Introduction to OpenVPN

Practical Use of OpenVPN to Secure Remote Networks

Hi!
Introduce Yourselves!

1. What’s your name?
2. Where are you from?
3. What is your workstation platform of choice? What did you bring for use today?
4. What brings you to BSDCan 2012?
5. How familiar are you with OpenVPN?
   • None
   • Novice
   • Expert
6. What, in particular, are you hoping to learn by attending this OpenVPN tutorial?
7. Post-conference beverage of choice?

What is a VPN?

• How VPNs Are Used:
  – Connect Multiple Networks
  – Connect Client Devices to Remote Networks
  – Provide Authentication and Confidentiality

• VPNs Are NOT:
  – TOR!

• Why Use A VPN?
  – Keep Private Traffic Private
  – Create a Remote Endpoint on a LAN
  – Secure Communication on a Hostile Network (WiFi/Coffee Shops/Girl/Boy-Friend/Mom & Dad)
What OpenVPN is **NOT**

- Internet Anonymizer (private browsing)
- NAT appliance/replacement
- Firewall (some filtering)
- Policy-Based Routing
- PPTP, IPSec, Cisco SSL, etc.
- SSL CA Management Suite

What OpenVPN **IS**

- Creates Secure Point-to-Point Tunnels Using SSL
- Ethernet (Layer 2) Traffic
- IP/TCP/ICMP/etc (Layer 3)
- OpenVPN Can:
  - Push Routes
  - Assign IP (v4 & v6 (soon))
  - Encrypt, or Not (up to you)
  - Basic Filtering (really really basic)
  - Authenticate Users (PAM, LDAP, Others)
  - Track Usage/Statistics (with help)

OpenVPN Usage

- Client/Server Model
  - Optionally, single (point-to-point) connection, like IPSec

  **SERVER:**
  I. Authenticate Clients
  II. Route Specific Traffic
  III. Layer 2/3 can Be Filtered (pf/ipfw/etc)
  IV. **ALL** Client -> VPN Traffic Routes Through Server

  **CLIENT:**
  I. Same Binary as Server, Different Config
  II. Based on Server Config, CAN Route All Traffic Through VPN

The OpenVPN Community

- James Yonan (founder)
- OpenVPN Technologies, Inc
- Key Players:
  - David Sommerseth
  - Samuli Seppänen
  - Gert Doering
  - Alon Bar-Lev
  - Heiko Hund
  - Eric F Criz
  - Jan Just Keijser
  - Kzree King
- Testing & Snapshots:
  - Progress Toward Testing Framework
  - Source Snapshots Available Weekly
    - FreeBSD net/openvpn-devel Updated Regularly
The OpenVPN Community

- Help Needed:
  - Developers!
  - Help on Specific Architectures (Linux, SPARC, *BSD, Embedded, Windows, etc)
  - GUI/Interface
  - Graphics
  - TESTING TESTING TESTING!
- Forum
  - Moderators
  - Contributors
- IRC
  - Contributors
  - Documentation
  - Yes! Please.
- Resources:
  - IRC: #openvpn & #openvpn-devel on Freenode (irc.freenode.net)
  - Forum: https://forums.openvpn.net
  - Wiki/Community Site: https://community.openvpn.net
  - Mailing Lists: http://openvpn.net/mail.html

Tutorial Outline

- Routed Server Setup
  - basic routed server configuration
  - OpenVPN configuration
  - FreeBSD rc.conf configuration
  - client OpenVPN configuration
  - ssl-admin and certificate generation
- Connecting Clients
  - connect attendee laptops to demonstration servers
  - ping other attendee vpn IPs
  - view VPN web server
- Connecting Networks
  - connect demonstration networks together
  - ping between separate VPN endpoints
  - view other VPN web servers

Bridged VPN Demonstration

Daemon
port 1194
proto udp
dev tap
tc /usr/local/etc/openvpn/ca.crt
cert /usr/local/etc/openvpn/example.crt
key /usr/local/etc/openvpn/example.key
dh /usr/local/etc/openvpn/dh2048.pem
server-bridge 10.0.5.1 255.255.255.0 10.0.5.20 10.0.5.50
script-security 2
up /usr/local/etc/openvpn/up.sh
client-to-client
keepalive 10 120
user vpn
group vpn
float
persist-key
persist-tun
status /var/openvpn/openvpn-status.log
log-append /var/log/openvpn.log
verb 2
management 127.0.0.1 1194

Tutorial Outline

- Other Information
  - revoking SSL certificates
  - PAM/LDAP authentication
  - logs and trouble-shooting
  - management interface
  - connection statistic tracking
  - starting/stopping OpenVPN
  - IPv6 support

ClaimLynx
OPENVPN
BSDCan 2012
Bridged VPN Demonstration

`#!/bin/sh
/sbin/ifconfig tap0 up`

```
closed Interfaces: "bridge tap0"
ifconfig bridge "inet 10.0.5.1 netmask 255.255.255.0 addm em0 addm tap0 up"
ifconfig bridge_all="10.0.4.14/16"
ifconfig tap0="up"
```

- Primary problem with bridged setups is tap0 isn’t ‘up’ administratively.
- Passes all ethernet frames, potential for broadcast storms/loops!

