Infra 01 : DNS

Installation et configuration des VMS

Les machines seront ainsi paramétrés :

Nom	Configuration
ns01.projet.intranet	Adresse : 192.168.42.10
ns02.projet.intranet	Adresse : 192.168.42.11
resolv.projet.intranet	Adresse : 192.168.42.12
client01.projet.intranet	Adresse : 192.168.42.101

On vérifie que les machines peuvent bien se joindre :

```
root@NS02:~# ping 192.168.42.101
PING 192.168.42.101 (192.168.42.101) 56(84) bytes of data.
64 bytes from 192.168.42.101: icmp_seq=1 ttl=64 time=0.676 ms
root@NS02:~# ping 192.168.42.10
PING 192.168.42.10 (192.168.42.10) 56(84) bytes of data.
64 bytes from 192.168.42.10: icmp_seq=1 ttl=64 time=1.52 ms
root@NS02:~# ping 192.168.42.11
PING 192.168.42.11 (192.168.42.11) 56(84) bytes of data.
64 bytes from 192.168.42.11: icmp_seq=1 ttl=64 time=0.014 ms
root@NS02:~# ping 192.168.42.12
PING 192.168.42.12 (192.168.42.12) 56(84) bytes of data.
64 bytes from 192.168.42.12: icmp_seq=1 ttl=64 time=0.416 ms
```

On peut voir que le réseau est bien paramétré et que les machines peuvent communiquer entre elles.

Installation du DNS unbound

On installe unbound via la commande `apt install unbound` puis on vérifié son bon fonctionnement :

```
root@resolv:~# service unbound status
● unbound.service - Unbound DNS server
   Loaded: loaded (/lib/systemd/system/unbound.service; enabled; vendor preset:
enabled)
   Active: active (running) since Fri 2020-12-11 13:41:49 CET; 24s ago
     Docs: man:unbound(8)
 Main PID: 842 (unbound)
    Tasks: 1 (limit: 2327)
   Memory: 8.0M
   CGroup: /system.slice/unbound.service
           └─842 /usr/sbin/unbound -d
```

Le service est bien démarré.

Configuration de unbound

Voici la configuration définie sur Unbound afin de le limiter à l'interface privée (192.168.42.12) et de limiter les requêtes au réseau privé

```
server:
        interface: 192.168.42.12 access
        access-control: 192.168.42.0/24
```

Recursif

Si rien n'est ajouté, le server dns fonctionne en récursif :

```
root@resolv:~# tcpdump port 53
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ens33, link-type EN10MB (Ethernet), capture size 262144 bytes
14:00:01.422049 IP 192.168.42.101.50226 > 192.168.42.12.domain: 65126+ [1au] A?
google.fr. (50)
14:00:01.422428 IP 192.168.42.12.60097 > bbox.lan.domain: 12094+ PTR?
12.42.168.192.in-addr.arpa. (45)
14:00:01.422645 IP 192.168.42.12.31050 > a.root-servers.net.domain: 27006% [1au]
NS? . (28)
14:00:01.427205 IP bbox.lan.domain > 192.168.42.12.60097: 12094 NXDomain* 0/0/0
(45)
14:00:01.427372 IP 192.168.42.12.41712 > bbox.lan.domain: 50934+ PTR?
101.42.168.192.in-addr.arpa. (45)
[...]
14:00:01.827228 IP 192.168.42.12.40789 > bbox.lan.domain: 30916+ PTR?
4.36.112.192.in-addr.arpa. (43)
14:00:02.440942 IP bbox.lan.domain > 192.168.42.12.40789: 30916 1/0/0 PTR
G.ROOT-SERVERS.NET. (75)
14:00:02.441211 IP 192.168.42.12.56499 > bbox.lan.domain: 21145+ PTR?
1.36.0.194.in-addr.arpa. (41)
14:00:02.482130 IP bbox.lan.domain > 192.168.42.12.56499: 21145 1/0/0 PTR
g.ext.nic.fr. (67)
```

On peut observer que le DNS qui fournit la réponse au serveur resolv est la box me fournissant l'accès internet, celle ci ayant elle même fait la demande avant de me la transférer pour qu'elle arrive finalement jusqu'au client01.

Transitaire

On modifie la configuration pour qu'il passe en transitaire :

```
forward-zone:
   name: "."
   forward-addr: 1.0.0.1@53
   forward-addr: 1.1.1.1@53
```

