch: Enable/Disable SSL for this virtual host. SLEngine on 	
# # # \$ \$ \$	A self-signed (snakeoil) certificate can be created by insta the ssl-cert package. See /usr/share/doc/apache2/README.Debian.gz for more info. If both key and certificate are stored in the same file, onl SSLCertificateFile directive is needed. SLCertificateFile /etc/ssl/certs/mbway.crt SLCertificateKeyFile /etc/ssl/private/mbway_key	lling y the
#	Server Certificate Chain:	

Ð		utilisateur@debian11: ~
GNU nano 5.4		digitalschool-ssl.conf
<ifmodule mod_s<="" td=""><td>ssl.c></td><th></th></ifmodule>	ssl.c>	
<virtua< td=""><td>alHost _default_:443> ServerAdmin webmaster@localhost</td><th></th></virtua<>	alHost _default_:443> ServerAdmin webmaster@localhost	
	ServerName www.digitalschool.lan DocumentRoot /var/www/html/digitalschool	
	<pre># Available loglevels: trace8,, trace1, debu # error, crit, alert, emerg. # It is also possible to configure the loglevel # modules, e.g. #LogLevel info ssl:warn</pre>	g, info, notice, warn, for particular
	ErrorLog \${APACHE_LOG_DIR}/error.log CustomLog \${APACHE_LOG_DIR}/access.log combined	
	# For most configuration files from conf-availat # enabled or disabled at a global level, it is p # include a line for only one particular virtual # following line enables the CGI configuration f # after it has been globally disabled with "a2di #Include conf-available/serve-cgi-bin.conf	<pre>le/, which are ossible to host. For example the or this host only sconf".</pre>
	<pre># SSL Engine Switch: # Enable/Disable SSL for this virtual host. SSLEngine on</pre>	
	<pre># A self-signed (snakeoil) certificate can be # the ssl-cert package. See # /usr/share/doc/apache2/README.Debian.gz for # If both key and certificate are stored in th # SSLCertificateFile directive is needed. SSLCertificateFile /etc/ssl/certs/digitalsc SSLCertificateKeyFile /etc/ssl/private/digitalsc</pre>	created by installing more info. e same file, only the hool.crt hool.key
	" como constructo conto	

On va activer nos sites avec la commande a2ensite + le nom du virtual Host

Puis redemarrer apache2 : sudo systemctl restart apache2

Maintenant si on ajoute www.mbway.lan en alias à notre serveur dans le fichier /etc/hosts de notre client, vous pouvez voir la page d'accueil de notre site dans notre navigateur :



Bienvenue sur le site de Mbway !

Nous arrivons donc bien a nous connecter en https !

Configuration du DNS :

On commence par faire apt-get upgrade

Installation de BIND9 :

On fait ensuite sudo apt install Bind9 puis sudo apt install bind9utils.

Déclaration et création des zones :

On va commencer pas déclarer les zones dans /etc/bind

Ð				ι	utilis	ateur@	debianSo	erveur: /etc/bind	۹	×
utilisate	ur@d	ebiar	Serve	eur:~9	\$ C(d /etc,	/bind			
utilisate	ur@d	ebiar	Serve	eur:/e	etc,	/bind\$	ls -l			
total 56										
- rw-rr-	- 1	root	root	1991	29	juil.	12:05	bind.keys		
-rw-rr-	- 1	root	root	237	29	juil.	12:05	db.0		
- rw-rr-	- 1	root	root	271	29	juil.	12:05	db.127		
-rw-rr-	- 1	root	root	237	29	juil.	12:05	db.255		
-rw-rr-	- 1	root	bind	362	14	oct.	15:42	db.digitalschool.lan		
- rw-rr-	- 1	root	root	353	29	juil.	12:05	db.empty		
-rw-rr-	- 1	root	root	270	29	juil.	12:05	db.local		
-rw-rr-	- 1	root	bind	337	14	oct.	15:42	db.mbway.lan		
-rw-rr-	- 1	root	bind	463	29	juil.	12:05	named.conf		
-rw-rr-	- 1	root	bind	498	29	juil.	12:05	named.conf.default-zones		
- rw-rr-	- 1	root	bind	318	30	sept.	14:55	named.conf.local		
-rw-rr-	- 1	root	bind	848	7	oct.	16:53	named.conf.options		
- rw-r	- 1	bind	bind	100	30	sept.	14:15	rndc.key		
-rw-rr-	- 1	root	root	1317	29	juil.	12:05	zones.rfc1918		
utilisate	ur@d	ebiar	Serve	eur:/e	etc,	/bind\$				

On va aller dans le fichiers named.conf.local et le configurer :

```
utilisateur@debianServeur:/etc/bind$ cat named.conf.local
11
// Do any local configuration here
11
// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";
zone "mbway.lan" {
        type master;
        file "/etc/bind/db.mbway.lan";
};
zone "digitalschool.lan" {
        type master;
        file "/etc/bind/db.digitalschool.lan";
};
utilisateur@debianServeur:/etc/bind$
```

On le configure en mode master et ont créé deux zones pour chacun de nos sites et on respecte la convention db.nomDomaine pour le déclarer.

