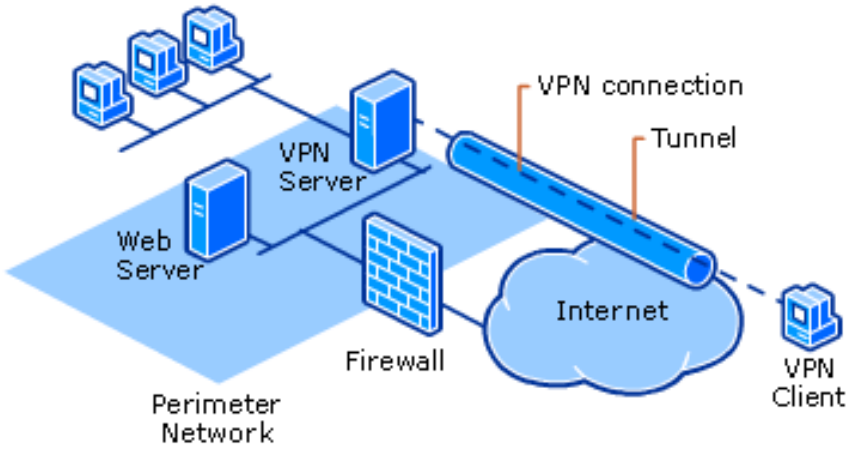
VPN

Virtual Private Network

hlku

What is VPN

- Extension of a private network that encompasses links across shared or public networks like the Internet.
- Enable to send data between two computers across a shared or public internetwork in a manner that emulates the properties of a point-to-point private link.

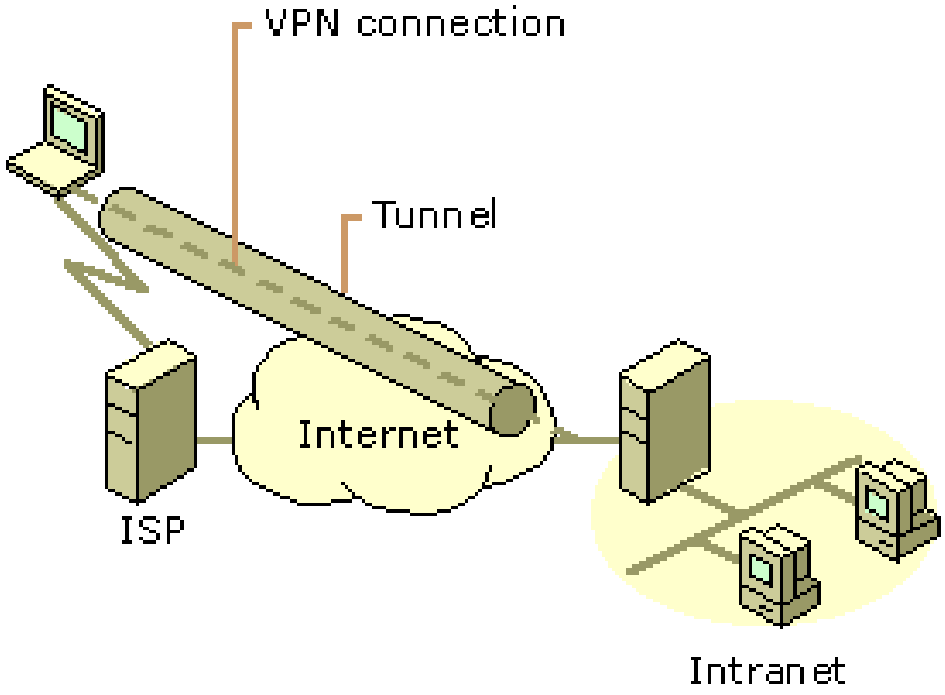


Why ?

- Cheap
  - Legacy private network uses remote connectivity through dial-up modems or through leased line connections, it's expensive.
- Scalable
  - Extending a leased line connection is complex.
  - Easy to administer.
- Security
  - Provide encryption and file integrity.