Voici ce que l'on peut voir en mode transitaire :

```
14:02:27.815951 IP 192.168.42.101.45309 > 192.168.42.12.domain: 51776+ [1au] A?
one.one.one.one. (56)
14:02:27.816096 IP 192.168.42.12.49614 > one.one.one.one.domain: 54027+ [1au] A?
one.one.one.one. (44)
14:02:27.816321 IP 192.168.42.12.39470 > bbox.lan.domain: 43062+ PTR? 1.0.0.1.in-
addr.arpa. (38)
14:02:27.827224 IP bbox.lan.domain > 192.168.42.12.39470: 43062 1/0/0 PTR
one.one.one.one. (67)
14:02:27.837251 IP one.one.one.one.domain > 192.168.42.12.49614: 54027 2/0/1 A
1.0.0.1, A 1.1.1.1 (76)
[...]
14:02:27.931369 IP 192.168.42.12.55209 > one.one.one.one.domain: 23448+% [1au]
DNSKEY? one.one. (36)
14:02:27.952188 IP one.one.one.one.domain > 192.168.42.12.55209: 23448$ 2/0/1
DNSKEY, RRSIG (219)
14:02:27.953825 IP 192.168.42.12.58087 > one.one.one.one.domain: 50178+% [1au]
DS? one.one.one. (40)
14:02:27.978079 IP one.one.one.one.domain > 192.168.42.12.58087: 50178$ 0/4/1
(390)
14:02:27.978769 IP 192.168.42.12.domain > 192.168.42.101.45309: 51776 2/0/1 A
1.0.0.1, A 1.1.1.1 (76)
```

En mode transitaire, on peut observer que le serveur DNS fais la requête lui même aux différents servers et la transmet ensuite au client01.

On vérifie depuis le client01 et le serveur que les requêtes fonctionnent bien :

```
#Sur le client01 :
root@client01:~# dig +short @192.168.42.12 one.one.one.one
1.1.1.1
1.0.0.1
root@client01:~# dig +short @192.168.42.12 google.fr
142.250.74.227

#Sur le serveur :
root@resolv:~# dig +short @192.168.42.12 yahoo.fr
106.10.248.150
74.6.136.150
98.136.12.23
124.108.115.101
212.82.101.150
root@resolv:~# dig +short @192.168.42.12 epsi.fr
149.255.137.13
```

Zone 'projet.intranet'

Installation de bind

On installe le paquet avec `apt install bind9` puis on vérifie son bon fonctionnement :

```
root@NS01:~# service bind9 status
● bind9.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/bind9.service; enabled; vendor preset:
enabled)
   Active: active (running) since Fri 2020-12-11 14:25:28 CET; 58s ago
     Docs: man:named(8)
 Main PID: 727 (named)
    Tasks: 4 (limit: 2327)
   Memory: 13.6M
   CGroup: /system.slice/bind9.service
           └─727 /usr/sbin/named -u bind
```

Configuration de la zone

On configure la zone ainsi :

```
$TTL 600
projet.intranet.        IN      SOA     ns01.projet.intranet.
master.projet.intranet.(
                          2020121101          ; serial
                          7200           ; refresh
                          3600          ; retry
                          1209600           ; expire
                          3600)          ; negative caching TTL
@       IN      NS      ns01.projet.intranet
@       IN      NS      ns02.projet.intranet
ns01    IN      A       192.168.42.10
ns02    IN      A       192.168.42.11
resolv  IN      A       192.168.42.12
client01        IN      A       192.168.42.101
```

On vérifie que le fichier de zone est bon puis on recharge les zones:

```
root@NS01:~# named-checkzone projet.intranet. /etc/bind/db.projet.intranet
zone projet.intranet/IN: loaded serial 2020121101
OK
root@NS01:~# rndc reload
server reload successful
```

On vérifie le bon fonctionnement de la résolution :

```
root@NS01:~# dig +short @127.0.0.1 ns01.projet.intranet
192.168.42.10
root@NS01:~# dig +short @127.0.0.1 client01.projet.intranet
192.168.42.101
```

Configuration de Unbound sur resolv

on modifie la configuration de unbound pour :

```
include: "/etc/unbound/unbound.conf.d/*.conf"
server:
        interface: 192.168.42.12
        access-control: 192.168.42.0/24 allow
        domain-insecure: "projet.intranet"
stub-zone:
        name: "projet.intranet"
        stub-addr: 192.168.42.10@53
```

On teste la résolution depuis client01 :

```
root@client01:~# dig +short @192.168.42.12 ns01.projet.intranet
192.168.42.10
```

Installation et configuration du dns secondaire

Configuration de dns01

On modifie named.conf.local :

```
zone "projet.intranet" IN {
        type master;
        file "/etc/bind/db.projet.intranet";
        allow-transfer { localhost; 192.168.42.11; };
        notify yes;
};
```

On redémarre le service bind9

Configuration de dns02

On modifie named.conf.local

```
zone "projet.intranet" IN {
        type slave;
        masters { 192.168.42.10; };
        file "/etc/bind/db.projet.intranet";
};
```

On vérifie dans les logs de dns02 le bon transfert de la zone :

```
root@NS02:~# tail -f /var/log/syslog
Dec 11 15:22:01 NS02 named[1156]: transfer of 'projet.intranet/IN' from
192.168.42.10#53: connected using 192.168.42.11#33785
Dec 11 15:22:01 NS02 named[1156]: zone projet.intranet/IN: transferred serial
2020121101
Dec 11 15:22:01 NS02 named[1156]: transfer of 'projet.intranet/IN' from
192.168.42.10#53: Transfer status: success
Dec 11 15:22:01 NS02 named[1156]: transfer of 'projet.intranet/IN' from
192.168.42.10#53: Transfer completed: 1 messages, 8 records, 230 bytes, 0.001
secs (230000 bytes/sec)
Dec 11 15:22:01 NS02 named[1156]: zone projet.intranet/IN: sending notifies
(serial 2020121101)
```

Le transfert de zone a bien été effectué on peut maintenant tester.

