

Zeroshell _Lan-to-Lan_VPN_ IPSEC "for Dummies"

fonte: <http://www.zeroshell.net/forum/viewtopic.php?t=2057&highlight=ipsec>

si ringrazia [atavachron](#) per la guida iniziale senza la quale non avrei saputo dove sbattere la testa e tutti gli altri utenti del forum che hanno aggiunto nei commenti la loro esperienza.

Scopo della guida: aiutare gli utenti più o meno niubby come il sottoscritto che nonostante la guida di atavachron hanno impiegato circa 2/3 giorni di esperimenti per far funzionare il tutto correttamente.

Considerazioni personali: Fulvio mi hai illuminato il cammino, non smetteremo mai di complimentarci con te per il tuo lavoro superlativo. Quello delle vpn lan to lan su ipsec probabilmente è uno dei pochissimi nei di questo progetto. Le VPN con Openvpn, magari tra 2 firewall con sopra zeroshell funzionano benissimo e ci vuole si e no mezz'ora per farle funzionare a dovere, ma la possibilità che dall'altro lato del tunnel non ci sia uno Zeroshell è più che realistica e purtroppo bisogna fronteggiare anche questo problema. Immagino che Fulvio abbia altre cose a cui pensare, ma se un domani si riuscisse ad inserire questa funzionalità nell'interfaccia web di Zeroshell sarebbe una cosa meravigliosa (Mi riferisco qui anche ad utenti più bravi di me che abbiano la possibilità di aiutare Fulvio nell'impresa).

Considerazioni su questa guida: con quello che sto per scrivere non ho assolutamente intenzione di affermare che il mio sia il solo modo di fare questa cosa, ma vorrei solo essere di aiuto e dare il mio contributo ad u progetto così bello.

Mi permetto di riportare qui sotto lo schema tipo di una Lan di quelle che più o meno tutti noi ci troviamo a dover “sbrogliare”. E aggiungo che inizierei la configurazione della VPN con l’esempio pratico più facile a mio parere ovvero quello in cui Zeroshell è esposto a internet cioè: o possiede il suo bellissimo indirizzo ip pubblico oppure il traffico sull’unico ip pubblico disponibile nella rete arriva al router e viene reindirizzato o Nattato verso Zeroshell.

Ho trovato utile per me, lasciare sulla sinistra lo schema vuoto e compilare di volta in volta quello di destra con i dati della rete interessata.

MODELLO LAN:

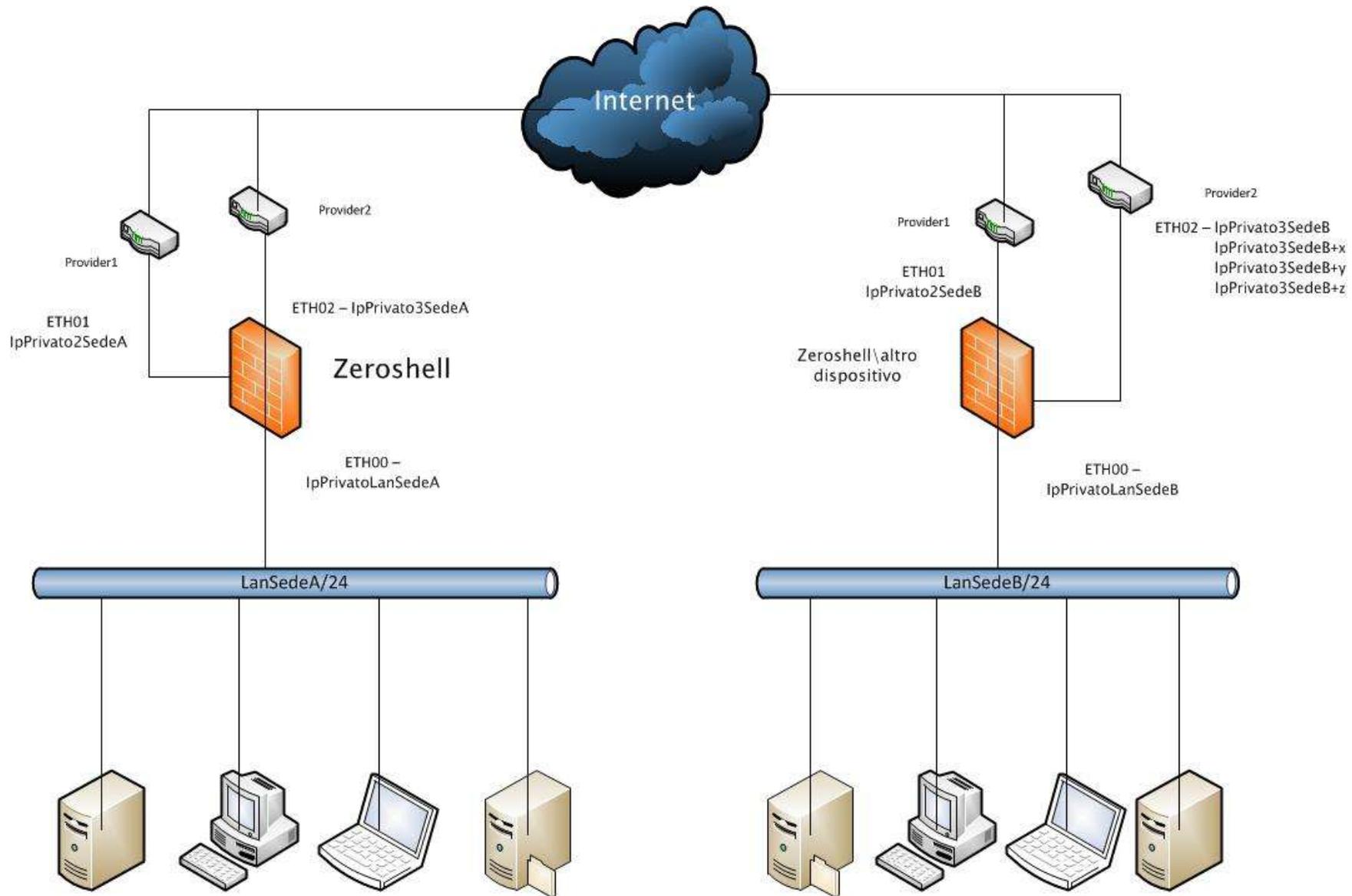
LanSedeA/24

|
| ETH00 IpPrivatoLanSedeA
ZeroShell
| ETH01 IpPrivato2SedeA\ **ipPubblicoSedeA**
|
| IpPrivato1SedeA
Router Adsl
| IpPubblicoSedeA
|
Internet
|
| IpPubblicoSedeB
Router Adsl
| IpPrivato1SedeB
|
| ETH01 IpPrivato2SedeB\ **ipPubblicoSedeB**
ZeroShell
| ETH00 IpPrivatoLanSedeB
|
LanSedeB/24