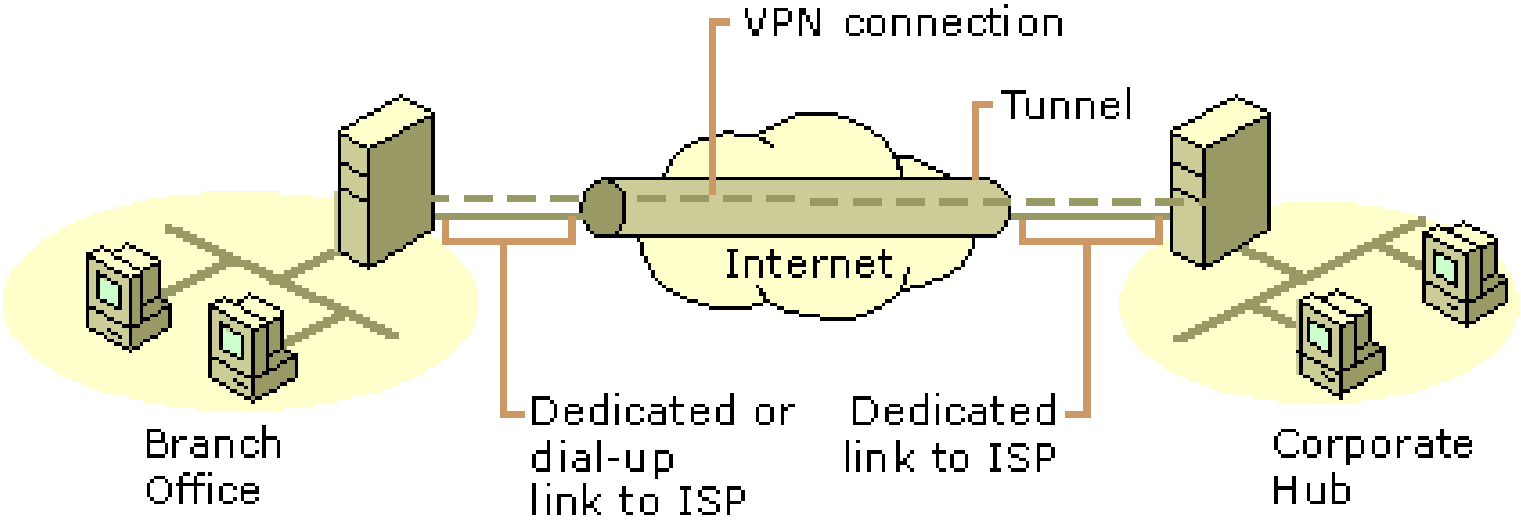
Common Uses of VPNs – 1

- Remote Access Over the Internet



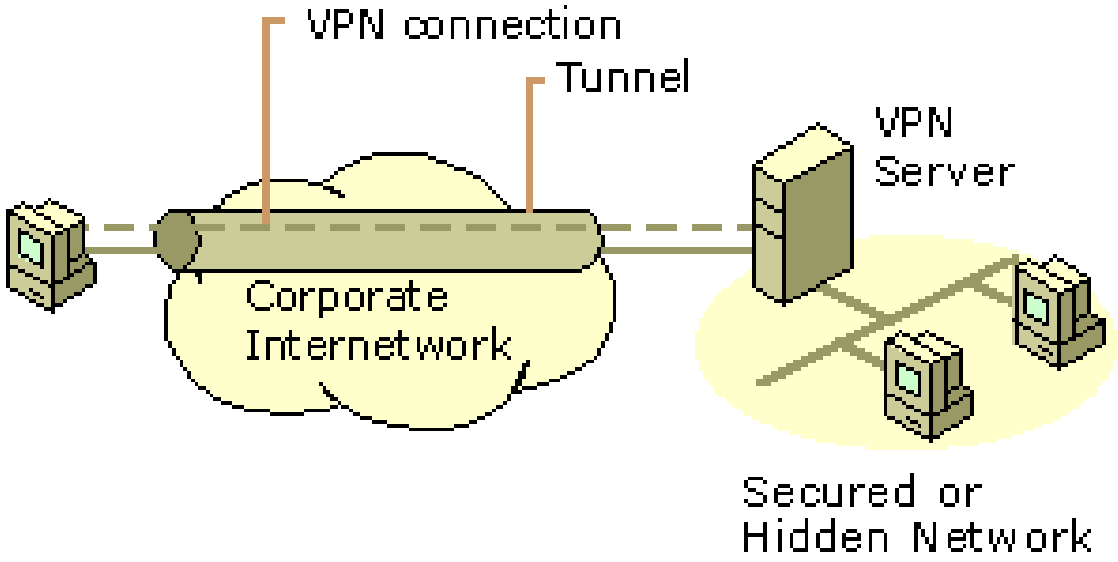
Common Uses of VPNs – 2

- Connecting Networks Over the Internet (Site to Site VPN)



