IPsec based VPN

Using libreswan

Presented by
Paul Wouters
Redhat

Today's Topics

1. Quick IPsec primer
2. Libreswan configuration examples
3. Building your own tunnels

IPsec Primer

The IKE daemon (pluto)

- Internet Keying Exchange (“IKE”) daemon in userland
  - IKE is the “command channel” of IPsec
  - Peer authentication
  - Connection parameter negotiation
  - IPsec symmetric encryption key generation
  - Injecting/removing keys and policies from the kernel IPsec state (SPD and SAD)
  - IKE itself is encrypted!
  - IKE does not encrypt the data!

Kernel IPsec

- kernel level IPsec packet encrypter and decrypter
  - does not depend on routing
- Userland and kernel talk to each other via netlink/XFRM
  - See also “ip xfrm state” and “ip xfrm pol”
- iptables rules via:
  - -m policy -- dir in|out --pol ipsec [--reqid XXX]

The IKE protocol

- IKEv1 (1998) and IKEv2 (2005)
- Runs over UDP port 500
- And over UDP 4500 for NAT_TRAVERSAL
- Creates Security Associations (SA)
  - IKE SA (Parent SA or “Phase 1”)
    - Authentication: PreSharedKey, RSA, X509, GSSAPI
  - IPsec SA (Child SA or “Phase 2”)
    - Negotiation of IP address ranges, crypto params

The IPsec protocol

- Encapsulated Secure Payload (**ESP**)
  - Protocol 50 (not port 50)
  - Can be encapsulated in a UDP 4500 packet
    - Called **ESPinUDP**
  - **Tunnel Mode** (full IP packet in ESP packet)
  - Transport Mode (Encrypt packet itself)
    - Don't use
- Authenticated Header (AH) [don't use]
  - Protocol 51 (not port 51)
- IPcomp [don't use]

Installing libreswan

- dnf | yum | apt-get install libreswan
- Enable the “ipsec” service
  - Via chkconfig or systemctl, etc
- Start the “ipsec” service
  - ipsec start (will expand to init system)

- For client side GUI, install NetworkManager plugin:
  - NetworkManager-libreswan-gnome

IPsec tunnel with PSK

```
# /etc/ipsec.d/yourtunnel.conf
conn YourTunnel
    # you can also use hostnames
    left=193.110.157.124
    right=194.111.228.1
    authby=secret
    auto=start

# /etc/ipsec.d/yourtunnel.secret
193.110.157.124 194.111.228.1 PSK \
   “YourSharedS3cr3t”
```

subnet-to-subnet

```
# /etc/ipsec.d/yourtunnel.conf
conn YourTunnel
    # you can also use hostnames
    left=193.110.157.124
    leftsubnet=192.168.0.0/16
    right=194.111.228.1
    rightsubnet=10.0.0.0/8
    authby=secret
    auto=start

(same /etc/ipsec.d/yourtunnel.secret)
```

Subnet extrusion

- Reroute a part of your network to elsewhere
  # /etc/ipsec.d/yourtunnel.conf
  conn YourTunnel
      # Amsterdam has 193.110.157.0/24
      left=193.110.157.1
      leftsubnet=0.0.0.0/0
      # my DSL machine in Toronto
      right=76.20.157.65
      rightsubnet=193.110.157.16/28
      authby=secret
      auto=start

Using RSA instead of PSK

- Generate RSA keys on both machines:
  - ipsec newhostkey
- Display public RSA key:
  - ipsec showhostkey --left (or –right)
- Exchange public RSA keys over email
- Make up an “ID”, like “Paul” and “Nikos”

Libreswan config with RSA

- # /etc/ipsec.d/yourtunnel.conf
  conn YourTunnel
      # you can also use hostnames
      left=193.110.157.124
      leftid=@Paul
      leftrsasigkey=0x1234567890[...]
      right=194.111.228.1
      rightid=@Nikos
      rightrsasigkey=0x9876543210[...]
      authby=rsasig
      auto=start