LanSedeA/24

|
| ETH00 IpPrivatoLanSedeA
ZeroShell
| ETH01 IpPrivato2SedeA\ **ipPubblicoSedeA**
|
| IpPrivato1SedeA
Router Adsl
| IpPubblicoSedeA
|
Internet
|
| IpPubblicoSedeB
Router Adsl
| IpPrivato1SedeB
|
| ETH01 IpPrivato2SedeB\ **ipPubblicoSedeB**
ZeroShell
| ETH00 IpPrivatoLanSedeB
|
LanSedeB/24

Ricordo immediatamente che in una situazione in cui Zeroshell è esposto e dunque **senza NAT-T**, **IpPrivato2SedeA** coincide con **IpPubblicoSedeA**, e questo vale anche per **IpPrivato2SedeB = IpPubblicoSedeB**.



Possiamo iniziare :

Sede A:

1. Creare una directory "racoon" in /Database/etc/
2. All'interno della directory appena creata /Database/etc/racoon/ Creare un file **pskey.conf** contenente la chiave condivisa PSK, con la sintassi IpPubblicoSedeB : chiavecondivisadaconservareenondivulgaretantoprimaopoilabucanocomunque
(ex: xxx.xxx.xxx.xxx : chiavecondivisa)
3. Sempre all'interno di /Database/etc/racoon Creare un file **racoon.conf** (e compilarlo come da diagramma di esempio con i valori interessati)
4. Creiamo l'ultimo file sempre in /Database/etc/racoon/ **setkey.conf** (e compilarlo come da diagramma di esempio con i valori interessati)
5. E per ultimo aggiungiamo in **Startup/Cron alla sezione PostBoot:**

Startup Script

```
setkey -f /Database/etc/racoon/setkey.conf  
route add -net LanSedeB/24 gw IpPrivatoLanSedeA  
racoon -f /Database/etc/racoon/racoon.conf
```

(per avere molte più informazioni per il DEBUG potete sostituire la terza riga con **racoon -d -F -f /Database/etc/racoon/racoon.conf**)

6. Assicurarsi che il file /Database/etc/racoon/pskey.conf abbia i permessi 600 dunque un bel chmod 600 /Database/etc/racoon/pskey.conf, perchè altrimenti il demone racoon inizia a fare casini
7. Dopo il passaggio 6 su entrambe le reti vi direi che molto probabilmente il tunnel verrà "stabilito" con successo ma che dire del traffico di dati fra le due sedi remote ? niente di niente... allora ecco cosa dobbiamo ancora fare:

In Startup/Cron, alla sezione NAT and Virtual Server

Translate outgoing connections for xxx.xxx.xxx.xxx/24 network

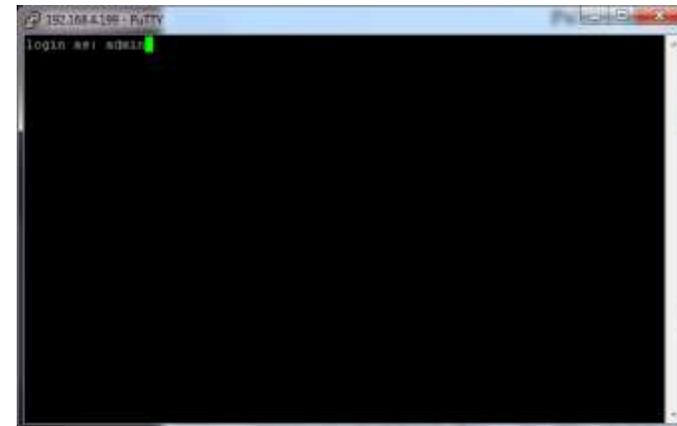
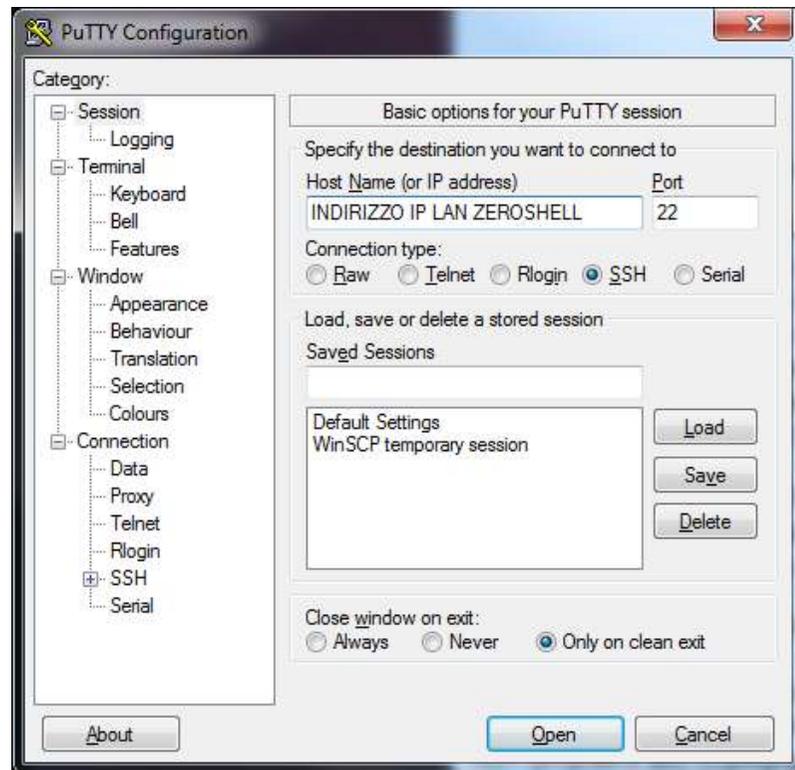
```
iptables -t nat -I POSTROUTING -o ETHXX -d reteprivataremota/24 -j ACCEPT
```