On vérifie que la résolution fonctionne bien :

```
root@NS02:~# dig +short @127.0.0.1 ns01.projet.intranet
192.168.42.10
root@NS02:~# dig +short @127.0.0.1 client01.projet.intranet
192.168.42.101
```

Installation et configuration du reverse DNS sur ns01

On modifie named.conf.local :

```
zone "42.168.192.in-addr.arpa" IN {
        file "/etc/bind/db.reverse.projet.intranet";
        type master;
        allow-transfer { localhost; 192.168.42.11; };
        notify yes;

};
```

On créer le fichier de zone :

```
$TTL 600
@       IN      SOA     ns01.projet.intranet.   master.projet.intranet.(
                          2020121103         ; serial
                          7200           ; refresh
                          3600          ; retry
                          1209600           ; expire
                          3600)          ; negative caching TTL

@       IN      NS      ns01.projet.intranet.
@       IN      NS      ns02.projet.intranet.
10      IN      PTR     ns01.projet.intranet.
11      IN      PTR     ns02.projet.intranet.
12      IN      PTR     resolv.projet.intranet.
101     IN      PTR     client01.projet.intranet.
```

On redémarre le service bind9

On recharge les zones on teste la résolution

```
root@NS01:/etc/bind# service bind9 restart
root@NS01:/etc/bind# rndc reload
server reload successful
root@NS01:/etc/bind# dig @192.168.42.10 -x 192.168.42.11 +short
ns02.projet.intranet.
```

Installation et configuration du dns secondaire

On modifie named.conf.local

```
zone "42.168.192.in-addr.arpa" IN {
        type slave;
        masters { 192.168.42.10; };
        file "/etc/bind/db.reverse.projet.intranet";
};
```

On vérifie dans les logs de dns02 le bon transfert de la zone :

```
root@NS02:/etc/bind# tail -f /var/log/syslog
Dec 11 16:15:36 NS02 named[1156]: running
Dec 11 16:15:36 NS02 named[1156]: zone 42.168.192.in-addr.arpa/IN: Transfer
started.
Dec 11 16:15:36 NS02 named[1156]: transfer of '42.168.192.in-addr.arpa/IN' from
192.168.42.10#53: connected using 192.168.42.11#51821
Dec 11 16:15:36 NS02 named[1156]: zone 42.168.192.in-addr.arpa/IN: transferred
serial 2020121103
Dec 11 16:15:36 NS02 named[1156]: transfer of '42.168.192.in-addr.arpa/IN' from
192.168.42.10#53: Transfer status: success
Dec 11 16:15:36 NS02 named[1156]: transfer of '42.168.192.in-addr.arpa/IN' from
192.168.42.10#53: Transfer completed: 1 messages, 8 records, 258 bytes, 0.001
secs (258000 bytes/sec)
Dec 11 16:15:36 NS02 named[1156]: zone 42.168.192.in-addr.arpa/IN: sending
notifies (serial 2020121103)
```

Le transfert de zone a bien été effectué on peut maintenant tester.

On vérifie que la résolution fonctionne bien :

```
root@NS02:/etc/bind# dig @192.168.42.11 -x 192.168.42.11 +short
ns02.projet.intranet.
root@NS02:/etc/bind# dig @192.168.42.11 -x 192.168.42.12 +short
resolv.projet.intranet.
```

Configuration du serveur resolv

On configure la stub-zone :

```
server:
        interface: 192.168.42.12
        access-control: 192.168.42.0/24 allow
        domain-insecure: "projet.intranet"
        local-zone: "42.168.192.in-addr.arpa." transparent
stub-zone:
        name: "projet.intranet"
        stub-addr: 192.168.42.10@53
stub-zone:
        name: "42.168.192.in-addr.arpa"
        stub-addr: 192.168.42.10@53
```

On teste depuis client01:

```
root@client01:~# dig @192.168.42.12 -x 192.168.42.11 +short
ns02.projet.intranet.
root@client01:~# dig @192.168.42.12 -x 192.168.42.12 +short
resolv.projet.intranet.
root@client01:~# dig @192.168.42.12 -x 192.168.42.101 +short
client01.projet.intranet.
```

Apache

Installation et configuration des VMS

Les machines seront ainsi paramétrés :

Nom	Configuration
ns01.projet.intranet	Adresse : 192.168.42.10
ns02.projet.intranet	Adresse : 192.168.42.11
resolv.projet.intranet	Adresse : 192.168.42.12
client01.projet.intranet	Adresse : 192.168.42.101
client02.projet.intranet	Adresse : 192.168.42.102
apache.projet.intranet	Adresse : 192.168.42.20

