KELVIN SEVERINO

ABNER CARLUCCI

TRABALHO 1

OPENVPN SERVIDOR E CLIENTE

TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

4º ADS

DISCIPLINA: Redes de Computadores

Prof. Me. Jean

São José dos Campos – SP

2018

O Conteúdo abaixo é referente ao trabalho 1 – VPN da matéria de Redes de Computadores.

Para a realizar deste trabalho, foi necessário a utilização de 2 máquinas virtuais (VM):
1 – VPN Server
2 – VPN Cliente

Configuração Inicial no Servidor OpenVPN

#apt-get install openvpn
#apt-get install easy-rsa
#apt-get install gzip

Copiar diretório **easy-rsa** para **/etc/openvpn/**
 # cp -r /usr/share/easy-rsa/ /etc/openvpn/

Dentro do diretório do easy-rsa, abrir o arquivo vars e inserir as seguintes informações
 # nano vars
Obs. Note que você deverá inserir as seguintes informações de acordo com a sua necessidade

##
 # export KEY_COUNTRY="BR"   #
 # export KEY_PROVINCE="SP"   #
 # export KEY_CITY="Cacapava"   #
 # export KEY_ORG="Kelvin Corp"   #
 # export KEY_EMAIL="kelvin@dominio.com.br"   #
##

```
export EASY_RSA="`pwd`"

export OPENSSL="openssl"
export PKCS11TOOL="pkcs11-tool"
export GREP="grep"

export KEY_CONFIG=`$EASY_RSA/whichopensslcnf $EASY_RSA`

export KEY_DIR="$EASY_RSA/keys"


echo NOTE: If you run ./clean-all, I will be doing a rm -rf on $KEY_DIR

export PKCS11_MODULE_PATH="dummy"
export PKCS11_PIN="dummy"

export KEY_SIZE=1024

export CA_EXPIRE=3650

export KEY_EXPIRE=3650

export KEY_COUNTRY="BR"
export KEY_PROVINCE="SP"
export KEY_CITY="Cacapava"
export KEY_ORG="Kelvin Corp"
export KEY_EMAIL="kelvin@dominio.com"

export KEY_NAME=""
```

Criar o diretório keys no easy-rsa e inserir os sequintes arquivos

 # mkdir /etc/openvpn/easy-rsa/keys
 # echo 01 > /etc/openvpn/easy-rsa/keys/serial
 # touch /etc/openvpn/easy-rsa/keys/index.txt

Retornar para o diretório easy-rsa
 # cd /etc/openvpn/easy-rsa/

Rodar o comando source sobre o arquivo vars
 # source vars

OBS. Caso seja apresentado o erro "No /etc/openvpn/easy-rsa/openssl.cnf file could be found Further invocations will fail". É necessário ir em **/etc/openvpn/easy-rsa** para renomar o openssl, pois o mesmo não esta sendo encontrado.

 # cd /etc/openvpn/easyrsa/
 # mv openssl-1.0.0.cnf /etc/openvpn/easyrsa/openssl.cnf

Agora vamos criar a autoridade certificadora build-ca
 # ./build-ca

Agora vamos criar o dh com build-dh
 # ./build-dh

Após a criação do CA, criaremos a chave privada do Servidor com seguinte comando e depois do cliente
 # ./build-key-server servidor
 # ./build-key cliente
Note que a variável servidor na frente de ./build-key-server ja preencherá o Common Name (CN), isso acontecerá também na geração de chave do cliente

Agora devemos transferir as chaves a serem utilizadas na máquina cliente, sendo as chaves abaixo.

- **ca.crt**
- **cliente.crt**
- **cliente.key**
- **dh1024.pem**

 # scp ca.crt kelvin@192.168.10.114:/etc/openvpn/easy-rsa/keys
 # scp cliente.crt kelvin@192.168.10.114:/etc/openvpn/easy-rsa/keys
 # scp cliente.key kelvin@192.168.10.114:/etc/openvpn/easy-rsa/keys
 # scp dh1024.pem kelvin@192.168.10.114:/etc/openvpn/easy-rsa/keys

Configuração do Servidor OpenVPN

Devemos ir no diretório /etc/openvpn e criar o arquivo servidor.conf

cp /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz /etc/openvpn/

tar -vzxf server.conf

cp server.conf /etc/openvpn/servidor.conf

nano /etc/openvpn/servidor.conf

```
# /etc/openvpn/servidor.conf
port 1194
proto udp
dev tun

server 10.8.0.0 255.255.255.0

dh easy-rsa/keys/dh1024.pem
ca easy-rsa/keys/ca.crt
cert easy-rsa/keys/servidor.crt
key easy-rsa/keys/servidor.key

ifconfig-pool-persist /var/log/openvpn/ipp.txt
cipher AES-256-CBC
persist-key
persist-tun
status /var/log/openvpn/openvpn-status.log
verb 3
explicit-exit-notify 1
```

Após terminar de configurar o arquivo **servidor.conf**, vamos ativar o mesmo com o comando **openvpn --config servidor.conf**.