Common Uses of VPNs – 3

- Connecting Computers over an Intranet



Basic VPN Requirements

- User Authentication
- Key Management
- Address Management
- Data Encryption

Basic VPN Requirements – 1

- User Authentication
  - Verify the VPN client's identity and restrict VPN access to authorized users only.
  - Provide audit and accounting records to show who accessed what information and when.
  - X.509, pre-share key….
- Key Management
  - Generate and refresh encryption keys for the client and the server.
  - Simple Key Management for IP, ISAKMP/Oakley…

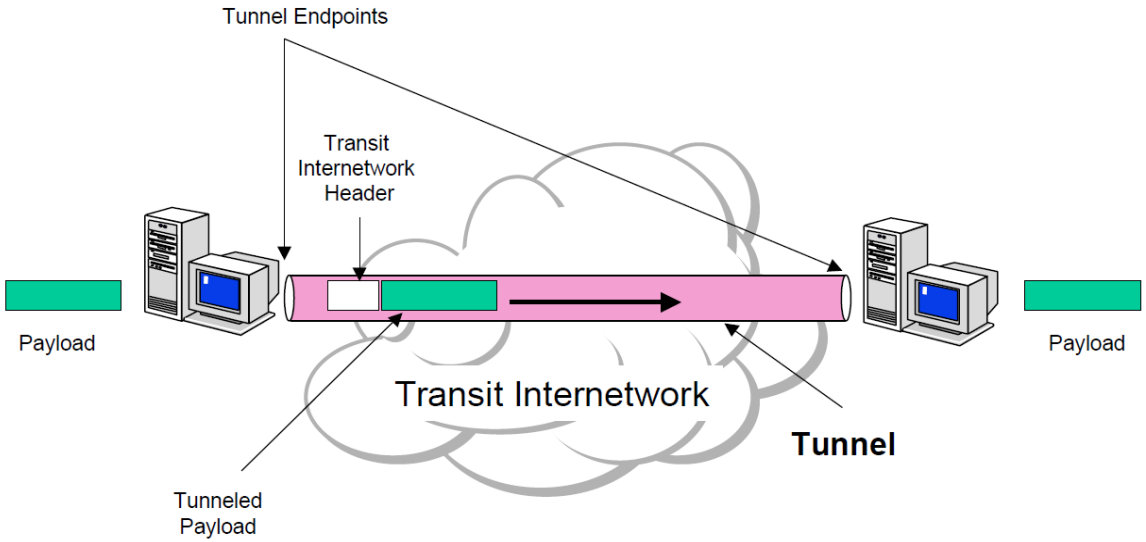
Basic VPN Requirements – 2

- Address Management
  - Assign a VPN client's address on the intranet and ensure that private addresses are kept private.

- Data Encryption
  - No one outside the VPN can alter the VPN.
  - Data carried on the public network must be rendered unreadable to unauthorized clients on the network.

Tunneling

- VPN consists of a set of point to point connections tunneled over the Internet.
- In order to achieve tunneling, the packets are encapsulated as the payload of packets.
  - Payloads, to and from addresses, port numbers and other standard protocol packet headers
  - As seen by the external routers carrying the connection



Common Implementations

- Point-to-Point Tunneling Protocol (PPTP) [RFC 2637]
- Layer Two Tunneling Protocol (L2TP) [RFC 2661]
- IPSec Tunnel Mode [RFC 2401]
- Secure Socket Tunneling Protocol (SSTP) [Spec]
- BGP/MPLS IP VPN [RFC 4364]
- SSL VPN