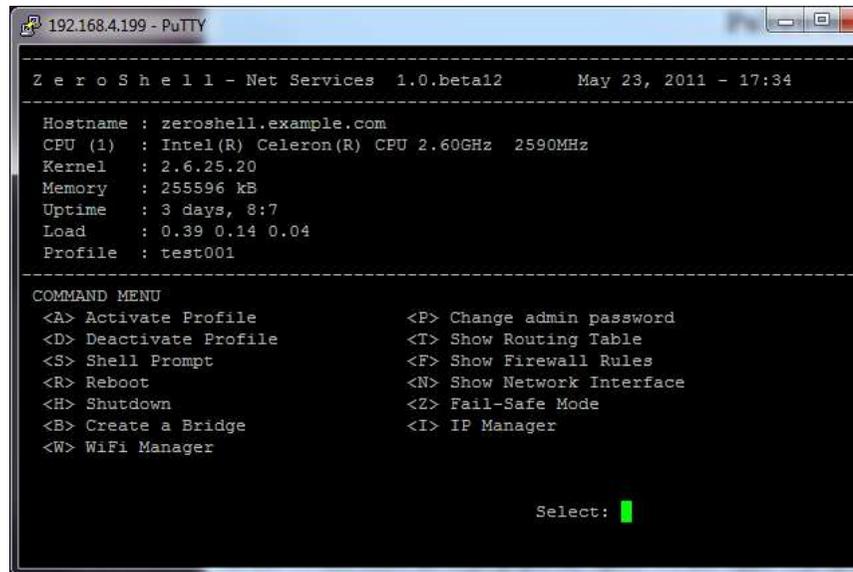
Ho preferito indicare l'interfaccia di rete di Zeroshell con ETHXX in quanto potremmo non necessariamente aver usato tutti la stessa configurazione oppure potremmo trovarci in casi particolari con più interfacce per poter "dialogare" con Gateway diversi.

Quello che conta è inserire l'interfaccia di rete corretta associata al nostro IP PUBBLICO

Bene ! Come le faccio tutte queste cose ?

Immaginiamo pure di stare utilizzando una versione qualsiasi del buon vecchio Windows di casa Microsoft, (anche se alcuni storceranno il naso)se qualcuno di quelli che stanno leggendo questa guida non conoscesse questo programmino (immagino pochi), vi segnalo che esiste un bellissimo client ssh di nome putty e lo trovate [qui](#)





```
192.168.4.199 - PuTTY
-----
Z e r o S h e l l - Net Services 1.0.beta12      May 23, 2011 - 17:34
-----
Hostname : zeroshell.example.com
CPU (1)  : Intel(R) Celeron(R) CPU 2.60GHz  2590MHz
Kernel   : 2.6.25.20
Memory   : 255596 kB
Uptime   : 3 days, 8:7
Load     : 0.39 0.14 0.04
Profile  : test001
-----
COMMAND MENU
<A> Activate Profile           <P> Change admin password
<D> Deactivate Profile        <T> Show Routing Table
<S> Shell Prompt              <F> Show Firewall Rules
<R> Reboot                    <N> Show Network Interface
<H> Shutdown                  <Z> Fail-Safe Mode
<B> Create a Bridge           <I> IP Manager
<W> WiFi Manager

                               Select: █
```

Figura 1 - Putty SSH

Immagino che in qualche modo la figura vi sia familiare !

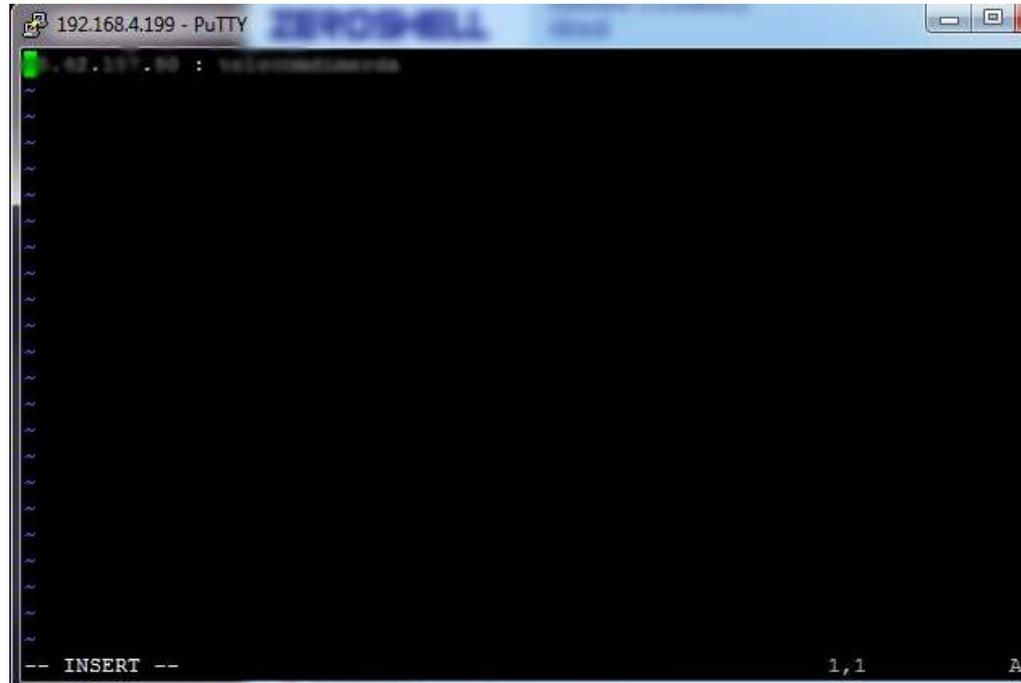
Una volta raggiunta la Shell del nostro Zeroshell cominciamo con un bel **mkdir /Database/etc/racoon**

Subito dopo possiamo utilizzare un Editor, molto familiare ai più per poter creare e modificare i file menzionati sopra. L'editor integrato in praticamente tutte le distribuzioni Unix based è **Vi**.