Obs. Para que o comando funcione, devemos estar no diretório /etc/openvpn

openvpn --config servidor.conf

Se o serviço for iniciado com sucesso, você verá a imagem abaixo.

```
Sun Sep  2 22:50:32 2018 OpenVPN 2.4.4 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [
LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on Feb 10 2018
Sun Sep  2 22:50:32 2018 library versions: OpenSSL 1.1.0g  2 Nov 2017, LZO 2.08
Sun Sep  2 22:50:32 2018 WARNING: --keepalive option is missing from server config
Sun Sep  2 22:50:32 2018 Diffie-Hellman initialized with 1024 bit key
Sun Sep  2 22:50:32 2018 ROUTE_GATEWAY 192.168.10.1/255.255.255.0 IFACE=enp0s3 HWA
DDR=08:00:27:c6:8e:96
Sun Sep  2 22:50:32 2018 TUN/TAP device tun0 opened
Sun Sep  2 22:50:32 2018 TUN/TAP TX queue length set to 100
Sun Sep  2 22:50:32 2018 do_ifconfig, tt->did_ifconfig_ipv6_setup=0
Sun Sep  2 22:50:32 2018 /sbin/ip link set dev tun0 up mtu 1500
Sun Sep  2 22:50:32 2018 /sbin/ip addr add dev tun0 local 10.8.0.1 peer 10.8.0.2
Sun Sep  2 22:50:32 2018 /sbin/ip route add 10.8.0.0/24 via 10.8.0.2
Sun Sep  2 22:50:32 2018 Could not determine IPv4/IPv6 protocol. Using AF_INET
Sun Sep  2 22:50:32 2018 Socket Buffers: R=[212992->212992] S=[212992->212992]
Sun Sep  2 22:50:32 2018 UDPv4 link local (bound): [AF_INET][undef]:1194
Sun Sep  2 22:50:32 2018 UDPv4 link remote: [AF_UNSPEC]
Sun Sep  2 22:50:32 2018 MULTI: multi_init called, r=256 v=256
Sun Sep  2 22:50:32 2018 IFCONFIG POOL: base=10.8.0.4 size=62, ipv6=0
Sun Sep  2 22:50:32 2018 ifconfig_pool_read(), in='cliente,10.8.0.4', TODO: IPv6
Sun Sep  2 22:50:32 2018 succeeded -> ifconfig_pool_set()
Sun Sep  2 22:50:32 2018 IFCONFIG POOL LIST
Sun Sep  2 22:50:32 2018 cliente,10.8.0.4
Sun Sep  2 22:50:32 2018 Initialization Sequence Completed
```

Configuração da Máquina Cliente que acessa a VPN

Devemos ir no diretório /etc/openvpn e criar o arquivo cliente.conf

cd /etc/openvpn/

> cliente.conf

```
client
dev tun
proto udp
remote 192.168.10.113 1194 #IP e Porta de Conexão com o Servidor
resolv-retry infinite
nobind
persist-key
persist-tun

dh easy-rsa/keys/dh1024.pem
ca easy-rsa/keys/ca.crt
cert easy-rsa/keys/cliente.crt
key easy-rsa/keys/cliente.key

remote-cert-tls server
cipher AES-256-CBC
ping 10
verb 3
mute 10
```

Após terminar de configurar o arquivo **cliente.conf**, vamos ativar o mesmo com o comando **openvpn --config cliente.conf**.