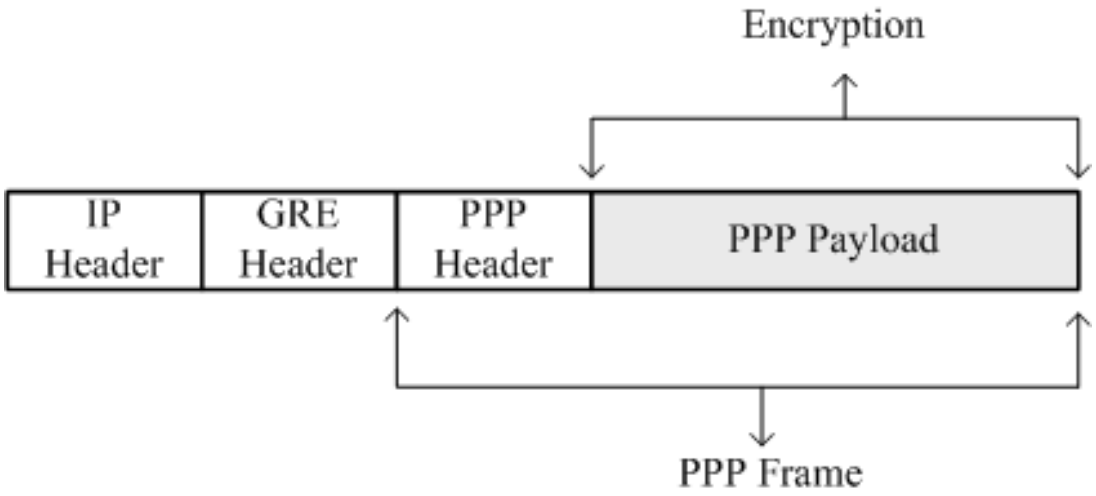
…, etc

PPP

- Point-to-Point Protocol [RFC 1661]
- PPP was designed to send data across dial-up or dedicated point-to-point connections.
  - PPP encapsulates IP, IPX, and NetBEUI packets within PPP frames, and then transmits the PPP-encapsulated packets across a point-to-point link.
- User Authentication
  - Password Authentication Protocol (PAP)
  - Challenge Handshake Authentication Protocol (CHAP)
  - M$ Challenge Handshake Authentication Protocol (M$-CHAP)
  - M$-CHAPv2
- Data can be compressed or encrypted before transmission.
  - Microsoft Point to Point Compression / Encryption (MPPC / E)

PPTP

- Point-to-Point Tunneling Protocol
  - PPTP doesn't describe encryption or authentication
    - Rely on the PPP protocol
  - PPTP encapsulates PPP frames in IP datagrams for transmission over an IP internetwork by TCP connection.
  - PPTP uses a modified version of Generic Routing Encapsulation (GRE) to encapsulate PPP frames for tunneled data.

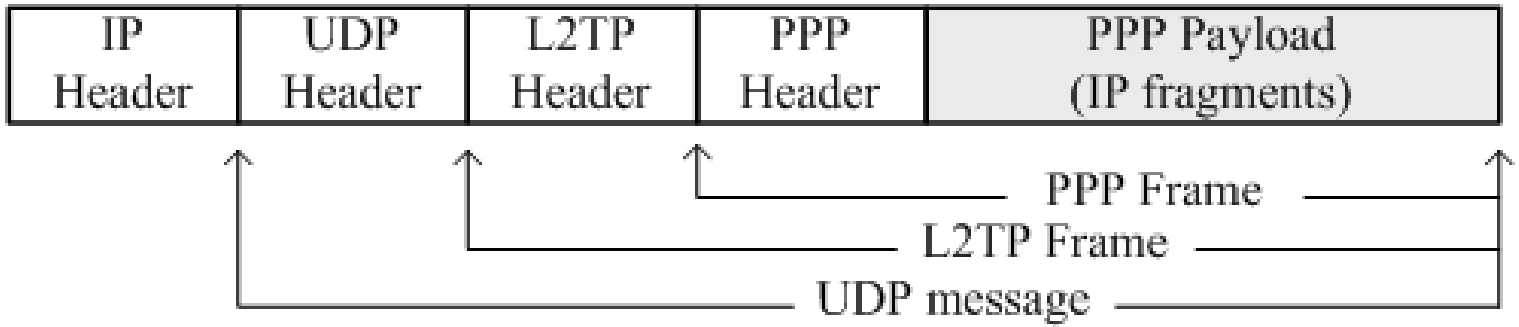


Security of PPTP

- PPTP has been the subject of many security analyses and serious security vulnerabilities have been found
  - MS-CHAP is fundamentally insecure.
  - MS-CHAPv2 is vulnerable to dictionary attack on the captured challenge response packets.
- EAP-TLS (Extensible Authentication Protocol – TLS) is the superior authentication choice for PPTP.

