Configure IPSec VPN at Softlayer

**Introduction:**

This document shows IPSec VPN with pre-configured keys can be setup at Softlayer on a VM that is created on Softlayer with Vyatta OS (community edition). In a typical deal only one half of this setup would be needed at a datacenter. This script configures IPSec VPN in 2 datacenters and demonstrates the tunnel between these 2 datacenters.

**Note:** Please see attached document for Python and for other setup details

****

**Configuring IPSec VPN on Vyatta using Python**

This document outlines the steps to create an IPSec VPN on Vyatta at Softlayer. In order to prove that a IPSec tunnel is created 2 identical IPSec VPN tunnels are created in 2 datacenters (Singapore and Amsterdam). IPSec VPN is configured on CCIs running Vyatta OS with the peer IP being that of the remote DC at each end. Finally a check is made by ping the tunnel and capturing the packets at the remote end.

The steps are as follows

1. Edit account.txt and enter your username and API\_KEY
2. **Create VMs**
3. **Create VM at Singapore**
4. Edit vm-cfg.txt. Set the DC as SNG01 (for e.x.). Set all the values as required

Run

PS C:\Python27\myscripts\ipsec-vpn\final> python vmcreate-dc.py

API Username : tinniam\_ganesh

API Key : 69c9c1665a770f0ff43e852aeaa3359bf48310a57ff55a41873ac2824d9fc57

Configuration details

Datacenter: sng01

Domain: vyatta.softlayer.com

Hourly Flag: y

Local Disk Flag: y

Max memory : 1G

Hostname: tvganesh

CPU: 1

OS: VYATTA\_LATEST

Configuring VM at sng01

{'accountId': 284483,

'createDate': '2015-07-08T05:01:39-06:00',

'domain': 'vyatta.softlayer.com',

'fullyQualifiedDomainName': 'tvganesh.vyatta.softlayer.com',

'globalIdentifier': 'd0264356-dccc-4dd6-856d-28e440be165b',

'hostname': 'tvganesh',

'id': 10665705,

'lastPowerStateId': '',

'lastVerifiedDate': '',

'maxCpu': 1,

'maxCpuUnits': 'CORE',

'maxMemory': 1024,

'metricPollDate': '',

'modifyDate': '',

'provisionDate': '',

'startCpus': 1,

'statusId': 1001,

'uuid': '89d806b4-30ef-479f-9d5e-5b2ffb8b6c1d'}

1. **Create VM at Amsterdam**
2. Edit vm-cfg.txt. Set the DC as AMS01 (for e.x.). Set all the values as required
3. python vm-create.py
4. **Edit IPSec VPN details at Singapore DC.**
5. Edit vm<dc>-ip-log.txt (e.g. vmsng01-ip-login.txt). Enter the values as provided in the header. A sample is shown below

# The values in this file should be as

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

# Vyatta IPSec VPN Configuration file

# Designed and developed by : Tinniam V Ganesh

# Date : 08 Jul 2015

# email : tv\_ganesh@in.ibm.com

#

# The values in this file should be as

# 1. IP Address of Vyatta VM

# 2. Password

# 3. Public IP address of remote peer

# 4. Local Public IP

# 5. Local Private IP

# 6. Remote Private IP

# IKE values

# 7. Proposal 1 - encryption

# 8. Proposal 1 - hash

# 9. Proposal 1 - dh-group

# 10. Proposal 2 - encryption

# 11. Proposal 2 - hash

# 12. Proposal 2 - dh-group

# 13. Lifetime - lifetime

# ESP values

# 14. Proposal 1 - vpn\_encryption

# 15. Proposal 1 - hash

# 16. Proposal 2 - vpn\_encryption

# 17. Proposal 2 - hash

# 18. Lifetime - lifetime

# Pre Shared secret

# 19. Pre shared secret

#

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

119.81.103.139

Jdedg3EN

159.253.140.61

119.81.103.139

10.67.55.194

10.68.231.232

#IKE paramaters

3des

sha1

2

3des

sha1

2

86400

#ESP parameters

3des

sha1

3des

sha1

28800

#Pre-shared secret

IPSec\_demo

1. Run the Replace script

python replace\_ip\_login.py <dc> for e.g.

python replace\_ip\_login.py sng01

PS C:\Python27\myscripts\ipsec-vpn\final> python .\replace\_ip\_login.py sng01

Reading from file: ./vmsng01-ip-login.txt

Writing to output file: ipsec-vpn-create-sng01-sl.py

1. **Create IPSec VPN at the Singapore DC**
2. Run the modified file

python .\ipsec-vpn-create-sng01-sl.py

….

….