La sintassi è abbastanza semplice, per esempio: **vi /Database/etc/racoon/pskey.conf** per poter creare e modificare la nostra chiave di sicurezza. Quello che non è semplicissimo per chi usa vi per le prime volte è come salvare il file, come iniziare a scrivere nel file, come uscire dal file tralasciando le modifiche effettuate ecc ecc...

Bene, per completezza vi indico un paio di comandi utili per usare vi ai fini di questa guida e vi anticipo che, causa "elevata ignoranza" e poca confidenza con il caro vi, non mi sono nemmeno sognato di copiare/incollare la configurazione dei file più complessi come racoon.conf usando vi.

Allora: una volta entrati nel file con il comando qui sopra vi potete muovere tra le righe usando le frecce... ma come diavolo si fa a modificare il file ? Potete premere la **i** e avrete una schermata simile a questa:



Dopo aver effettuato correttamente le varie modifiche, premete **Esc** per uscire dalla modalità “inserimento” e premete **:wq** + Invio per salvare il file.

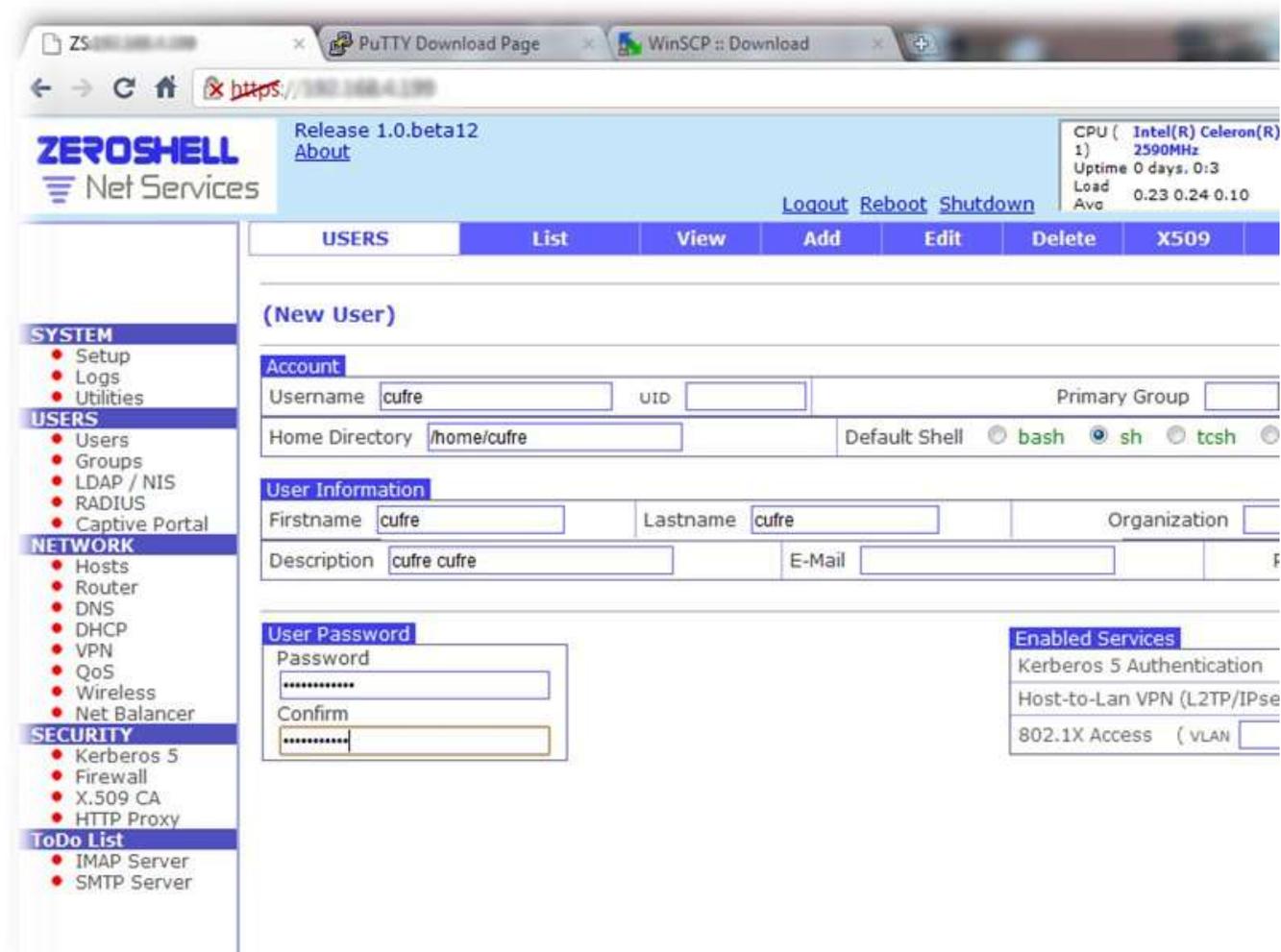
PS: se le modifiche che avete fatto non vi piacciono, dopo Esc, premete solo **:q** per uscire dal file senza salvarlo.

Come anticipato non mi sognerei mai di farvi editare tutti i file con il vi!, ma qual è la soluzione ?.

La soluzione è **abilitare il protocollo sftp** in ZeroShell ed usare un altro bellissimo client sftp come **WINSCP** che trovate [qui](#).in versione portable.