Configuration du dns

On rajoute les champs correspondant aux nouvelles machines et au nouveau site :

```
apache          IN      A       192.168.42.20
client02        IN      A       192.168.42.102
www.epsi        IN      CNAME   apache
rh.epsi         IN      CNAME   apache
```

On incremente le serial et on peut recharger la zone. Il sera aussi nécessaire de vérifier son bon transfert :

```
#Cote NS01
root@NS01:/etc/bind# nano db.projet.intranet
root@NS01:/etc/bind# rndc reload
server reload successful
root@NS01:/etc/bind# tail -f /var/log/syslog
Dec 18 12:08:54 NS01 named[473]: client @0x7f2ab00d59b0 192.168.42.11#48449
(projet.intranet): transfer of 'projet.intranet/IN': AXFR-style IXFR ended
root@NS01:/etc/bind# dig @127.0.0.1 client02.projet.intranet +short
192.168.42.102

#Cote NS02
root@NS02:~# tail -f /var/log/syslog
Dec 18 12:08:54 NS02 named[457]: client @0x7fa8680c7220 192.168.42.10#48378:
received notify for zone 'projet.intranet'
Dec 18 12:08:54 NS02 named[457]: zone projet.intranet/IN: notify from
192.168.42.10#48378: serial 2020121801
```

```
Dec 18 12:08:54 NS02 named[457]: zone projet.intranet/IN: Transfer started.
Dec 18 12:08:54 NS02 named[457]: transfer of 'projet.intranet/IN' from
192.168.42.10#53: connected using 192.168.42.11#48449
Dec 18 12:08:54 NS02 named[457]: zone projet.intranet/IN: transferred serial
2020121801
Dec 18 12:08:54 NS02 named[457]: transfer of 'projet.intranet/IN' from
192.168.42.10#53: Transfer status: success
Dec 18 12:08:54 NS02 named[457]: transfer of 'projet.intranet/IN' from
192.168.42.10#53: Transfer completed: 1 messages, 14 records, 358 bytes, 0.001
secs (358000 bytes/sec)
root@NS02:~# dig @127.0.0.1 client02.projet.intranet +short
192.168.42.102
```

Configuration d'apache

On créé l'arborescence pour les sites :

```
root@apache:/var/www# mkdir -p rh/salaire
root@apache:/var/www# mkdir -p site_epsi/admin
root@apache:/var/www# touch rh/index.html
root@apache:/var/www# touch site_epsi/index.html
root@apache:/var/www# chown www-data:www-data -R  */*
```

On génère les fichiers de conf apache pour les deux sites :

```
www_epsi.conf
<VirtualHost *:80>
        ServerName www.epsi.projet.intranet
        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/site_epsi

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
#--------
rh_epsi.conf

<VirtualHost *:80>
        ServerName rh.epsi.projet.intranet
        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/rh

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

On les active :

```
root@apache:/etc/apache2/sites-available# a2ensite www_epsi.conf
Enabling site www_epsi.
To activate the new configuration, you need to run:
  systemctl reload apache2
root@apache:/etc/apache2/sites-available# a2ensite rh_epsi.conf
Enabling site rh_epsi.
To activate the new configuration, you need to run:
  systemctl reload apache2
root@apache:/etc/apache2/sites-available# systemctl reload apache2
```

On vérifie l'accès aux sites :

```
root@client01:~# curl rh.epsi.projet.intranet
<h1> rh.epsi.projet.intranet </h1>
root@client01:~# curl www.epsi.projet.intranet
<h1>www.epsi.projet.intranet</h1>
```

Création d'un vhost par défaut

On modifie 000-default.conf en y rajoutant ce paramètre :

```
        <Location />
                Deny from all
        </Location>
```

On vérifie le bon fonctionnement :

```
root@client01:~# curl apache.projet.intranet
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>403 Forbidden</title>
</head><body>
<h1>Forbidden</h1>
<p>You don't have permission to access this resource.</p>
<hr>
<address>Apache/2.4.38 (Debian) Server at apache.projet.intranet Port
80</address>
</body></html>
root@client01:~# curl apache.projet.intranet/index.html
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>403 Forbidden</title>
</head><body>
<h1>Forbidden</h1>
<p>You don't have permission to access this resource.</p>
<hr>
<address>Apache/2.4.38 (Debian) Server at apache.projet.intranet Port
80</address>
</body></html>
root@client01:~# curl apache.projet.intranet/test/index.html
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>403 Forbidden</title>
</head><body>
```

```
<h1>Forbidden</h1>
<p>You don't have permission to access this resource.</p>
<hr>
<address>Apache/2.4.38 (Debian) Server at apache.projet.intranet Port
80</address>
</body></html>
root@client01:~# curl www.epsi.projet.intranet
<h1>www.epsi.projet.intranet</h1>
```

Configuration des logs

On modifie apache2.conf et on rajoute :

```
%{ms}T
```

Au format de log combined puis on recharge apache2 et on vérifie les logs :