}

vpn {

ipsec {

esp-group ESP-1W {

lifetime 28800

proposal 1 {

encryption 3des

hash sha1

}

proposal 2 {

encryption 3des

hash sha1

}

}

ike-group IKE-1W {

lifetime 86400

proposal 1 {

dh-group 2

encryption 3des

hash sha1

}

proposal 2 {

dh-group 2

encryption 3des

hash sha1

}

}

ipsec-interfaces {

interface eth1

}

site-to-site {

peer 159.253.140.61 {

authentication {

mode pre-shared-secret

pre-shared-secret IPSec-demo

}

default-esp-group ESP-1W

ike-group IKE-1W

local-address 119.81.103.139

tunnel 1 {

local {

prefix 10.67.55.194/32

}

remote {

prefix 10.68.231.232/32

}

}

}

}

}

}

1. **Edit IPSec VPN details at Amsterdam DC**
2. Edit vm<dc>-ip-log.txt (e.g. vmams01-ip-login.txt). Enter the values as provided in the header
3. Run the replace script

PS C:\Python27\myscripts\ipsec-vpn\final> python .\replace\_ip\_login.py ams01

Reading from file: ./vmams01-ip-login.txt

Writing to output file: ipsec-vpn-create-ams01-sl.py

1. **Create IPSec VPN at Amsterdam DC**
2. Run the modified file

PS C:\Python27\myscripts\ipsec-vpn\final> python .\ipsec-vpn-create-ams01-sl.py

…

…

vpn {

ipsec {

esp-group ESP-1W {

lifetime 28800

proposal 1 {

encryption 3des

hash sha1

}

proposal 2 {

encryption 3des

hash sha1

}

}

ike-group IKE-1W {

lifetime 86400

proposal 1 {

dh-group 2

encryption 3des

hash sha1

}

proposal 2 {

dh-group 2

encryption 3des

hash sha1

}

}

ipsec-interfaces {

interface eth1

}

site-to-site {

peer 119.81.103.139 {

authentication {

mode pre-shared-secret

pre-shared-secret IPSec-demo

}

default-esp-group ESP-1W

ike-group IKE-1W

local-address 159.253.140.61

tunnel 1 {

local {

prefix 10.68.231.232/32

}

remote {

prefix 10.67.55.194/32

}

}

}

}

}

}

1. Now IPSec VPN has been setup in Singapore and Amsterdam DCs. We can verify the connection is up by using Putty
2. Login to Singapore VM

Type the following

vyatta@tvganesh-a:~$ show vpn ipsec status

IPSec Process Running PID: 2148

1 Active IPsec Tunnels

IPsec Interfaces :

eth1 (no IP on interface statically configured as local-ip for any VPN peer)

1. Similarly check at Amsterdam

vyatta@tvganesh-b:~$ show vpn ipsec status

IPSec Process Running PID: 2191

1 Active IPsec Tunnels

IPsec Interfaces :

eth1 (no IP on interface statically configured as local-ip for any VPN peer)

1. Now ping Amsterdam private IP through the IPSec VPN tunnel from Singapore DC. A reply will be received if the tunnel is up

vyatta@tvganesh-a:~$ sudo ping -I eth0 10.68.231.232

PING 10.68.231.232 (10.68.231.232) from 10.67.55.194 eth0: 56(84) bytes of data.

64 bytes from 10.68.231.232: icmp\_req=1 ttl=64 time=313 ms

64 bytes from 10.68.231.232: icmp\_req=2 ttl=64 time=313 ms

64 bytes from 10.68.231.232: icmp\_req=3 ttl=64 time=313 ms

64 bytes from 10.68.231.232: icmp\_req=4 ttl=64 time=313 ms

H) Check for the ping from Singapore DC at Amsterdam using tcpdump

vyatta@tvganesh-b:~$ /usr/sbin/tcpdump -i eth1 icmp -nn

tcpdump: verbose output suppressed, use -v or -vv for full protocol decode

listening on eth1, link-type EN10MB (Ethernet), capture size 65535 bytes

05:36:31.780890 IP 10.67.55.194 > 10.68.231.232: ICMP echo request, id 2648, seq 6, length 64

05:36:32.782573 IP 10.67.55.194 > 10.68.231.232: ICMP echo request, id 2648, seq 7, length 64

05:36:33.784272 IP 10.67.55.194 > 10.68.231.232: ICMP echo request, id 2648, seq 8, length 64

05:36:34.785960 IP 10.67.55.194 > 10.68.231.232: ICMP echo request, id 2648, seq 9, length 64

In actual client deal only 1 half of the IPSec VPN tunnel will need to be configured

The files are included in the zip below