OPS: Fulvio non ci ha dato la possibilità di usare SFTP al volo... vi copio qui sotto i suggerimenti che ho trovato sul forum per poterlo abilitare nel modo più semplice secondo il mio modesto parere: (mi scuso anticipatamente ma non ho ancora ritrovato il link alla guida originale nel forum inglese).

1. Aggiungete un utente dalla GUI di Zeroshell... (exp: **cuFRE**)



2. Dalla shell, magari usando il nostro amato putty eseguite questo comando:
useradd cuFRE -p (anyword)
3. Questa volta dovete per forza usare vi, vi **/etc/ssh/sshd_config**, trovate la riga "AllowUsers admin" e modificatela aggiungendo il nome del vostro utente **AllowUsers admin cuFRE** (nel mio caso l'ho trovata alla riga 67, quindi anche per voi dovrebbe essere più o meno lì)

4. Sempre dall'interfaccia web di Zeroshell riavviate SSH
5. Provate un po' a vedere cosa succede con **WINSCP**
6. **Attenzione perché queste modifiche saranno valide solo fino al prossimo riavvio e poi dovrete rifarle** (qualcuno ha suggerito di inserire dei comandi sempre in /Cron/startup/postboot, ma anche rifare questi 4 passaggi ogni volta è davvero veloce)
7. A questo punto, creare e modificare i file del caso dovrebbe essere molto più semplice. !!!

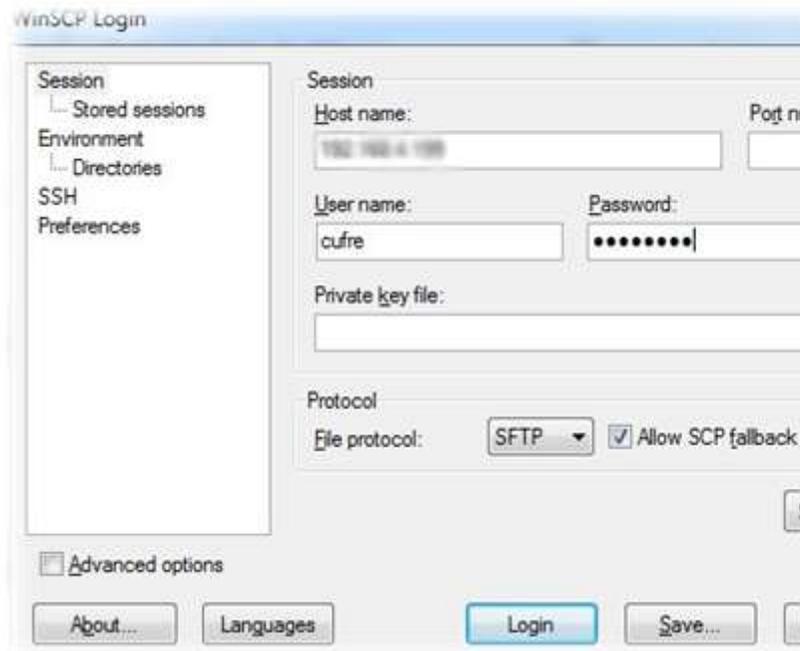


Figura 2 WINSCP

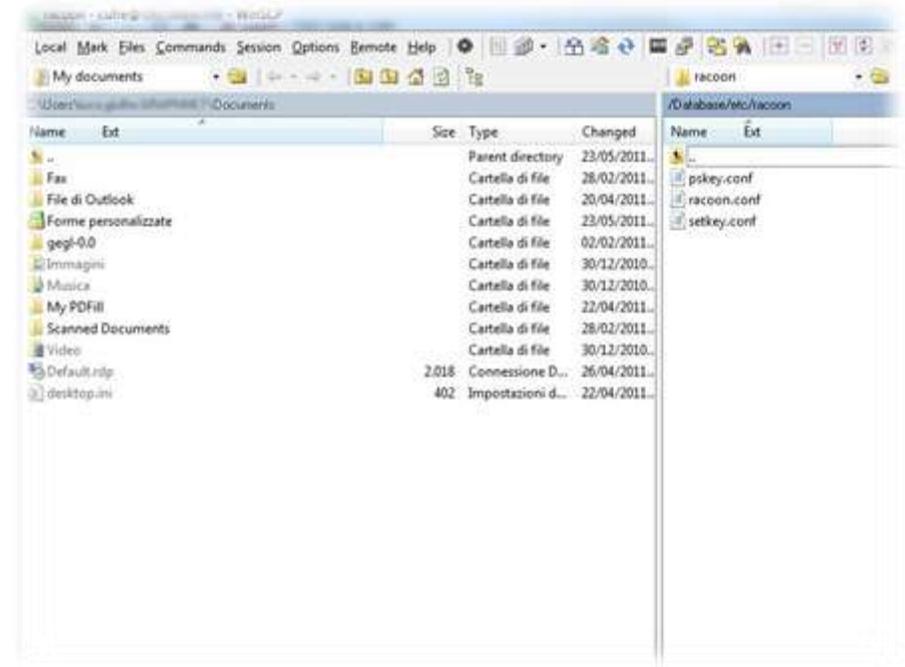


Figura 3 WINSCP

A questo punto non mi resta che inserirvi il codice per i 3 file:

File – SEDE A con Firewall direttamente esposto :

racoon.conf

```
path pre_shared_key "/Database/etc/racoon/pskey.conf";
listen {
isakmp IpPubblicoSedeA [500];
#isakmp_natt IpPubblicoSedeA [4500];
}
remote IpPubblicoSedeB {
exchange_mode aggressive,main;
my_identifier address IpPubblicoSedeA;
initial_contact off;
#nat_traversal on;
proposal {
encryption_algorithm 3des;
hash_algorithm md5;
authentication_method pre_shared_key;
dh_group 2;
}
}
sainfo address LanSedeA/24 any address LanSedeB/24 any
{
pfs_group 2;
encryption_algorithm 3des;
authentication_algorithm hmac_md5;
compression_algorithm deflate;
lifetime time 28800 sec;
}
```

setkey.conf

```
flush;  
spdflush;  
spdadd LanSedeB/24 LanSedeA/24 any -P in ipsec esp/tunnel/IpPubblicoSedeB-IpPubblicoSedeA/require;  
spdadd LanSedeA/24 LanSedeB/24 any -P out ipsec esp/tunnel/IpPubblicoSedeA-IpPubblicoSedeB /require;
```