Obs. Para que o comando funcione, devemos estar no diretório /etc/openvpn

openvpn --config cliente.conf

Se o serviço do cliente for iniciado com sucesso, você verá a imagem abaixo.

```
root@vpn_cliente:/etc/openvpn# openvpn --config cliente.conf
Thu Sep  6 00:40:54 2018 WARNING: Ignoring option 'dh' in tls-client mode, please only include this in your server configuration
Thu Sep  6 00:40:54 2018 OpenVPN 2.4.4 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on Feb 10 2018
Thu Sep  6 00:40:54 2018 library versions: OpenSSL 1.1.0g  2 Nov 2017, LZO 2.08
Thu Sep  6 00:40:54 2018 TCP/UDP: Preserving recently used remote address: [AF_INET]192.168.10.113:1194
Thu Sep  6 00:40:54 2018 Socket Buffers: R=[212992->212992] S=[212992->212992]
Thu Sep  6 00:40:54 2018 UDP link local: (not bound)
Thu Sep  6 00:40:54 2018 UDP link remote: [AF_INET]192.168.10.113:1194
Thu Sep  6 00:40:54 2018 TLS: Initial packet from [AF_INET]192.168.10.113:1194, sid=f391e8d3 524682d8
Thu Sep  6 00:40:54 2018 VERIFY OK: depth=1, C=BR, ST=SP, L=Cacapava, O=Kelvin Corp, CN=Kelvin Corp CA, emailAddress=kelvin@email.com
Thu Sep  6 00:40:54 2018 VERIFY KU OK
Thu Sep  6 00:40:54 2018 Validating certificate extended key usage
Thu Sep  6 00:40:54 2018 ++ Certificate has EKU (str) TLS Web Server Authentication, expects TLS Web Server Authentication
Thu Sep  6 00:40:54 2018 VERIFY EKU OK
Thu Sep  6 00:40:54 2018 VERIFY OK: depth=0, C=BR, ST=SP, L=Cacapava, O=Kelvin Corp, CN=servidor, emailAddress=kelvin@email.com
Thu Sep  6 00:40:54 2018 Control Channel: TLSv1.2, cipher TLSv1.2 ECDHE-RSA-AES256-GCM-SHA384, 1024 bit RSA
Thu Sep  6 00:40:54 2018 [servidor] Peer Connection Initiated with [AF_INET]192.168.10.113:1194
Thu Sep  6 00:40:55 2018 SENT CONTROL [servidor]: 'PUSH_REQUEST' (status=1)
Thu Sep  6 00:40:55 2018 PUSH: Received control message: 'PUSH_REPLY,route 10.8.0.1,topology net30,ifconfig 10.8.0.6 10.8.0.5,peer-id 0,cipher AES-256-GCM'
Thu Sep  6 00:40:55 2018 OPTIONS IMPORT: --ifconfig/up options modified
Thu Sep  6 00:40:55 2018 OPTIONS IMPORT: route options modified
Thu Sep  6 00:40:55 2018 OPTIONS IMPORT: peer-id set
Thu Sep  6 00:40:55 2018 OPTIONS IMPORT: adjusting link_mtu to 1624
Thu Sep  6 00:40:55 2018 OPTIONS IMPORT: data channel crypto options modified
Thu Sep  6 00:40:55 2018 Data Channel: using negotiated cipher 'AES-256-GCM'
Thu Sep  6 00:40:55 2018 Outgoing Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit key
Thu Sep  6 00:40:55 2018 Incoming Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit key
Thu Sep  6 00:40:55 2018 ROUTE_GATEWAY 192.168.10.1/255.255.255.0 IFACE=enp0s3 HWADDR=08:00:27:3a:6c:81
Thu Sep  6 00:40:55 2018 TUN/TAP device tun0 opened
Thu Sep  6 00:40:55 2018 TUN/TAP TX queue length set to 100
Thu Sep  6 00:40:55 2018 do_ifconfig, tt->did_ifconfig_ipv6_setup=0
Thu Sep  6 00:40:55 2018 /sbin/ip link set dev tun0 up mtu 1500
Thu Sep  6 00:40:55 2018 /sbin/ip addr add dev tun0 local 10.8.0.6 peer 10.8.0.5
Thu Sep  6 00:40:55 2018 /sbin/ip route add 10.8.0.1/32 via 10.8.0.5
Thu Sep  6 00:40:55 2018 WARNING: this configuration may cache passwords in memory -- use the auth-nocache option to prevent this
Thu Sep  6 00:40:55 2018 Initialization Sequence Completed
```

Testes para Confirmação do Funcionamento da VPN

Para confirmar se a foi criado um túnel da vpn entre o cliente e servidor.

Vamos pingar utilizando o IP da VPN atribuído no cliente e servidor
$ ping 10.8.0.6

```
kelvin@vpn_server:~$ ping 10.8.0.6
PING 10.8.0.6 (10.8.0.6) 56(84) bytes of data.
64 bytes from 10.8.0.6: icmp_seq=1 ttl=64 time=0.643 ms
64 bytes from 10.8.0.6: icmp_seq=2 ttl=64 time=0.695 ms
64 bytes from 10.8.0.6: icmp_seq=3 ttl=64 time=0.827 ms
64 bytes from 10.8.0.6: icmp_seq=4 ttl=64 time=0.647 ms
^C
--- 10.8.0.6 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3071ms
rtt min/avg/max/mdev = 0.643/0.703/0.827/0.074 ms
kelvin@vpn_server:~$
```

Vamos enviar um arquivo via SSH para a máquina cliente utilizando o servidor.
 # scp arquivo.txt cliente@10.8.0.6:/home/cliente

```
kelvin@vpn_server:~$ scp arquivo.txt cliente@10.8.0.6:/home/cliente
cliente@10.8.0.6's password:
arquivo.txt                                   100%    0     0.0KB
kelvin@vpn_server:~$ _
```

Existem outros testes que podem ser feitos, como realizar o Ping de uma máquina fora do túnel da VPN, e esta máquina não pode conseguir pingar ou enviar qualquer dado para as máquinas que estejam no túnel da VPN por meio do IP da VPN.