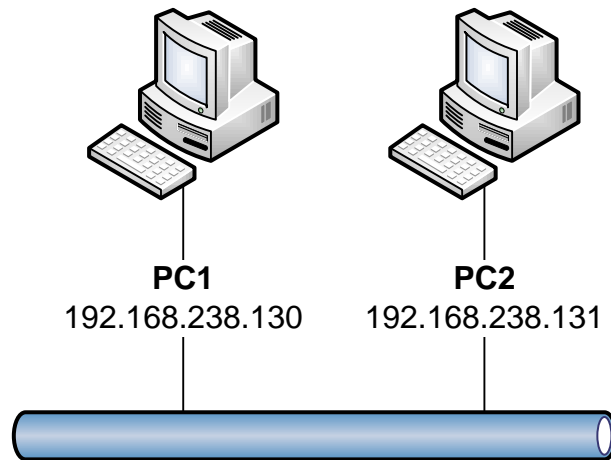


Méthode 1 : Mise en place IPSEC



Installation des outils « ipsec-tools » et « racoon » via les paquets ubuntu :

```
sudo -s  
apt-get install ipsec-tools  
apt-get install racoon
```

Il faut ensuite charger manuellement les modules noyau nécessaires :

```
modprobe esp4  
modprobe ah4  
modprobe ipcomp
```

Vous faut aussi configurer le chargement permanent, en modifiant le fichier « /etc/module » :

```
vim /etc/modules  
  
# Ajout des modules  
  
esp4  
  
ah4  
  
ipcomp
```

Configuration IPSEC du PC1 pour une authentification par clé partagée

Il faut modifier le fichier « /etc/racoon/setkey.conf » du PC1 comme suit :

Attention : Les retours à la ligne doivent être respectés !

```
#!/usr/sbin/setkey -f

# On efface les politiques de sécurité

# Flush the Security Association Database (SAD)

# And the Security Policy Database (SPD)

flush;

spdflush;

# Attention: Utiliser vos propres clés celle-ci n'est qu'un exemple !

# AH SAs using 128 bit long keys

add 192.168.238.130 192.168.238.131 ah 0x200 -m transport -A hmac-md5
0xc0291ff014dccdd03874d9e8e4cdf3e6;

add 192.168.238.131 192.168.238.130 ah 0x300 -m transport -A hmac-md5
0x96358c90783bbfa3d7b196ceabe0536b;

# ESP SAs using 192 bit long keys (168 + 24 parity)

add 192.168.238.130 192.168.238.131 esp 0x201 -m transport -E 3des-cbc
0x7aeaca3f87d060a12f4a4487d5a5c3355920fae69a96c831;

add 192.168.238.131 192.168.238.130 esp 0x301 -m transport -E 3des-cbc
0xf6ddb555acfd9d77b03ea3843f2653255afe8eb5573965df;

# Politiques de sécurité

spdadd 192.168.238.130 192.168.238.131 any -P out ipsec

    esp/transport//require

    ah/transport//require;

spdadd 192.168.238.131 192.168.238.130 any -P in ipsec

    esp/transport//require

    ah/transport//require;
```

Mettre à jour les droits :

```
chmod 600 /etc/racoon/setkey.conf
```

Lancer la mise à jour :

```
setkey -f /etc/racoon/setkey.conf
```

Vérifier les modifications :

```
setkey -D
```

```
setkey -DP
```

On sauvegarde les paramètres pour chaque redémarrage dans le fichier « /etc/rc.local » :

```
#!/bin/sh -e
```

```
# rc.local
```

```
# Force IPSEC dès le démarrage
```

```
setkey -f /etc/racoon/setkey.conf
```

```
exit 0
```