Startup/Cron sezione PostBoot

```
# Startup Script  
setkey -f /Database/etc/racoon/setkey.conf  
route add -net LanSedeB /24 gw IpPrivatoLanSedeA  
racoon -d -F -f /Database/etc/racoon/racoon.conf
```

Startup/Cron sezione NAT and Virtual Server

```
# Translate outgoing connections for xxx.xxx.xxx.xxx/24 network  
iptables -t nat -I POSTROUTING -o ETHXX -d reteprivataremota/24 -j ACCEPT
```

File – SEDE B con Firewall direttamente esposto :

racoon.conf

```
path pre_shared_key "/Database/etc/racoon/pskey.conf";
listen {
isakmp IpPubblicoSedeB [500];
#isakmp_natt IpPubblicoSedeB [4500];
}
remote IpPubblicoSedeA {
exchange_mode aggressive,main;
my_identifier address IpPubblicoSedeB;
initial_contact off;
#nat_traversal on;
proposal {
encryption_algorithm 3des;
hash_algorithm md5;
authentication_method pre_shared_key;
dh_group 2;
}
}
sainfo address LanSedeB/24 any address LanSedeA/24 any
{
pfs_group 2;
encryption_algorithm 3des;
authentication_algorithm hmac_md5;
compression_algorithm deflate;
lifetime time 28800 sec;
}
```

setkey.conf

```
flush;  
spdflush;  
spdadd LanSedeA/24 LanSedeB/24 any -P in ipsec esp/tunnel/IpPubblicoSedeA-IpPubblicoSedeB/require;  
spdadd LanSedeB/24 LanSedeA/24 any -P out ipsec esp/tunnel/IpPubblicoSedeB-IpPubblicoSedeA /require;
```

Startup/Cron sezione PostBoot

```
# Startup Script  
setkey -f /Database/etc/racoon/setkey.conf  
route add -net LanSedeA /24 gw IpPrivatoLanSedeB  
racoon -d -F -f /Database/etc/racoon/racoon.conf
```

Startup/Cron sezione NAT and Virtual Server

```
# Translate outgoing connections for xxx.xxx.xxx.xxx/24 network  
iptables -t nat -I POSTROUTING -o ETHXX -d reteprivataremota/24 -j ACCEPT
```

Modulino – SEDE A – per compilare i file :

racoon.conf

```
path pre_shared_key "/Database/etc/racoon/pskey.conf";
listen {
isakmp IpPrivato2SedeA [500];
isakmp_natt IpPrivato2SedeA [4500];
}
remote IpPubblicoSedeB {
exchange_mode aggressive,main;
my_identifier address IpPrivato2SedeA;
initial_contact off;
nat_traversal on;
proposal {
encryption_algorithm 3des;
hash_algorithm md5;
authentication_method pre_shared_key;
dh_group 2;
}
}
sainfo address LanSedeA/24 any address LanSedeB/24 any
{
pfs_group 2;
encryption_algorithm 3des;
authentication_algorithm hmac_md5;
compression_algorithm deflate;
lifetime time 28800 sec;
}
```

racoon.conf

```
path pre_shared_key "/Database/etc/racoon/pskey.conf";
listen {
isakmp xx.xx.xx.x1 [500];
isakmp_natt xx.xx.xx.x1 [4500];
}
remote yy.yy.yy.y1 {
exchange_mode aggressive,main;
my_identifier address xx.xx.xx.x1;
initial_contact off;
nat_traversal on;
proposal {
encryption_algorithm 3des;
hash_algorithm md5;
authentication_method pre_shared_key;
dh_group 2;
}
}
sainfo address 192.168.100.0/24 any address 192.168.170.0/24 any
{
pfs_group 2;
encryption_algorithm 3des;
authentication_algorithm hmac_md5;
compression_algorithm deflate;
lifetime time 28800 sec;
}
```

setkey.conf

```
flush;  
spdflush;  
spdadd LanSedeB/24 LanSedeA/24 any -P in ipsec esp/tunnel/IpPubblicoSedeB-IpPrivato2SedeA/require;  
spdadd LanSedeA/24 LanSedeB/24 any -P out ipsec esp/tunnel/IpPrivato2SedeA-IpPubblicoSedeB/require;
```

setkey.conf

```
flush;  
spdflush;  
spdadd LanSedeB/24 LanSedeA/24 any -P in ipsec esp/tunnel/IpPubblicoSedeB-IpPrivato2SedeA/require;  
spdadd LanSedeA/24 LanSedeB/24 any -P out ipsec esp/tunnel/IpPrivato2SedeA-IpPubblicoSedeB/require;
```

Startup/Cron sezione PostBoot

```
# Startup Script  
setkey -f /Database/etc/racoon/setkey.conf  
route add -net LanSedeB/24 gw IpPrivatoLanSedeA  
racoon -f /Database/etc/racoon/racoon.conf
```

Startup/Cron sezione PostBoot

```
# Startup Script  
setkey -f /Database/etc/racoon/setkey.conf  
route add -net LanSedeB/24 gw IpPrivatoLanSedeA  
racoon -f /Database/etc/racoon/racoon.conf
```