On va ensuite créer nos fichiers de zones :



On lui donne les bons noms et adresse IP : debianRouteur.mbway.lan. et debianRouteur.digitalschool.lan.

On ajoute WWW pour avoir le <u>www.mbway.lan</u> et on voit également que l'on a un enregistrement de type A.

On va également configurer le fichier named.conf.options avec le DNS de Google.

Les forwarders sont d'autres serveurs DNS vers lesquels le serveur DNS local peut envoyer les requêtes qu'il ne peut pas résoudre.

On doit aussi configurer Bind9 pour accepter toutes les requêtes provenant d'adresses privées définies par les standards RFC1918 (qui inclut les réseaux LAN).

Voici les plages d'adresses locales selon RFC1918 :

- 10.0.0.0/8 : Pour les réseaux de classe A.
- 172.16.0.0/12 : Pour les réseaux de classe B.
- 192.168.0.0/16 : Pour les réseaux de classe C.

Dans notre fichier named.conf.options, remplacez la directive allow-recursion par une liste incluant ces plages avec :

- allow-query : qui contrôle qui peut poser n'importe quelle question DNS (locale ou externe).
- allow-recursion : qui contrôle qui peut poser des questions récursives (requêtes nécessitant que le serveur DNS interroge d'autres serveurs).

€	utilisateur@debian11: ~	٩	Ξ	×
GN	U nano 5.4 named.conf.options // If there is a firewall between you and nameservers you want // to talk to, you may need to fix the firewall to allow multipl // ports to talk. See http://www.kb.cert.org/vuls/id/800113 // If your ISP provided one or more IP addresses for stable // nameservers, you probably want to use them as forwarders. // Uncomment the following block, and insert the addresses repla // the all-0's placeholder.	e cing		
	allow-recursion { localhost; 10.0.0.0/8; 172.16.0.0/12; 192.168. allow-query { localhost; 10.0.0.0/8; 172.16.0.0/12; 192.168.0.0/ forwarders { 8.8.8.8; 4.4.4.4; };	0.0/1	.6; }; ;	:
^G А ^Х Q	<pre>//===================================</pre>	ind-k Empl	eys aceme	ent gne

On redémarre Bind9 afin que les modifications prennent effet : sudo systemctl restart bind9

Dans nos machines Debian on va devoir leur indiquer le nouveau DNS dans le fichier

Nano /etc/resolv.conf :

Nameserver : 192.168.30.10

On peut vérifier notre configuration grâce à la commande nslookup :

utilisateur@debianClient2:~\$ nslookup www.mbway.lan
Server: 192.168.30.10
Address: 192.168.30.10#53
Name: www.mbway.lan

```
Address: 192.168.30.10
```

On peut donc maintenant taper https://www.mbway.lan/

On teste la connexion :

Activ	vités 🕑 F	Firefox ESR 🔻	18 oct.	. 21:19	Ð		
۲	mbway.lan/	×	digitalschool.lan/		×	+	
\leftarrow	\rightarrow G	O 🔒 https://	/www. mbway.lan				
Bi Ceci	i est la page	nue sui e d'acceuil. efox ESR -	r le site	de	M	BWA	Y
	mbway.lan/	× dig	gitalschool.lan/	×	+		~
÷	\rightarrow G	🔿 🔒 https://www	.digitalschool.lan			\$	\bigtriangledown

Configuration réseaux :

On va d'abord aller dans le fichier /etc/network/interfaces afin de paramétrer nos cartes réseaux avec la commande sudo nano /etc/network/interfaces.

On redémarrera les interfaces avec la commande sudo systemctl restart networking.service pour que les changements prennent effet.

Il ne faut pas oublier d'activer l'accès par pont.

Pour le poste Administrateur :

source /etc/network/interfaces.d/*
The loopback network interface
auto lo
iface lo inet loopback
auto ens33
iface ens33 inet static
 address 192.168.10.10
 netmask 255.255.255.0
 gateway 192.168.10.254

Pour le poste Formation :

source /etc/network/interfaces.d/*
The loopback network interface
auto lo
iface lo inet loopback
auto ens33
iface ens33 inet static
 address 192.168.20.10
 netmask 255.255.0
 gateway 192.168.20.254

Filtrage du trafic par Iptables :

Réseau administrateur : ens34 192.168.10.0/24 IP Poste Administrateur : 192.168.10.10/24

Réseau formation : ens35 192.168.20.0/24 IP Poste Formateur192.168.20.50/24

Réseau serveur : ens36 192.168.30.0/24 IP Serveur 192.168.30.10/24

Réseau internet : ens34 DHCP

Nous allons utiliser iptables afin de créer des règles de filtrage est sécurisée notre connexion :

Nous allons regarder si on a déjà des règles en place :

root@debianRouter:~# iptables -L Chain INPUT (policy DROP) target prot opt source destination Chain FORWARD (policy DROP) target prot opt source destination Chain OUTPUT (policy ACCEPT) target prot opt source destination root@debianRouter:~#

On va bloquer le trafic qui traverse le routeur ainsi que le trafic entrant. Mais on ne bloque pas le trafic sortant.