Configuration IPSEC du PC2 pour une authentification par clé partagée

Il faut suivre la même méthode que pour le PC1, mais le fichier de configuration « /etc/racoon/setkey.conf » devient :

```
#!/usr/sbin/setkey -f
```

```
# On efface les politiques de sécurité
```

```
# Flush the Security Association Database (SAD)
```

```
# And the Security Policy Database (SPD)
```

```
flush;
```

```
spdflush;
```

```
# Attention: Utiliser vos propres clés celle-ci n'est qu'un exemple !
```

```
# AH SAs using 128 bit long keys
```

```
add 192.168.238.130 192.168.238.131 ah 0x200 -m transport -A hmac-md5
```

```
0xc0291ff014dccdd03874d9e8e4cdf3e6;
```

```
add 192.168.238.131 192.168.238.130 ah 0x300 -m transport -A hmac-md5
```

```
0x96358c90783bbfa3d7b196ceabe0536b;
```

```
# ESP SAs using 192 bit long keys (168 + 24 parity)
```

```
add 192.168.238.130 192.168.238.131 esp 0x201 -m transport -E 3des-cbc
```

```
0x7aeaca3f87d060a12f4a4487d5a5c3355920fae69a96c831;
```

```

add 192.168.238.131 192.168.238.130 esp 0x301 -m transport -E 3des-cbc
0xf6ddb555acfd9d77b03ea3843f2653255afe8eb5573965df;

# Security policies

spdadd 192.168.238.130 192.168.238.131 any -P in ipsec

    esp/transport//require

    ah/transport//require;

spdadd 192.168.238.131 192.168.238.130 any -P out ipsec

    esp/transport//require

    ah/transport//require;

```

Lancer la mise à jour :

```
setkey -f /etc/racoon/setkey.conf
```

Vérifier les modifications :

```
setkey -D
setkey -DP
```

On sauvegarde les paramètres pour chaque redémarrage dans le fichier « /etc/rc.local » :

```

#!/bin/sh -e

# rc.local

# Force IPSEC dès le démarrage

setkey -f /etc/racoon/setkey.conf

exit 0

```

Maintenant on peut tester :

Sur PC2

```

tcpdump -i eth1 host 192.168.238.131 and 192.168.238.130 > /dev/pts/0

ping 192.168.238.130

```

Sur PC1

```

tcpdump -i eth1 host 192.168.238.130 and 192.168.238.131 > /dev/pts/0

ping 192.168.238.131

```

Résultats

```
PC1:~# ping 192.168.238.131
```

```

PING 192.168.238.131 (192.168.238.131) 56(84) bytes of data.

19:38:13.422427 IP 192.168.238.130 > 192.168.238.131: AH(spi=0x0582a70f,seq=0x35):
ESP(spi=0x0635e817,seq=0x35), length 100

64 bytes from 192.168.238.131: icmp_seq=1 ttl=64 time=2.23 ms

19:38:13.424239 IP 192.168.238.131 > 192.168.238.130: AH(spi=0x06ba7da2,seq=0x35):
ESP(spi=0x04669426,seq=0x35), length 100

19:38:14.425512 IP 192.168.238.130 > 192.168.238.131: AH(spi=0x0582a70f,seq=0x36):
ESP(spi=0x0635e817,seq=0x36), length 100

```

PC2:~# ping 192.168.238.130

```

PING 192.168.238.130 (192.168.238.130) 56(84) bytes of data.

19:37:21.561815 IP 192.168.238.131 > 192.168.238.130: AH(spi=0x06ba7da2,seq=0x30):
ESP(spi=0x04669426,seq=0x30), length 100

64 bytes from 192.168.238.130: icmp_seq=1 ttl=64 time=3.89 ms

19:37:21.565499 IP 192.168.238.130 > 192.168.238.131: AH(spi=0x0582a70f,seq=0x30):
ESP(spi=0x0635e817,seq=0x30), length 100

19:37:22.562545 IP 192.168.238.131 > 192.168.238.130: AH(spi=0x06ba7da2,seq=0x31):
ESP(spi=0x04669426,seq=0x31), length 100

```

Méthode 2 : Génération des certificats X 509

Installation de OpenSSL via les paquets ubuntu

```

sudo -s

apt-get install openssl

```

Générons l'autorité de certification par le script perl fourni lors de l'installation précédente

```

PC1:~# /usr/lib/ssl/misc/CA.pl -newca

CA certificate filename (or enter to create)

Making CA certificate ...

Generating a 1024 bit RSA private key

.....++++++

.....++++++

writing new private key to './demoCA/private/cakey.pem'

Enter PEM pass phrase:          MotDePasse

```

Verifying - Enter PEM pass phrase: **MotDePasse**

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:FR

State or Province Name (full name) [Some-State]:ILE-DE-FRANCE

Locality Name (eg, city) []:PARIS

Organization Name (eg, company) [Internet Widgits Pty Ltd]:ECE

Organizational Unit Name (eg, section) []:PFE

Common Name (eg, YOUR name) []:CAS

Email Address []:pfe@ece.fr

Please enter the following 'extra' attributes

to be sent with your certificate request

A challenge password []:CALYPSO

An optional company name []:ALCINOOS

Using configuration from /usr/lib/ssl/openssl.cnf

Enter pass phrase for ./demoCA/private/cakey.pem:

Check that the request matches the signature

Signature ok

Certificate Details:

Serial Number:

dd:a6:e3:ce:59:a8:5c:80

Validity

Not Before: Jan 8 12:24:18 2008 GMT

Not After : Jan 7 12:24:18 2011 GMT

Subject:

countryName = FR

stateOrProvinceName = ILE-DE-FRANCE

organizationName = ECE

organizationalUnitName = PFE

commonName = CAS

emailAddress = pfe@ece.fr

X509v3 extensions:

X509v3 Subject Key Identifier:

40:CB:19:7E:00:F0:4F:2E:C0:EF:BB:D9:8B:EE:2E:32:71:67:9F:2E

X509v3 Authority Key Identifier:

keyid:40:CB:19:7E:00:F0:4F:2E:C0:EF:BB:D9:8B:EE:2E:32:71:67:9F:2E

DirName:/C=FR/ST=ILE-DE-FRANCE/O=ECE/OU=PFE/CN=CAS/emailAddress=pfe@ece.fr

serial:DD:A6:E3:CE:59:A8:5C:80

X509v3 Basic Constraints:

CA:TRUE

Certificate is to be certified until Jan 7 12:24:18 2011 GMT (1095 days)

Write out database with 1 new entries

Data Base Updated

Configuration du certificat du PC1

Génération du certificat pour le PC1 : (*)

```
PC1:~# /usr/lib/ssl/misc/CA.pl -newreq
```

```
Generating a 1024 bit RSA private key
```

```
.....++++++
```

```
.....++++++
```

```
writing new private key to 'newkey.pem'
```

```
Enter PEM pass phrase:
```

```
MotDePasse
```

Verifying - Enter PEM pass phrase: **MotDePasse**

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:FR

State or Province Name (full name) [Some-State]:ILE-DE-FRANCE

Locality Name (eg, city) []:PARIS

Organization Name (eg, company) [Internet Widgits Pty Ltd]:ECE

Organizational Unit Name (eg, section) []:PFE

Common Name (eg, YOUR name) []:ALCINOOS

Email Address []:pfe@ece.fr

Please enter the following 'extra' attributes

to be sent with your certificate request

A challenge password []:ULYSSE

An optional company name []:ALCINOOS

Request is in newreq.pem, private key is in newkey.pem

Il nous faut maintenant signer le certificat via notre autorité de certification pour PC1 (*)

PC1:~# /usr/lib/ssl/misc/CA.pl -sign

Using configuration from /usr/lib/ssl/openssl.cnf

Enter pass phrase for ./demoCA/private/cakey.pem: MotDePasse

Check that the request matches the signature

Signature ok

Certificate Details:

Serial Number:

dd:a6:e3:ce:59:a8:5c:81

Validity

Not Before: Jan 8 12:40:36 2008 GMT

Not After : Jan 7 12:40:36 2009 GMT

Subject:

countryName = FR

stateOrProvinceName = ILE-DE-FRANCE

localityName = PARIS

organizationName = ECE

organizationalUnitName = PFE

commonName = ALCINOOS

emailAddress = pfe@ece.fr

X509v3 extensions:

X509v3 Basic Constraints:

CA:FALSE

Netscape Comment:

OpenSSL Generated Certificate

X509v3 Subject Key Identifier:

51:47:3D:24:EA:1B:CC:50:32:95:FA:01:3B:99:BB:E1:0F:E8:D7:6E

X509v3 Authority Key Identifier:

keyid:40:CB:19:7E:00:F0:4F:2E:C0:EF:BB:D9:8B:EE:2E:32:71:67:9F:2E

Certificate is to be certified until Jan 7 12:40:36 2009 GMT (365 days)

Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y

Write out database with 1 new entries

Data Base Updated

Signed certificate is in newcert.pem

Voici les certificats générés :

```
PC1:~# ls
newcert.pem newkey.pem newreq.pem
```

On peut alors personnaliser les certificats en les renommant, puis on les insère dans le répertoire de racoon :

```
PC1:~# mv newcert.pem /etc/racoon/certs/IPSECcert.pem
PC1:~# mv newkey.pem /etc/racoon/certs/IPSECkey.pem
PC1:~# mv newreq.pem /etc/racoon/certs/IPSECreq.pem
```