Modulino – SEDE B – per compilare i file:

racoon.conf

```
path pre_shared_key "/Database/etc/racoon/pskey.conf";
listen {
isakmp IpPrivato2SedeB [500];
isakmp_natt IpPrivato2SedeB [4500];
}
remote IpPubblicoSedeA {
exchange_mode aggressive,main;
my_identifier address IpPrivato2SedeB;
initial_contact off;
nat_traversal on;
proposal {
encryption_algorithm 3des;
hash_algorithm md5;
authentication_method pre_shared_key;
dh_group 2;
}
}
sainfo address LanSedeB/24 any address LanSedeA/24 any
{
pfs_group 2;
encryption_algorithm 3des;
authentication_algorithm hmac_md5;
compression_algorithm deflate;
lifetime time 28800 sec;
}
```

racoon.conf

```
path pre_shared_key "/Database/etc/racoon/pskey.conf";
listen {
isakmp 85.42.157.90 [500];
isakmp_natt 85.42.157.90 [4500];
}
remote 213.255.43.254 {
exchange_mode aggressive,main;
my_identifier address 85.42.157.90;
initial_contact off;
nat_traversal on;
proposal {
encryption_algorithm 3des;
hash_algorithm md5;
authentication_method pre_shared_key;
dh_group 2;
}
}
sainfo address 192.168.1.0/24 any address 192.168.4.0/24 any
{
pfs_group 2;
encryption_algorithm 3des;
authentication_algorithm hmac_md5;
compression_algorithm deflate;
lifetime time 28800 sec;
}
```

setkey.conf

```
flush;  
spdflush;  
spdadd LanSedeA/24 LanSedeB/24 any -P in ipsec esp/tunnel/IpPubblicoSedeA-IpPrivato2SedeB/require;  
spdadd LanSedeB/24 LanSedeA/24 any -P out ipsec esp/tunnel/IpPrivato2SedeB-IpPubblicoSedeA/require;
```

setkey.conf

```
flush;  
spdflush;  
spdadd 192.168.4.0/24 192.168.1.0/24 any -P in ipsec esp/tunnel/213.255.43.254-85.42.157.90/require;  
spdadd 192.168.1.0/24 192.168.4.0/24 any -P out ipsec esp/tunnel/85.42.157.90-213.255.43.254/require;
```

Startup/Cron sezione PostBoot

```
# Startup Script  
setkey -f /Database/etc/racoon/setkey.conf  
route add -net LanSedeA/24 gw IpPrivatoLanSedeB  
racoon -f /Database/etc/racoon/racoon.conf
```

Startup/Cron sezione PostBoot

```
# Startup Script  
setkey -f /Database/etc/racoon/setkey.conf  
route add -net 192.168.4.0/24 gw 85.42.157.90  
racoon -f /Database/etc/racoon/racoon.conf
```

Modulino – SEDE A con Firewall direttamente esposto – per compilare i file:

racoon.conf

```
path pre_shared_key "/Database/etc/racoon/pskey.conf";
listen {
isakmp IpPubblicoSedeA [500];
#isakmp_natt IpPubblicoSedeA [4500];
}
remote IpPubblicoSedeB {
exchange_mode aggressive,main;
my_identifier address IpPubblicoSedeA;
initial_contact off;
#nat_traversal on;
proposal {
encryption_algorithm 3des;
hash_algorithm md5;
authentication_method pre_shared_key;
dh_group 2;
}
}
sainfo address LanSedeA/24 any address LanSedeB/24 any
{
pfs_group 2;
encryption_algorithm 3des;
authentication_algorithm hmac_md5;
compression_algorithm deflate;
lifetime time 28800 sec;
}
```

racoon.conf

```
path pre_shared_key "/Database/etc/racoon/pskey.conf";
listen {
isakmp 213.255.43.254 [500];
#isakmp_natt 213.255.43.254 [4500];
}
remote 85.142.157.90 {
exchange_mode aggressive,main;
my_identifier address 213.255.43.254;
initial_contact off;
#nat_traversal on;
proposal {
encryption_algorithm 3des;
hash_algorithm md5;
authentication_method pre_shared_key;
dh_group 2;
}
}
sainfo address 192.168.4.0/24 any address 192.168.1.0/24 any
{
pfs_group 2;
encryption_algorithm 3des;
authentication_algorithm hmac_md5;
compression_algorithm deflate;
lifetime time 28800 sec;
}
```

setkey.conf

```
flush;  
spdflush;  
spdadd LanSedeB/24 lanSedeA/24 any -P in ipsec esp/tunnel/IpPubblicoSedeB-IpPubblicoSedeA/require;  
spdadd LanSedeA/24 lanSedeB/24 any -P out ipsec esp/tunnel/IpPubblicoSedeA-IpPubblicoSedeB /require;
```

setkey.conf

```
flush;  
spdflush;  
spdadd 192.168.1.0/24 192.168.4.0/24 any -P in ipsec esp/tunnel/85.142.157.90-213.255.43.254/require;  
spdadd 192.168.4.0/24 192.168.1.0/24 any -P out ipsec esp/tunnel/213.255.43.254-85.142.157.90/require;
```

Startup/Cron sezione PostBoot

```
# Startup Script  
setkey -f /Database/etc/racoon/setkey.conf  
route add -net LanSedeB /24 gw IpPrivatoLanSedeA  
racoon -f /Database/etc/racoon/racoon.conf
```

Startup/Cron sezione PostBoot

```
# Startup Script  
setkey -f /Database/etc/racoon/setkey.conf  
route add -net 192.168.1.0/24 gw 192.168.4.199  
racoon -f /Database/etc/racoon/racoon.conf
```

Modulino – SEDE B con Firewall direttamente esposto – per compilare i file:

racoon.conf

```
path pre_shared_key "/Database/etc/racoon/pskey.conf";
listen {
isakmp IpPubblicoSedeB [500];
#isakmp_natt IpPubblicoSedeB [4500];
}
remote IpPubblicoSedeA {
exchange_mode aggressive,main;
my_identifier address IpPubblicoSedeB;
initial_contact off;
#nat_traversal on;
proposal {
encryption_algorithm 3des;
hash_algorithm md5;
authentication_method pre_shared_key;
dh_group 2;
}
}
sainfo address LanSedeB/24 any address LanSedeA/24 any
{
pfs_group 2;
encryption_algorithm 3des;
authentication_algorithm hmac_md5;
compression_algorithm deflate;
lifetime time 28800 sec;
}
```