  # no secret entry required – stored in NSS DB

Libreswan config with RSA

- # /etc/ipsec.d/yourtunnel.conf
  conn YourTunnel
      left=193.110.157.124
      leftid=@Paul
      leftrsasigkey=0x1234567890[...]
      right=194.111.228.1
      rightid=@Nikos
      rightrsasigkey=0x9876543210[...]
      authby=rsasig
      auto=start

  # no secret file needed – stored in NSS DB

On demand tunnel

```
# /etc/ipsec.d/yourtunnel.conf
conn YourTunnel
    # you can also use hostnames
    left=193.110.157.124
    leftid=@Paul
    leftrsasigkey=0x1234567890[...]
    right=194.111.228.1
    rightid=@Nikos
    rightrsasigkey=0x9876543210[...]
    authby=rsasig
    auto=ondemand
```

Dynamic IP configuration

```
# /etc/ipsec.d/yourtunnel.conf
conn YourTunnel
    left=%defaultroute
    leftid=@Paul
    leftrsasigkey=0x1234567890[...]
    right=%any
    rightid=@Nikos
    rightrsasigkey=0x9876543210[...]
    authby=rsasig
    auto=add
     rekey=no
```

fedora

IKEv1 XAUTH with X.509

```
# /etc/ipsec.d/yourtunnel.conf
# also known as “Cisco IPsec” or “RSA XAUTH”
conn YourTunnel
    left=vpn.example.com
    leftid=%fromcert
    leftcert=friendlyname (comes from PKCS#12)
    leftxauthserver=yes
    leftmodecfgserver=yes
    #
    right=%any
    rightaddresspool=100.64.0.1-100.64.0.254
    rightxauthclient=yes
    rightmodecfgclient=yes
    rightsubnet=0.0.0.0/0
    #
    modecfgpull=yes
    modecfgdns1=10.1.2.3
    modecfgdomain=”example.com”
    authby=rsasig
    auto=add
```