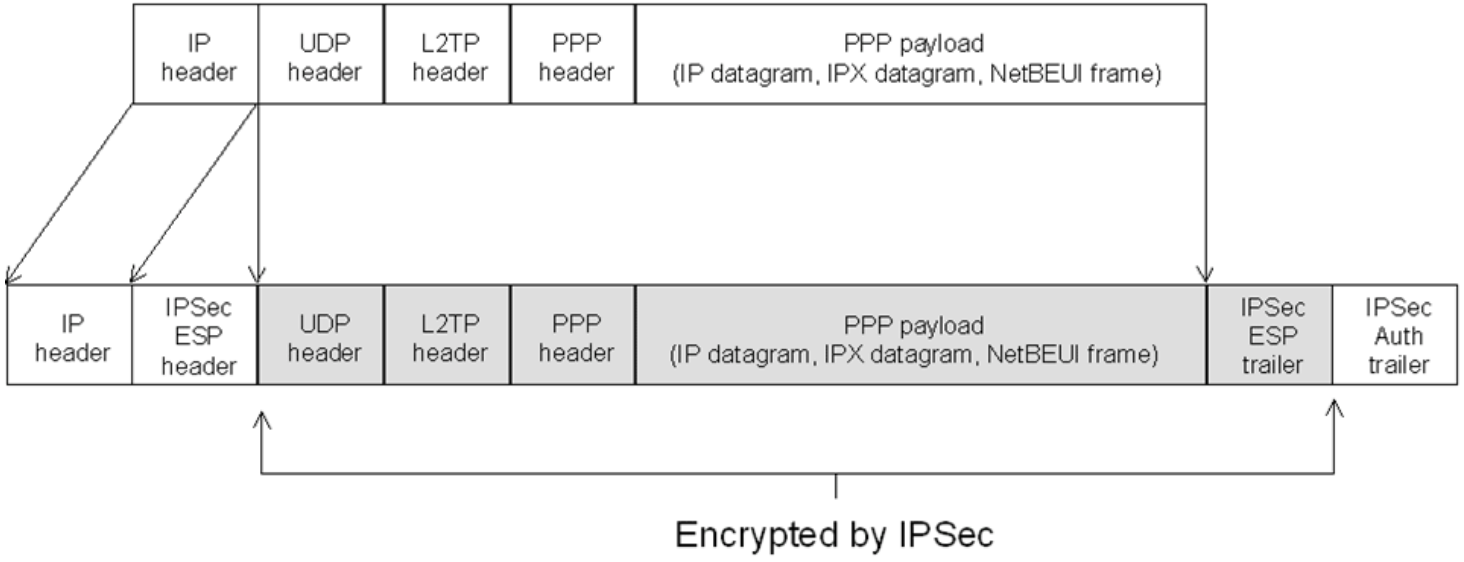
L2TP

- Layer Two Tunneling Protocol
  - PPTP+L2F (Layer Two Forwarding)
  - L2TP over IP internetworks uses UDP and a series of L2TP messages for tunnel maintenance.
  - A tunnel can contain multiple connection at once.



L2TP/IPsec

- Usually use IPsec ESP (Encapsulating Security Payload) to encrypt the L2TP packet.
  - Data encryption begins before the PPP connection process by negotiating an IPSec security association.
  - Require computer-level authentication using computer certificates.