racoon.conf

```
path pre_shared_key "/Database/etc/racoon/pskey.conf";
listen {
isakmp 85.142.157.90 [500];
#isakmp_natt 85.142.157.90 [4500];
}
remote 213.255.43.254 {
exchange_mode aggressive,main;
my_identifier address 85.142.157.90;
initial_contact off;
#nat_traversal on;
proposal {
encryption_algorithm 3des;
hash_algorithm md5;
authentication_method pre_shared_key;
dh_group 2;
}
}
sainfo address 192.168.1.0/24 any address 192.168.4.0/24 any
{
pfs_group 2;
encryption_algorithm 3des;
authentication_algorithm hmac_md5;
compression_algorithm deflate;
lifetime time 28800 sec;
}
```

setkey.conf

```
flush;  
spdflush;  
spdadd LanSedeA/24 lanSedeB/24 any -P in ipsec esp/tunnel/IpPubblicoSedeA-IpPubblicoSedeB/require;  
spdadd LanSedeB/24 lanSedeA/24 any -P out ipsec esp/tunnel/IpPubblicoSedeB-IpPubblicoSedeA /require;
```

setkey.conf

```
flush;  
spdflush;  
spdadd 192.168.4.0/24 192.168.1.0/24 any -P in ipsec esp/tunnel/213.255.43.254-85.142.157.90/require;  
spdadd 192.168.1.0/24 192.168.4.0/24 any -P out ipsec esp/tunnel/85.142.157.90-213.255.43.254/require;
```

Startup/Cron sezione PostBoot

```
# Startup Script  
setkey -f /Database/etc/racoon/setkey.conf  
route add -net LanSedeB /24 gw IpPrivatoLanSedeA  
racoon -f /Database/etc/racoon/racoon.conf
```

Startup/Cron sezione PostBoot

```
# Startup Script  
setkey -f /Database/etc/racoon/setkey.conf  
route add -net 192.168.4.0/24 gw 192.168.1.250  
racoon -f /Database/etc/racoon/racoon.conf
```

Vi inserisco anche alcuni commenti alla guida che potranno tornare utili:

questo , nel mio caso specifico non mi sento di confermarlo per la mia chiave ha funzionato benissimo con la formattazione originale.

“Sperando di evitare agli altri le svariate ore che ho perso per diagnosticare il problema di collegamento di questo tipo di VPN, vi riporto una precisazione importante.

C'è un errore nel formato del file con le chiavi precondivise che viene indicato nel post:

Citazione:

```
Creare un file pskey.conf contenente la chiave condivisa PSK, con la sintassi  
IpPubblicoSedeB : chiavecondivisadaconservareenondivulgaretantoprimaopoilabucanocomunque
```

In realtà il separatore : fra l'IP e la chiave NON va inserito perché viene interpretato come parte della chiave stessa.

Il formato giusto è

IpPubblicoSedeB chiavecondivisadaconservareenondivulgaretantoprimaopoilabucanocomunque

Ricordo che IPSEC Passthrough = IPSec NAT-T = NAT Traversal; dunque in queste condizioni il lato esterno del tunnel è indirizzo pubblico ma quello esposto verso il router: e cioè rispettivamente (IpPrivato2SedeA) e (IpPrivato2SedeB).

Osservazioni:

- Assicurarsi di aprire le porte dei router udp 500 e udp 4500, virtual server, source nat, IPSEC Passthrough, chiamatelo come volete ma quelle due porte devono puntare sull'ip esterno di ZeroShell .

- Assicurarsi che il file /Database/etc/racoon/pskey.conf abbia i permessi 600 dunque un bel chmod 600 /Database/etc/racoon/pskey.conf, perchè altrimenti il demone racoon inizia a fare casini.

- Invece di riavviare potete lanciare manualmente i comandi che avete copiato in Startup/Cron/PostBoot ed aggiungere il "-F" al demone racoon "F" -f /Database/etc/racoon/racoon.conf così partirà non in versione demone e dunque vi mostrerà tutto quello che fa. Se aggiungete pure una vi fa vedere più roba! debug.

- Ultima osservazione: se avete copiato l'intero documento con un editor decente ed avete raccolto tutti i dati detti all'inizio facendo un bel trova e sostituisci in tutto il documento per ogni parametro, vi troverete le configurazioni pronte all'uso !!! wow!!!

Adesso passiamo alla spiegazione estremamente concisa e lasciata intenzionalmente alla fine:

Il file /Database/etc/racoon/pskey.conf contiene le Pre shared key o PSK cioè il riferimento IPPubblico : chiavecondivisa rispettivamente per le sedi con cui ci si deve

collegare.

Il file /Database/etc/racoon/racoon.conf contiene le direttive IKE per il collegamento ed in particolare:

listen = direttiva per mettere in ascolto il demone solo su un ip specifico, senza questa direttiva si metterebbe in ascolto su tutte le interfacce che troverebbe nel sistema. Isakmp e isakmp_natt sono le porte standard con cui il demone lavora, ma anche qualsiasi sistema IPSEC di questo pianeta !!!

Remote ippubblico = direttiva che specifica la fase1 di IKE che contiene l'indirizzo a cui ci vogliamo collegare ed i parametri della fase1 con la specifica nat_traversal che ci permette di attraversare un router/firewall.

my_identifier = Ho utilizzato address per perchè in possesso di IP pubblici statici, ma si potrebbe utilizzare l'FQDN per gli IP dinamici registrati a qualche servizio tipo dyndns.org.

Proposal = parametri di interscambio chiavi e tipo di algoritmo condiviso. Ovviamente esistono molti tipi di algoritmi anche piu' solidi, ma quelli utilizzati sono di gran lunga i piu' compatibili che io abbia provato.

sainfo = parametri fase2 IKE e relativi algoritmi condivisi di interscambio o compressione pacchetto.

Il file /Database/etc/racoon/setkey.conf contiene le policy di ingresso ed uscita del tunnel; in poche parole se non gli diciamo quale traffico incanalare nel tunnel come fa a sapere quale pacchetti deve incanalare e quali no?

Buona Lavoro a tutti.

[Luca Giuffrè](#)