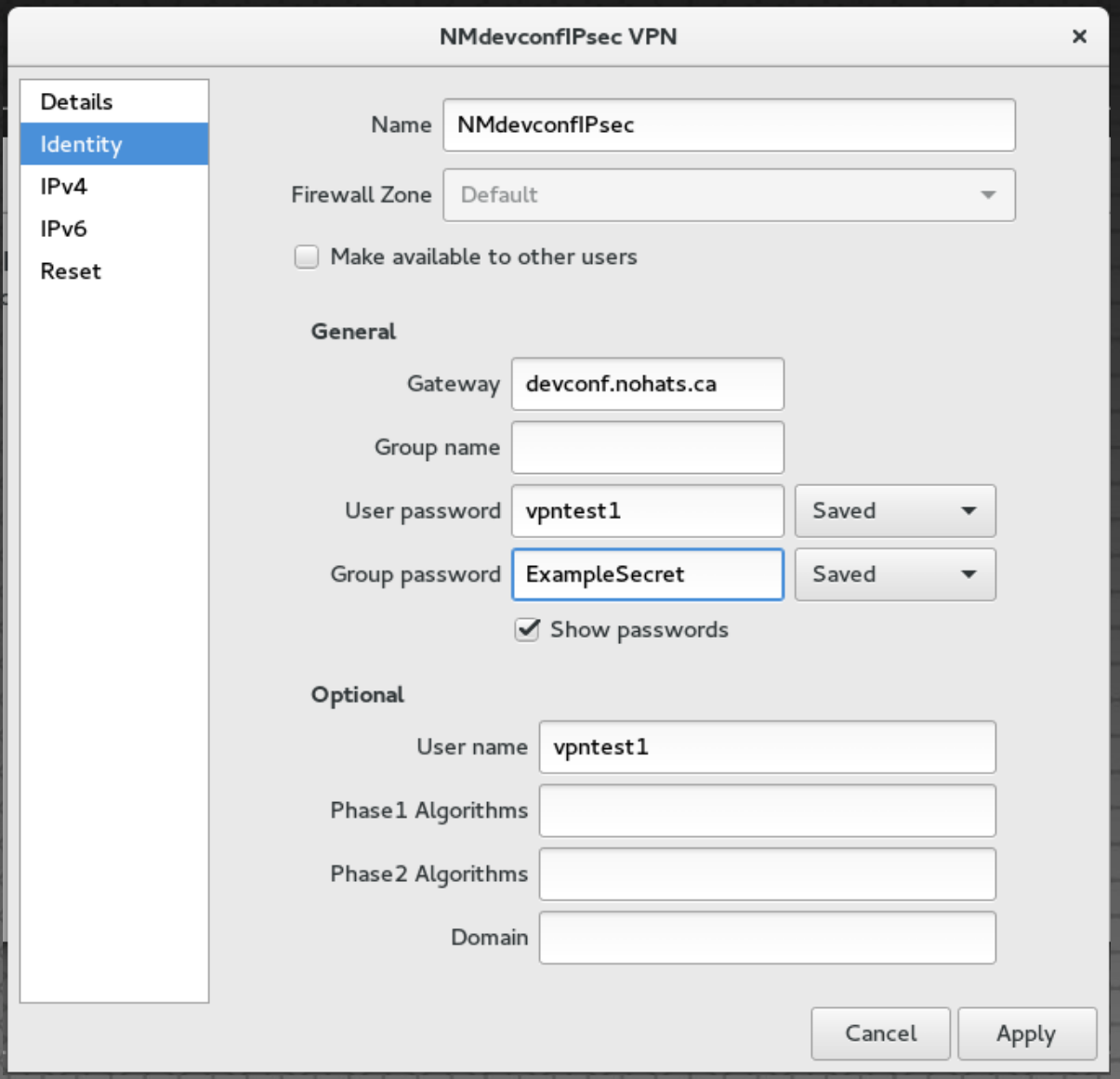
fedora

IKEv1 XAUTH with PSK

```
# /etc/ipsec.d/yourtunnel.conf
# also known as “Cisco IPsec” or “PSK XAUTH”
conn YourTunnel
    left=%defaultroute
    leftid=@GroupName
    leftxauthclient=yes
    leftmodecfgclient=yes
    leftxauthusername=pwouters
    right=vpn.corp.com
    rightxauthserver=yes
    rightmodecfgserver=yes
    rightsubnet=0.0.0.0/0
    modecfgpull=yes
    remote_peer_type=cisco
    aggrmode=yes
    ikelifetime=24h (workaround for bad Cisco's)
    salifetime=24h  (workaround for bad Cisco's)
    ike=aes256sha1;modp1024,aes256-sha1;modp1024
    esp=aes-sha1
    authby=secret
    auto=add
```

fedora

Try NetworkManager plugin



fedora

Libreswan commands

- ipsec auto --add yourconn
- ipsec auto --delete yourconn
- ipsec auto --down yourconn
- ipsec auto --up yourconn
-
- ipsec stop | start | restart
- ipsec whack --listen (run on network change)

Libreswan commands

- ipsec verify (quick system check)
- ipsec whack --trafficstatus (brief overview)
- ipsec status (ridiculous dump for developers)
- ipsec barf (snapshot including logs, system, etc)

- ipsec import /path/to/file.p12
- certutil -d sql:/etc/ipsec.d/ -L

But our true goal

Opportunistic Encryption

- Encrypt the entire internet with IPsec
  - (been trying since 1995 with FreeS/WAN)
- Authenticated if possible
  - One-sided authenticated if client desires
  - GSSAPI, DNSSEC, LetsEncrypt-CA
    (if you don't trust any of these, write a bitcoin auth plugin for us)
- Unauthenticated if all else fails
  - but don't tell user we encrypted at all

DEMO

fedora

If you want to try OE

- (for now, no NAT support, coming soon)
- cd /etc/ipsec.d/
- wget github.com/libreswan/libreswan/examples/oe-upgrade-authnull.conf
- echo “0.0.0.0/0” >> /etc/ipsec.d/policies/private-or-clear
- ipsec restart
- ping oe.libreswan.org
- ipsec whack --trafficstatus
  or browse to http://oe.libreswan.org/

fedora

Questions?

Contact:
pwouters@redhat.com