```
#---En millisecondes
192.168.42.101 - - [18/Dec/2020:12:36:50 +0100] "GET /test/index.html HTTP/1.1"
403 448 "-" "curl/7.64.0" "Generated:0"
192.168.42.101 - - [18/Dec/2020:12:37:28 +0100] "GET /test/index.html HTTP/1.1"
403 448 "-" "curl/7.64.0" "Generated:0"
#--- en microsecondes
192.168.42.101 - - [18/Dec/2020:12:37:41 +0100] "GET /test/index.html HTTP/1.1"
403 448 "-" "curl/7.64.0" "Generated: 356 "
192.168.42.101 - - [18/Dec/2020:12:37:45 +0100] "GET /help HTTP/1.1" 404 447 "-"
"curl/7.64.0" "Generated: 325 "
192.168.42.101 - - [18/Dec/2020:12:37:47 +0100] "GET / HTTP/1.1" 200 261 "-"
"curl/7.64.0" "Generated: 336 "
```

On peut observer que le temps en millisecondes est de 0, en le modifiant pour afficher les microsecondes, on peut observer que le temps est inférieur à 0.5ms et donc arrondi à 0 par apache à l'affichage des logs.

Configuration de la rotation des logs

On modifie /etc/logrotate.d/apache2

```
/var/log/apache2/*.log {
    daily
    missingok
->  rotate 366
    compress
    delaycompress
    notifempty
->  dateformat .%Y-%m-%d
    create 640 root adm
    sharedscripts
    postrotate
                if invoke-rc.d apache2 status > /dev/null 2>&1; then \
                    invoke-rc.d apache2 reload > /dev/null 2>&1; \
                fi;
    endscript
    prerotate
```

```
        if [ -d /etc/logrotate.d/httpd-prerotate ]; then \
            run-parts /etc/logrotate.d/httpd-prerotate; \
        fi; \
    endscript
}
```

On ajoute ainsi l'horodatage et la rétention sur 366 jours des logs

Contrôle d'accès

On souhaite autoriser client2 à accéder à admin mais client01 dans le ficher www_epsi.conf on ajoute :

```
        <Location /admin>
                Require ip 192.168.42.102
        </Location>
```

On observe alors que client2 a bien l'accès et pas client01 :

```
root@client01:~# curl www.epsi.projet.intranet/admin/index.html
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>403 Forbidden</title>
</head><body>
<h1>Forbidden</h1>
<p>You don't have permission to access this resource.</p>
<hr>
<address>Apache/2.4.38 (Debian) Server at www.epsi.projet.intranet Port
80</address>
</body></html>

#--------------

root@client02:~# curl www.epsi.projet.intranet/admin/index.html
<h1> Admin </h1>
```

Controle d'acces par mdp

Création des utilisateurs :

```
root@apache:~# htpasswd -c /var/www/passwd/passwords rh
New password:
Re-type new password:
Adding password for user rh
root@apache:~# htpasswd /var/www/passwd/passwords salaire
New password:
Re-type new password:
Adding password for user salaire
```

Autorisation du .htaccess :

```
root@apache:/etc/apache2/sites-enabled# cat rh_epsi.conf
<VirtualHost *:80>
        #ServerName www.example.com
        ServerName rh.epsi.projet.intranet
        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/rh

        <Directory "/var/www/rh">
                AllowOverride All
        </Directory>

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

Configuration des .htacces :

```
root@apache:/var/www/rh# cat .htaccess
AuthType Basic
AuthName "Restricted Files"
# (Following line optional)
AuthBasicProvider file
AuthUserFile "/var/www/passwd/passwords"
Require user rh


root@apache:/var/www/rh/salaire# cat .htaccess
AuthType Basic
AuthName "Restricted Files"
# (Following line optional)
AuthBasicProvider file
AuthUserFile "/var/www/passwd/passwords"
<RequireAny>
        Require user salaire
        Require ip 192.168.42.101
</RequireAny>
```