Voici les résultats obtenus :

```
PC1:/etc/racoon/certs# ls
IPSECcert.pem IPSECkey.pem IPSECreq.pem
```

On copie le certificat de l'autorité de certification dans le répertoire de racoon :

```
PC1:~# cp ./demoCA/cacert.pem /etc/racoon/certs/
```

On génère la liste de révocation des certificats :

```
PC1:~# openssl ca -genctrl -out crl.pem
Using configuration from /usr/lib/ssl/openssl.cnf
Enter pass phrase for ./demoCA/private/akey.pem: MotDePasse
```

On copie la liste dans le répertoire racoon :

```
PC1:~# cp crl.pem /etc/racoon/certs/
```

Voici la liste finale des certificats :

```
PC1:/etc/racoon/certs/# ls
cacert.pem crl.pem IPSECcert.pem IPSECkey.pem IPSECreq.pem
```

Pour que OpenSSL trouve le certificat, il doit être renommé en utilisant son nom version hashée :

```
PC1:/etc/racoon/certs# ln -s cacert.pem `openssl x509 -noout -hash < cacert.pem`.0
```

Il faut faire de même pour la CRL : liste des certificats révoqués :

```
PC1:/etc/racoon/certs# ln -s crl.pem `openssl x509 -noout -hash < cacert.pem`.r0
```

Voici la liste finale des certificats

```
PC1:/etc/racoon/certs# ls
843ea2ad.0 843ea2ad.r0 cacert.pem crl.pem IPSECcert.pem IPSECkey.pem IPSECreq.pem
```

Racoon ne sait pas déchiffrer les mots de passe :

```
PC1:/etc/racoon/certs# openssl rsa -in IPSECKey.pem -out IPSECKey.pem

Enter pass phrase for IPSECKey.pem: MotDePasse

writing RSA key
```

Configuration du certificat du PC2

Il faut installer OpenSSL, de la même manière que pour PC1.

Il faut générer le certificat sur l'autorité de certification du PC1, suivre les étapes précédentes marquées (*)

Il faut ensuite copier les certificats suivant sur le PC2 :

```
PC1:~# scp ./demoCA/cacert.pem ./newcert.pem ./newkey.pem ./newreq.pem ./crl.pem
root@192.168.238.131:/etc/racoon/certs/
```

On peut alors personnaliser les certificats en les renommant, puis on les insère dans le répertoire de racoon :

```
PC2:~# mv newcert.pem /etc/racoon/certs/IPSECcert.pem
PC2:~# mv newkey.pem /etc/racoon/certs/IPSECKey.pem
PC2:~# mv newreq.pem /etc/racoon/certs/IPSECreq.pem
PC2:~# mv cacert.pem /etc/racoon/certs/cacert.pem
PC2:~# mv crl.pem /etc/racoon/certs/crl.pem
```

Pour que OpenSSL trouve le certificat, il doit être renommé en utilisant son nom version hashée :

```
PC2:/etc/racoon/certs# ln -s cacert.pem `openssl x509 -noout -hash < cacert.pem`.0
```

Il faut faire de même pour la CRL : liste des certificats révoqués :

```
PC2:/etc/racoon/certs# ln -s crl.pem `openssl x509 -noout -hash < cacert.pem`.r0
```

Racoon ne sait pas déchiffrer les mots de passe :

```
PC2:/etc/racoon/certs# openssl rsa -in IPSECKey.pem -out IPSECKey.pem

Enter pass phrase for IPSECKey.pem: MotDePasse

writing RSA key
```

Les certificats étant mis en place, maintenant il nous faut modifier les paramètres d'IPSec

Configuration IPSEC du PC1 pour une authentification par certificat

Il faut modifier le fichier « /etc/racoon/setkey.conf » du PC1 comme suit :

```
#!/usr/sbin/setkey -f
# On efface les politiques de sécurité
# Flush the Security Association Database (SAD)
# And the Security Policy Database (SPD)
flush;
spdflush;
# Politiques de sécurité
spdadd 192.168.238.130 192.168.238.131 any -P out ipsec
    esp/transport//require
    ah/transport//require;
spdadd 192.168.238.131 192.168.238.130 any -P in ipsec
    esp/transport//require
    ah/transport//require;
```