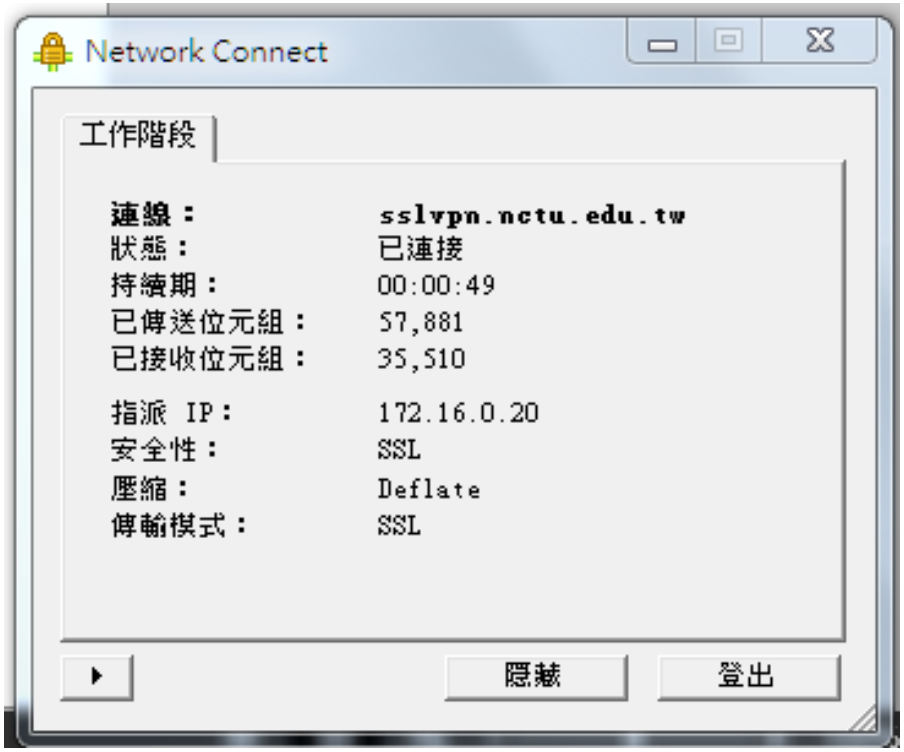


IPsec Tunnel Mode

- Internet Protocol Security Tunnel Mode
  - IPSec tunnel mode encapsulates and encrypts entire IP packets, and the encrypted payload is then encapsulated again with a plain-text IP header.
- Internet Key Exchange (IKE)
  - ISAKMP+OAKLEY
- Two functions that ensure confidentiality:
  - Authentication Header (AH)
    - Provide source authentication and integrity without encryption.
  - Encapsulating Security Payload (ESP)
    - Provide both data authentication, data integrity and data encryption.

SSL VPN

- A form of VPN that can be used with a standard Web browser.
- The traffic is encrypted with the SSL protocol or Transport Layer Security (TLS) protocol.



mpd

Multi-link PPP daemon

mpd

- http://mpd.sourceforge.net/
- An implementation of the multi-link PPP protocol for FreeBSD.
- Support PPP over PPTP or L2TP.
- PAP, CHAP, MS-CHAP or EAP authentication.

- /usr/ports/net/mpd5
- pkg install mpd5

mpd

- /etc/rc.conf

```
gateway_enable="YES"
mpd_flags="-b"
mpd_enable="YES"
```

- startup

```
sysctl net.inet.ip.forwarding=1
/usr/local/etc/rc.d/mpd5 {start|stop|restart|rcvar|status}
```

mpd.secret

- /usr/local/etc/mpd/mpd.secret
  - Syntax: username password ip_address

```
hlku        “5566neverdie"
darkgerm    “lolisoul“      192.168.55.66
gluecrow    “yacwu”         192.168.99.0/24
```

  - plain text
  - chmod 600 mpd.secret

mpd.conf

- /usr/local/etc/mpd/mpd.conf
  - Consists of a *label* followed by a sequence of mpd commands.
  - A label begins at the first column and ends with a colon character.
  - Commands are indented with a tab character and follow the label on the next and subsequent lines.

```
client:
        create bundle template B1
        create link static L1 modem
        set modem device /dev/cuad0
        set modem speed 115200
        set modem script DialPeer
        set modem idle-script AnswerCall
        set modem var $DialPrefix "DT"
```

```
set modem var $Telephone "1234567"
set link no pap chap eap
set link accept pap
set auth authname "MyLogin"
set auth password "MyPassword"
set link max-redial 0
set link action bundle B1
open
```

mpd.conf

- startup section
  - Added a new startup section to the config-file, wich is loaded once at startup.

```
startup:
        # configure mpd users
        set user hlku 123456
        # configure the console
        set console self 127.0.0.1 4567
        set console open
        # configure the web server
        set web self 0.0.0.0 5566
        set web open
```

mpd.conf

- default section
  - Set interface
    - ip range
  - Set bundle name
  - Link layer configuration