On va autoriser le retour pour les connexions déjà établie :

- iptables -A FORWARD -m state --state ESTABLISHED -j ACCEPT
- iptables -A INPUT -m state --state ESTABLISHED -j ACCEPT
- iptables -A OUTPUT -m state --state ESTABLISHED -j ACCEPT

Puis on va établir les règles de filtrages en suivent le cahier des charges suivant :

- Permettre l'accès au serveur Web dans la DMZ pour tous, LAN Administratif et Formation :

Nous devons d'abord créer la règle de filtrage autorisant la connexion au DNS :

Autoriser le trafic DNS (UDP) depuis le Lan Formation et Lan Administration vers le serveur 192.168.30.10.

- iptables -A FORWARD -p udp --dport 53 -s 192.168.10.0/24 -d 192.168.30.10 -j ACCEPT
- iptables -A FORWARD -p udp --dport 53 -s 192.168.20.0/24 -d 192.168.30.10 -j ACCEPT

Ensuite :

- iptables -A FORWARD -s 192.168.10.0/24 -p tcp --dport 80:443 -d 192.168.30.10 -j ACCEPT
- iptables -A FORWARD -s 192.168.20.0/24 -p tcp --dport 80:443 -d 192.168.30.10 -j ACCEPT

- Permettre l'accès à internet pour tous en utilisant le Routeur Debian (R) comme passerelle.:
- iptables -A FORWARD -s 192.168.10.0/24 -p tcp --dport 80 -j ACCEPT
- iptables -A FORWARD -s 192.168.20.0/24 -p tcp --dport 443 -j ACCEPT
- Permettre un accès SSH à un seul poste, celui de l'administrateur situé dans le LAN Administratif :
- iptables -A FORWARD -p tcp --dport 22 -s 192.168.10.10 -d 192.168.30.10 -j ACCEPT

On bloque toutes les autres connexions :

- iptables -A FORWARD -p tcp --dport 22 -d 192.168.30.10 -j DROP
- Permettre l'accès au service FTP à un seul poste, celui de l'administrateur situé dans le LAN Administratif :

On va utiliser le protocole SFTP afin de se connecter au serveur FTP. On utilisera donc le protocole SHH que l'on a autoriser plus haut.

Nous allons maintenant pouvoir passer au teste de connexions.

Phase d'expérimentation et de test du parc informatique :

Test de connexion avec notre Serveur :

Depuis le poste administrateur :



Bienvenue sur le site de MBWAY

Ceci est la page d'acceuil.

Thomas Fougeras

sftp://utilisateur	@192.168.30.10 - FileZilla				
Fichier Édition Affichage Transfert Serveur Favoris ?					
# ~ E T T # 0 # 0 1 1 1 A 9	> 8				
Hôte : sftp://192.168.30.1 Identifiant : utilisateur Mot de passe :	Port : Connexion rapide 👻				
Statut : Connexion à 192.168.30.10 Statut : Using username "utilisateur". Statut : Connected to 192.168.30.10 Statut : Démarrage du téléchargement de /home/utilisateur/Documents/procedure p Statut : Transfert de fichier réussi, 8148 octets transférés en 1 seconde	pour H.odt				
Site local : /home/utilisateur/Documents/	Site distant : /home/utilisateur/Documents				
	.local				
Documents	? .mozilla				
Nom de fichier A Taille de fic Type de fichier Dernière modifica	? .ssh				
······································	? Bureau				
procedure pour H.odt 8148 odt-fichier 06/01/2025 17:	Documents				
	? Images				
	Nom de fichier A Taille de fi Type de fict Dernière modi Droits d'acc Propriéta				
1 fichier. Taille totale : 8 148 octets	Sélection de 1 fichier. Taille totale : 8148 octets				
Serveur / Fichier local Directio Fichier distant Taille Priorit	té Temps				
📱 sftp://utilisateur@19					
/home/utilisateur/D << /home/utilisateur/Docu 8148 Norm.	06/01/2025 17:21:39				
Fichiers en file d'attente Transferts échoués Transferts réussis (1)					
	G ⑦ File d'attente : vide				

Depuis le poste formation :

ሰ ዞ	ome 🗙 🕞 s	Serveur 🗙 🕞	Administration-ClientDebian2	×	Formation-ClientDebian	×
Activ	vités 🛛 🕹 Fir	efox ESR 🔻	19 oct. 02:56			?
	mbway.lan/	×	digitalschool.lan/	×	+	~
÷	\rightarrow G	O 🔒 https:/	//www.digitalschool.lan		☆	${igsidential}$

Bienvenue sur le site de Digitalschool

Ceci est la page d'acceuil.