Il faut modifier le fichier de configuration de /etc/racoon/racoon.conf :

```
path certificate "/etc/racoon/certs";
remote 192.168.238.131 {
    exchange_mode main;
    certificate_type x509 "IPSECCert.pem" "IPSECkey.pem";
    verify_cert on;
    my_identifier asn1dn;
    peers_identifier asn1dn;
    proposal {
        encryption_algorithm 3des;
        hash_algorithm md5;
        authentication_method rsasig;
        dh_group modp1024;
    }
}
```

```

}
sainfo anonymous {
    pfs_group modp768;
    encryption_algorithm 3des;
    authentication_algorithm hmac_md5;
    compression_algorithm deflate;
}

```

Configuration IPSEC du PC2 pour une authentification par certificat

Il faut modifier le fichier « /etc/racoon/setkey.conf » du PC2 comme suit :

```

#!/usr/sbin/setkey -f
# On efface les politiques de sécurité
# Flush the Security Association Database (SAD)
# And the Security Policy Database (SPD)
flush;
spdflush;
# Security policies
spdadd 192.168.238.130 192.168.238.131 any -P in ipsec
    esp/transport//require
    ah/transport//require;
spdadd 192.168.238.131 192.168.238.130 any -P out ipsec
    esp/transport//require
    ah/transport//require;

```

Il faut modifier le fichier de configuration de /etc/racoon/racoon.conf :

```

path certificate "/etc/racoon/certs";
remote 192.168.238.130 {
    exchange_mode main;
    certificate_type x509 "IPSECcert.pem" "IPSECkey.pem";
    verify_cert on;
}

```

```

my_identifier asn1dn;

peers_identifier asn1dn;

proposal {
    encryption_algorithm 3des;

    hash_algorithm md5;

    authentication_method rsasig;

    dh_group modp1024;
}
}

sainfo anonymous {
    pfs_group modp768;

    encryption_algorithm 3des;

    authentication_algorithm hmac_md5;

    compression_algorithm deflate;
}

```

On redémarre la machine...

On peut alors tester :

Attention le premier Ping ne fonctionne jamais car il sert à initialiser l'échange des clés, la seconde sera la bonne !

Sur PC2

```

tcpdump -i eth1 host 192.168.238.131 and 192.168.238.130 > /dev/pts/0

ping 192.168.238.130

```

Sur PC1

```

tcpdump -i eth1 host 192.168.238.130 and 192.168.238.131 > /dev/pts/0

ping 192.168.238.131

```

Résultats

```

PC1:~# ping 192.168.238.131

PING 192.168.238.131 (192.168.238.131) 56(84) bytes of data.
19:38:13.422427 IP 192.168.238.130 > 192.168.238.131: AH(spi=0x0582a70f,seq=0x35):
ESP(spi=0x0635e817,seq=0x35), length 100

```

64 bytes from 192.168.238.131: icmp_seq=1 ttl=64 time=2.23 ms

19:38:13.424239 IP 192.168.238.131 > 192.168.238.130: AH(spi=0x06ba7da2,seq=0x35):
ESP(spi=0x04669426,seq=0x35), length 100

19:38:14.425512 IP 192.168.238.130 > 192.168.238.131: AH(spi=0x0582a70f,seq=0x36):
ESP(spi=0x0635e817,seq=0x36), length 100

PC2:~# ping 192.168.238.130

PING 192.168.238.130 (192.168.238.130) 56(84) bytes of data.

19:37:21.561815 IP 192.168.238.131 > 192.168.238.130: AH(spi=0x06ba7da2,seq=0x30):
ESP(spi=0x04669426,seq=0x30), length 100

64 bytes from 192.168.238.130: icmp_seq=1 ttl=64 time=3.89 ms

19:37:21.565499 IP 192.168.238.130 > 192.168.238.131: AH(spi=0x0582a70f,seq=0x30):
ESP(spi=0x0635e817,seq=0x30), length 100

19:37:22.562545 IP 192.168.238.131 > 192.168.238.130: AH(spi=0x06ba7da2,seq=0x31):
ESP(spi=0x04669426,seq=0x31), length 100

Webographie

| Lien | Description |
|---|---|
| Chapter 23. IPsec | Tutoriel IPsec avec clé partagée en anglais |
| IPSEC using Linux Kernel 2.6 | Tutoriel VPN IPsec en anglais |
| racoon.pdf | Tutoriel IPsec avec racoon en français |
| Linux Kernel 2.6 using KAME-tools | How to sur IPsec |
| IPsec : Présentation et Configuration | Evaluation d'IPsec |
| IPsec FAQ | Doc NetBSD sur IPsec |
| man racoon.conf | Manuel anglais sur racoon |