Vérification des règles :

```
----------------------------------------Client1-----------------------------------
--------------
root@client01:/etc/ansible# curl -u rh:epsi2 -I rh.epsi.projet.intranet
HTTP/1.1 401 Unauthorized
Date: Fri, 29 Jan 2021 13:59:39 GMT
Server: Apache/2.4.38 (Debian)
WWW-Authenticate: Basic realm="Restricted Files"
Content-Type: text/html; charset=iso-8859-1

root@client01:/etc/ansible# curl -u rh:epsi2 -I rh.epsi.projet.intranet/salaire
HTTP/1.1 301 Moved Permanently
```

```
Date: Fri, 29 Jan 2021 13:59:46 GMT
Server: Apache/2.4.38 (Debian)
Location: http://rh.epsi.projet.intranet/salaire/
Content-Type: text/html; charset=iso-8859-1

root@client01:/etc/ansible# curl -u rh:epsi -I rh.epsi.projet.intranet
HTTP/1.1 200 OK
Date: Fri, 29 Jan 2021 13:59:54 GMT
Server: Apache/2.4.38 (Debian)
Last-Modified: Fri, 18 Dec 2020 11:22:42 GMT
ETag: "23-5b6bb53ad1753"
Accept-Ranges: bytes
Content-Length: 35
Content-Type: text/html
----------------------------------------Client2----------------------------------
--------------
root@client02:~# curl -u rh:epsi -I rh.epsi.projet.intranet
HTTP/1.1 200 OK
Date: Fri, 29 Jan 2021 14:00:30 GMT
Server: Apache/2.4.38 (Debian)
Last-Modified: Fri, 18 Dec 2020 11:22:42 GMT
ETag: "23-5b6bb53ad1753"
Accept-Ranges: bytes
Content-Length: 35
Content-Type: text/html

root@client02:~# curl -u rh:epsi -I rh.epsi.projet.intranet/salaire
HTTP/1.1 401 Unauthorized
Date: Fri, 29 Jan 2021 14:00:33 GMT
Server: Apache/2.4.38 (Debian)
WWW-Authenticate: Basic realm="Restricted Files"
Content-Type: text/html; charset=iso-8859-1

root@client02:~# curl -I rh.epsi.projet.intranet/salaire
HTTP/1.1 401 Unauthorized
Date: Fri, 29 Jan 2021 14:00:39 GMT
Server: Apache/2.4.38 (Debian)
WWW-Authenticate: Basic realm="Restricted Files"
Content-Type: text/html; charset=iso-8859-1

root@client02:~# curl -I rh.epsi.projet.intranet/salaire -u salaire:secret
HTTP/1.1 301 Moved Permanently
Date: Fri, 29 Jan 2021 14:00:45 GMT
Server: Apache/2.4.38 (Debian)
Location: http://rh.epsi.projet.intranet/salaire/
Content-Type: text/html; charset=iso-8859-1
```

Les paramètres fonctionnent donc bien

Ansible

Configuration du DNS :

Il est nécessaire de modifier la zone DNS pour rajouter le champ suivant :

```
apache2         IN      A       192.168.42.21
```

On recharge les zones et le transfère se fait vers le DNS secondaire.

Installation d'ansible

On effectue les commandes suivantes :

```
root@client01:~# apt update
root@client01:~# apt install python3-pip
root@client01:~# python3 -m pip install --upgrade pip
root@client01:~# pip3 install ansible==2.9.17
root@client01:~# ansible --version
ansible 2.9.17
  config file = None
  configured module search path = ['/root/.ansible/plugins/modules',
'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/local/lib/python3.7/dist-
packages/ansible
  executable location = /usr/local/bin/ansible
  python version = 3.7.3 (default, Jul 25 2020, 13:03:44) [GCC 8.3.0]
```

Configuration SSH

Génération de la clef

```
root@client01:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:KVCTR8z5zirZRImwGHUtyPhKxfbrXMVJzIV2fvH/vr0 root@client01.projet.intranet
```

Copie de la clef sur les serveurs existants :

On enregistre l'identité :

```
root@client01:~# ssh-agent bash
root@client01:~# ssh-add .ssh/id_rsa
Enter passphrase for .ssh/id_rsa:
Identity added: .ssh/id_rsa (root@client01.projet.intranet)
```

Création du fichier d'inventaire

Création du fichier de configuration ansible

L'installation via pip ne créant pas le dossier /etc/ansible ou sont stockés les fichiers par défaut, il est nécessaire de le créer. C'est la que seront stockés les fichiers ansible.cfg et hosts

On modifie certaines valeurs du fichier ansible.cfg

```
[defaults]
inventory = /etc/ansible/hosts
gathering = smart
private_key_file = /root/.ssh/id_rsa
```

On modifie le fichier hosts :

```
all:
  children:
    web:
      hosts:
        apache.projet.intranet:
        apache2.projet.intranet:
    dns:
      children:
          recurse:
            hosts:
              resolv.projet.intranet:
          dnslan:
            hosts:
              ns01.projet.intranet:
              ns02.projet.intranet:
```

On vérifie la lecture du fichier d'inventaire par ansible :

```
root@client01:/etc/ansible# ansible-inventory --graph
@all:
  |--@dns:
  |  |--@dnslan:
  |  |  |--ns01.projet.intranet ns02.projet.intranet
  |  |--@recurse:
  |  |  |--resolv.projet.intranet
  |--@ungrouped:
  |--@web:
  |  |--apache.projet.intranet apache2.projet.intranet
```

Premier Playbook :