mpd layers

interface -> ipcp -> compression
-> encryption -> bundle -> links

```
default:
        load pptp_server

pptp_server:
        # Define dynamic IP address pool.
        set ippool add pool123 192.168.1.30 192.168.1.110
        # Create clonable bundle template
        create bundle template VPN
```

mpd.conf

- default section

```
…(cont'd)
set iface enable proxy-arp
set iface idle 1800
# adjust incoming and outgoing TCP SYN segments (MTU)
set iface enable tcpmssfix
# Van Jacobson TCP header compression
set ipcp yes vjcomp
# Specify IP address pool for dynamic assigment.
set ipcp ranges 192.168.1.1/32 ippool pool123
```

mpd.conf

- default section

```
…(cont'd)
# Create clonable link template named L
create link template VPNLINK pptp
# Set bundle template to use
set link action bundle VPN
# Multilink adds some overhead, but gives full 1500 MTU.
set link enable multilink
# Address and control field compression, save 2 bytes,
# Protocol field compression, save 1 byte
set link yes acfcomp protocomp
set link keep-alive 10 60

# Configure PPTP
set pptp self 140.113.x.x
set link enable incoming
```

mpd.conf - encryption

- Microsoft Point-to-point compression (MPPC) CCP subprotol
  - 'mppc' option should be enabled at the CCP layer

```
# The five lines below enable Microsoft Point-to-Point encryption
# (MPPE) using the ng_mppc(8) netgraph node type.
    set bundle enable compression
    set ccp yes mppc
    set mppc yes e40
    set mppc yes e128
    set mppc yes stateless
```

mpd.conf

- Minimum configuration

```
startup:
default:
        set ippool add pool123 192.168.1.31 192.168.1.35
        create bundle template NAVPN
        set ipcp ranges 192.168.1.1/32 ippool VPNPOOL
        create link template VPNLINK pptp
        set link action bundle NAVPN
        set link no pap chap eap
        set link enable chap-msv2
        set pptp self 140.113.x.x
        set link enable incoming
```

mpd

- /etc/syslog.conf

```
!mpd
*.* /var/log/mpd.log
```

- touch /var/log/mpd.log
- /etc/rc.d/syslogd reload

- Maybe firewall need some configuration.
  - Allow 1723 port, and GRE packets.

OpenVPN

OpenVPN

- OpenVPN is an open-source software application that implements VPN techniques.
- OpenVPN uses a custom security protocol that utilizes SSL/TLS for key exchange.
- https://openvpn.net/index.php/open-source.html

- /usr/ports/security/openvpn
- pkg install openvpn

OpenVPN

- Set environment variables first

```
setenv D `pwd`
setenv OPENSSL /usr/bin/openssl
setenv KEY_CONFIG /usr/local/share/easy-rsa/openssl-1.0.0.cnf
setenv KEY_DIR /usr/local/etc/openvpn/keys
setenv KEY_SIZE 1024
setenv KEY_COUNTRY TW
setenv KEY_PROVINCE TW
setenv KEY_CITY HsinChu
setenv KEY_ORG “NCTUCSCC"
setenv KEY_EMAIL hlku@cs.nctu.edu.tw
```

OpenVPN

- cd /usr/local/share/easy-rsa
- generate certificates

```
./build-ca
./build-key-server server
./build-dh
```

- Then move the keys to /usr/local/etc/openvpn/keys .

OpenVPN

- /usr/local/etc/openvpn/server.conf

```
port 1194
proto udp
dev tap              # firewall setting
.......
ca /usr/local/etc/openvpn/keys/ca.crt
cert /usr/local/etc/openvpn/keys/server.crt
key /usr/local/etc/openvpn/keys/server.key
dh /usr/local/etc/openvpn/keys/dh1024.pem

server 10.0.1.x 255.255.255.0
```

OpenVPN

- /etc/rc.conf

```
openvpn_enable="yes"
openvpn_if="tap"
openvpn_configfile="/usr/local/etc/openvpn/server.conf"
```

- startup

```
sysctl net.inet.ip.forwarding=1
/usr/local/etc/rc.d/openvpn start
```

Appendix

Appendix

- Seven Myths about VPN Logging and Anonymity
- https://technet.microsoft.com/zh-tw/library/bb742566.aspx