Thomas Fougeras

🟠 Home 🗙 🕞 Serveu	r 🗙 🕞 Administration-ClientDe	ebian2 X	Formation-Cli	entDebian ×	\leftarrow \blacktriangleright			
Activités 🗗 FileZilla	·▼ 19 oc	. 02:59		.?.				
FileZilla ×								
Fichier Édition Afficha	ge Transfert Serveur Favo	ris ?						
	T 🗰 🖸 🎼 🖸 🛼	1	o 🍳 🤌					
Hôte : 192.168.30.10	Identifiant : utilisateur	Mot de passe :	••••	Port:	Connexio			
Erreur : Impossible d'éta Statut : Attente avant nouve Statut : Connexion à 192.168 Statut : Échec de la tentative Erreur : Impossible d'éta	blir une connexion au serveur l essai 3.30.10:21 e de connexion avec "ENETUNRE blir une connexion au serveur	ACH - Réseau ina	accessible".					
Site local : /home/utilisat	eur/ 💌	Site distant :						
▼ 1								
Nom de fichier 🔺	Faille de fic Type de fichier De	Nom de fichie	r 🔨 Taille	de fi Type de ficł	Dernière mo			
4 fichiers et 14 dossiers. Ta	aille totale : 5 535 octets	Déconnecté.						
Serveur / Fichier local	Directio Fichier distant	Taille	Priorité Statu	t				
Fichiers en file d'attente	Transferts échoués Transfer	ts reussis						
		(Ø File d'attente : vide					
To direct input to this VM, o	lick inside or press Ctrl+G.			🖥 💿 🚯 📇 දුළ				

Accès Internet :

Depuis le poste administrateur :

PosteAdm	inistrateur ×	RouteurDebian	X 🗋 Serveur	X 🖵 My Computer	X RouteurDebian-iptables	X Debian 11	×	Zabbik	×
Activités	📐 Termina	l 🕶			18 janv. 22:24				
					-				
					- town O de bie w Client De				
		Ш		utilis	ateur@debianClient2: ~		4	=	×
		utilisat	eur@debianC	lient2:~\$ ping	8.8.8.8				
		64 bytes	from 8.8.8	.8; 56(84) byle .8: icmp seq=1	ttl=114 time=10.8 ms				
		64 bytes	from 8.8.8	.8: icmp_seq=2	ttl=114 time=10.2 ms				
		8.8.	8.8 ping sta	atistics					- 18
		2 packet	s transmitte	ed, 2 received,	0% packet loss, tim	e 1002ms			- 18
		rtt min/ utilisat	avg/max/mdev eur@debianCl	/ = 10.225/10.5	28/10.831/0.303 ms				- 18
			car ga an a dan a	Ψ					- 18
									- 18

Depuis le poste formation :

C Serveur	X 🖵 My Computer	X C RouteurDebian-iptables	X Debian11	X 🖸 Zabbik	× 🕞 Form	ation	×	•
Activités	🕞 Terminal 🔻	1 8 ja	18 janv. 22:23				•))	() ▼
Ð	utilisateur@debianClient2: ~						×	
utili PING 64 by 64 by ^C 8 2 pac rtt m utili	sateur@debianC 8.8.8.8 (8.8.8 vtes from 8.8.8 vtes from 8.8.8 8.8.8.8 ping st kets transmitt nin/avg/max/mde sateur@debianC	<pre>lient2:~\$ ping 8.8.8 .8) 56(84) bytes of .8: icmp_seq=1 ttl=: .8: icmp_seq=2 ttl=: atistics ed, 2 received, 0% p v = 10.675/10.966/1: lient2:~\$</pre>	3.8 data. 114 time=11.3 r 114 time=10.7 r Dacket loss, t: 1.257/0.291 ms	ms ms ime 1003ms				

Conclusion:

En conclusion, tous les critères du cahier des charges ont été respectés. Le serveur Debian fonctionne dans un réseau unique, incluant un serveur web, FTP et DNS.

Le serveur web héberge deux sites accessibles à tous les utilisateurs.

Le serveur FTP est exclusivement réservé aux administrateurs, et le réseau administratif est isolé du réseau de formation.

L'administrateur a un accès SSH sécurisé au serveur.

Nos tests confirment que les règles de filtrage ont été configurées correctement, garantissant que tout fonctionne comme prévu.