On écrit le playbook ainsi :

```
---
- name: install pakcages and deploy ssh
  hosts: all
  remote_user: root
  tasks:
  - name: ensure packages are  at the latest version
    yum:
      name: "{{ item }}"
      state: latest
    loop:
      - tcpdump
      - vim
      - strace
      - sudo
  - name: create user epsiadmin
```

```
    user:
      name: epsiadmin
      state: present
  - name: copy to /etc/sudoers.d/epsiadmin
    copy:
      src: /etc/ansible/playbooks/files/epsiadmin
      dest: /etc/sudoers.d/epsiadmin
  - name: copy ssh key to user epsiadmin
    authorized_key:
      user: epsiadmin
      state: present
      key: "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDZ3FZglVT/7h+jKOR6ANo88NJKXnhwjgyl4/3DbxOgF3G41UBh
2eKi9CrbYJGlwtYDZxcgqnpZVltOIZmcHPxP4jxIaK3QJNvPA9ULUjbbjg7SrjUx569B6hzsMsIX2htk
u8VAiUMUWsrTMQUlixHNfothDijgI5j8u5DkBKVY6xTBaMOh9Jf3faRBZBI5CEoL1OA5pSMLzbDIKHcJ
P4khYufHUMvYY9TmSF3JN7hsGEw7aFJcWxahvXQXMn99+kHVLZDJ3cz8xylLzj81/4JtkYoNQAA0vrIE
EuZWEqoUkzqwwLS3AvWIcgXkxqdOZw7iGnkoBqucJwjRVj+XfTFn
root@client01.projet.intranet"
```

On le déploie en répétant l'action pour chacun des serveurs:

```
root@client01:/etc/ansible/playbooks# ansible-playbook deploy.yml -l
resolv.projet.intranet --ask-pass
SSH password:

PLAY [install pakcages and deploy ssh]
*********************************************

TASK [Gathering Facts]
*************************************************************
ok: [resolv.projet.intranet]

TASK [ensure packages are  at the latest version]
***********************************
ok: [resolv.projet.intranet] => (item=tcpdump)
changed: [resolv.projet.intranet] => (item=vim)
changed: [resolv.projet.intranet] => (item=strace)
changed: [resolv.projet.intranet] => (item=sudo)
[WARNING]: Updating cache and auto-installing missing dependency: python-apt

TASK [create user epsiadmin]
*************************************************************
changed: [resolv.projet.intranet]

TASK [copy to /etc/sudoers.d/epsiadmin]
*********************************************
changed: [resolv.projet.intranet]

TASK [copy ssh key to user epsiadmin]
*********************************************
changed: [resolv.projet.intranet]

PLAY RECAP
*************************************************************************
resolv.projet.intranet     : ok=5    changed=4    unreachable=0    failed=0
 skipped=0    rescued=0    ignored=0
```

On vérifie le fonctionnement du ssh:

```
root@client01:/etc/ansible/playbooks# ansible -m ping all
ns01.projet.intranet | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
resolv.projet.intranet | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
ns02.projet.intranet | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
apache2.projet.intranet | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
apache.projet.intranet | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
```

Comme on peut l'observer, tous les hôtes répondent.

Playbooks pour le serveur web

```
root@client01:/etc/ansible/playbooks# cat web.yml
---
- name: install pakcages and deploy ssh
  hosts: apache2.projet.intranet
  become: yes
  become_method: sudo

  tasks:
  - name: ensure packages are  at the latest version
    apt:
      name: "{{ item }}"
      state: latest
```

```
    loop:
      - apache2
      - php
      - mariadb-server

  - name: Create /var/www/site_php
    file:
      path: /var/www/site_php
      state: directory
      owner: www-data
      group: www-data

  - name: copy php site to /var/www/site_php
    copy:
      src: /etc/ansible/playbooks/files/index.php
      dest: /var/www/site_php
      owner: www-data
      group: www-data
      mode: '0400'

  - name:  Delete default conf
    file:
      path: /etc/apache2/sites-enabled/000-default.conf
      state: absent
    notify:
    - Restart apache

  - name: Copy whost
    copy:
      src: /etc/ansible/playbooks/files/001-site-php.conf
      dest: /etc/apache2/sites-enabled/001-site-php.conf
      owner: www-data
      group: www-data
    notify:
    - Restart apache

  handlers:
  - name: Restart apache
    service:
      name: apache2
      state: restarted
```

On vérifie le bon fonctionnement du playbook et de l'handler :

```
root@client01:/etc/ansible/playbooks# ansible-playbook web.yml

PLAY [install pakcages and deploy ssh]
*********************************************

TASK [Gathering Facts]
************************************************************
ok: [apache2.projet.intranet]
```

```
TASK [ensure packages are  at the latest version]
***********************************
ok: [apache2.projet.intranet] => (item=apache2)
ok: [apache2.projet.intranet] => (item=php)
ok: [apache2.projet.intranet] => (item=mariadb-server)

TASK [Create /var/www/site_php]
*****************************************************
ok: [apache2.projet.intranet]

TASK [copy php site to /var/www/site_php]
*******************************************
ok: [apache2.projet.intranet]

TASK [Delete default conf]
*****************************************************
ok: [apache2.projet.intranet]

TASK [Copy whost]
*************************************************************************
changed: [apache2.projet.intranet]

RUNNING HANDLER [Restart apache]
*****************************************************
changed: [apache2.projet.intranet]

PLAY RECAP
*************************************************************************
apache2.projet.intranet    : ok=7    changed=2    unreachable=0    failed=0
 skipped=0    rescued=0    ignored=0

root@client01:/etc/ansible/playbooks# ansible-playbook web.yml

PLAY [install pakcages and deploy ssh]
*********************************************

TASK [Gathering Facts]
*************************************************************
ok: [apache2.projet.intranet]

TASK [ensure packages are  at the latest version]
***********************************
ok: [apache2.projet.intranet] => (item=apache2)
ok: [apache2.projet.intranet] => (item=php)
ok: [apache2.projet.intranet] => (item=mariadb-server)

TASK [Create /var/www/site_php]
*****************************************************
ok: [apache2.projet.intranet]

TASK [copy php site to /var/www/site_php]
*******************************************
ok: [apache2.projet.intranet]

TASK [Delete default conf]
*****************************************************
ok: [apache2.projet.intranet]
```

```
TASK [Copy whost]
*********************************************************************
ok: [apache2.projet.intranet]

PLAY RECAP
*********************************************************************************
apache2.projet.intranet    : ok=6    changed=0    unreachable=0    failed=0
 skipped=0    rescued=0    ignored=0
```

L'Handler semble bien fonctionner.

Vérifions le fonctionnement du site web :

```
root@client01:/home/epsiadmin/.ssh# curl apache2.projet.intranet
Bonjour l'EPSI<br/>
```

Le site web fonctionne bien.

Playbook base de données

```
---
- name: install pakcages and deploy ssh
  hosts: web
  become: yes
  become_method: sudo

  tasks:

  - name: create db01
    community.mysql.mysql_db:
      name: db01
      state: present
  - name: Template a file to /etc/file.conf
    template:
      src: /etc/ansible/playbooks/files/001-site-php.j2
      dest: /etc/apache2/sites-enabled/001-site-php.conf
      owner: www-data
      group: www-data
    notify:
    - Restart apache

  handlers:
  - name: Restart apache
    service:
      name: apache2
      state: restarted
```

Le fichier template est le suivant :

```
<VirtualHost *:80>
        ServerName apache02.projet.intranet
        DocumentRoot {{ direct_name | default('/var/www/site_php') }}
        <Directory {{ direct_name | default('/war/www/site_php') }}>
                Require all granted
                DirectoryIndex index.php
        </Directory>
        CustomLog {{ log_folder | default('${APACHE_LOG_DIR}/access.log') }} {{
log_type | default('combined') }}
</VirtualHost>
```

On a choisi de faire ces variables modifiables :

```
- direct_name
- log_folder
- log_type
```

Ainsi l'appel suivant :

```
ansible-playbook db.yml -l apache2.projet.intranet -e
"direct_name=/var/www/http/epsi log_folder=/var/log/apache_espi log_type=common"
```

Créera ce fichier :

```
<VirtualHost *:80>
        ServerName apache02.projet.intranet
        DocumentRoot var/www/http/epsi
        <Directory /var/www/http/epsi>
                Require all granted
                DirectoryIndex index.php
        </Directory>
        CustomLog /var/log/apache_espi common
</VirtualHost>
```

Bonus Stage 1

```
---
- name: install pakcages and deploy ssh
  hosts: all
  become: yes
  become_method: sudo

  tasks:

  - name: Disable root login over SSH
    lineinfile:
      dest: /etc/ssh/sshd_config
      regexp: "^PermitRootLogin"
      line: "PermitRootLogin no"
```

```
      state: present
    notify:
      - restart sshd

  handlers:
  - name: restart sshd
    service:
      name: sshd
      state: restarted
```

Bonus Stage 2

Créaton d'un role :

```
root@client01:/etc/ansible/playbooks# ansible-galaxy init stage-2
- Role stage-2 was created successfull
```

Modification du role :

Fichier tasks/main.yml :

```
---
# tasks file for stage-2

- name: import web
  include: web.yml
```

Fichier tasks/web.yml :

```
- name: Install apache
  apt:
    name: apache2
    state: present

- name: Template a file to /etc/file.conf
  template:
    src: 001-site-php.j2
    dest: /etc/apache2/sites-enabled/001-site-php.conf
    owner: www-data
    group: www-data
  notify:
  - Restart apache

- name: Template a file to /etc/apache2/ports.conf
  template:
    src: ports.j2
    dest: /etc/apache2/ports.conf
    owner: www-data
    group: www-data
  notify:
  - Restart apache
```

Fichier handlers/main.yml :

```
---
# handlers file for stage-2
- name: Restart apache
  service:
    name: apache2
    state: restarted
```

Fichier templates/001-site-php.j2 :

```
<VirtualHost *:80>
        ServerName apache02.projet.intranet
        DocumentRoot {{ direct_name | default('/var/www/site_php') }}
        <Directory {{ direct_name | default('/war/www/site_php') }}>
                Require all granted
                DirectoryIndex index.php
        </Directory>
        CustomLog {{ log_folder | default('${APACHE_LOG_DIR}/access.log') }} {{
log_type | default('combined') }}
</VirtualHost>
```

Fichier templates/ports.j2 :

```
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf


{% for line in ports %}
listen {{ line }}
{% endfor %}

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Fichiers de configurations

L'ensemble des fichiers configurations sont retrouvables à l'adresse https://github.com/maeltur/linux