Tutorial WiFi

`bsdcant_pub: bsdcant_openvpn`
`bsdcant_XX: bsdcant_openvpn`
`srvv1XX.example.org – Server`
`lan.v1XX.example.org – LAN IP`
User: root
Pass: password
LAB 1: Client → Server

1) Create OpenVPN server/client configuration
2) ssl-admin: setup & generate certificates
3) Install client and certificates on group machines
4) Connect to VPN and test

```bash
# server.conf
daemon
port 1194
proto udp
dev tun
cert /usr/local/etc/openvpn/openvpn-server.crt
cert /usr/local/etc/openvpn/openvpn-server.key
dh /usr/local/etc/openvpn/dh1024.pem
server 10.60.VLAN.0 255.255.255.0
push "route 192.168.VLAN.0 255.255.255.0"
topology net30
script-security 2
crl-verify /usr/local/etc/ssl-admin/prog/crl.pem
comp-lzo 10
persist-tun
persist-key
status /var/log/openvpn-status.log
verb 5
management 127.0.0.1 1194
```
LAB 1: Client → Server

/usr/local/etc/openvpn/client.conf

client
dev tun
proto udp
remote srv.v1.example.org
resolv-retry infinite
nobind
persist-key
persist-tun
remote-cert-tls server
cert client.crt
key client.key
verb 3

ssl-admin

- Easy-RSA is included with OpenVPN, but it sucks.
- security/ssl-admin
  - Fast, interactive.
  - No bulk support (yet)
  - Written in Perl
  - Maintains CRL
  - Can bundle certificate, key, CA cert, and OpenVPN config

ssl-admin

Example output:

Please select the menu option from the following list:  
1) Update run-time options:  
   Common Name: "John Doe"  
   Key Duration (days): 3650  
   Current Serial #: 01  
   Key Size (bits): 1024  
   Intermediate CA Signing: NO
2) Create new Certificate Request
3) Sign a Certificate Request
4) Perform a one-step request/sign
5) Revoke a Certificate
6) Renew/Re-sign a past Certificate Request
7) View current Certificate Revocation List
8) View index information for certificate.
z) Zip files for end user.
dh) Generate Diffie Hellman parameters.
CA) Create new Self-Signed CA certificate.
S) Create new Signed Server certificate.
q) Quit ssl-admin

Menu Item:

Main Menu:
LAB 1: Client → Server

- copy openssl.conf.default and ssl-admin.conf.default to non-default names
- create symbolic link from /usr/local/etc/openvpn/client.conf to /usr/local/etc/ssl-admin/packages/client.ovpn
- run ssl-admin and
  - create CA (auto, at startup)
  - create Diffie-Hellman key (option dh)
  - create server cert/key (option S)
- from /usr/local/etc/ssl-admin/active, copy the following to /usr/local/etc/openvpn:
  - ca.crt
  - openvpn-server.crt
  - openvpn-server.key
- from /usr/local/etc/ssl-admin, copy dh1024.pem to /usr/local/etc/openvpn
- edit server.conf for proper names/path of SSL certificates and keys

LAB 1: Client → Server

- ssl-admin
  - Generate CA certificate/key
  - Generate client certificate/keys for all groups
  - CERTIFICATE PASSWORDS? Up to you.
    - Need to be entered every time they’re used!
  - Distribute client packages (zip files) to group
  - Start OpenVPN:
    - # openvpn --config /usr/local/etc/openvpn/server.conf

LAB 1: Client → Server

Client Install
http://control.example.org/files/

Certificate Import/Install

LAB 1: Client → Server

- Once connected to the VPN, check the following:
  1. See web page http://lan.v1VLAN.example.org
  2. cat /var/log/openvpn-status.log, should see your connection listed.
- net30/subnet (topology):
  - net30 gives network blocks of 4 IPs
    - 10.60.1.0, 10.60.1.4, 10.60.1.8, etc
    - 10.60.1.6, 10.60.1.10, etc for VPN client IPs
  - subnet gives incremental client numbering
    - 10.60.1.1, 10.60.1.2, 10.60.1.3, etc
LAB 1: Client → Server

QUESTIONS?

LAB 2: Network → Network

- Groups are 1 & 2, 3 & 4, 5 & 6, etc
- Odd = server, even = client
- Connect two separate networks with OpenVPN such that the LAN and OpenVPN clients on either network can talk with the LAN and OpenVPN client on the other network

LAB 2: Network → Network

- ssl-admin: create client certificate/key pair for even-group’s server
- create client-config-dir and ccd entry for remote network
- update server config to support remote network and ccd
# We need to identify the networks BEHIND this client

c```
iroute 192.168.1EVEN.0 255.255.255.0
iroute 10.60.EVEN.0 255.255.255.0
```
LAB 2: Network → Network

QUESTIONS?

LAB 3: PAM Authentication

• configure OpenVPN server to require username/password
• configure OpenVPN client to prompt user for username/password
• enable-password-save/ --auth-user-pass
  • bug in configure scripts for this option -- fix in the pipe
• --username-as-common-name
  • use passed username instead of certificate CN
• --client-crt-not-required
  • still encrypted!
• operates like HTTPS, user/password important!

LAB 3: PAM Authentication

ODD EXAMPLE /usr/local/etc/openvpn/server.conf

daemon
port 1194
proto udp
dev tun
cert /usr/local/etc/openvpn/openvpn-server.crt
dh /usr/local/etc/openvpn/dh1024.pem
server 10.60.1.0 255.255.255.0
push "route 192.168.1ODD.0 255.255.255.0"
push "route 192.168.1ODD.0 255.255.255.0"
push "route 10.60.ODD.0 255.255.255.0"
topology net30
script-security 2
client-to-client
crl-verify /usr/local/etc/ssl-admin/prog/crl.pem
keepalive 10 120
verb 5
management 127.0.0.1 1194
plugin /usr/local/lib/openvpn-auth-pam.so "login login USERNAME password PASSWORD"

EVEN EXAMPLE /usr/local/etc/openvpn/server.conf

daemom
port 1194
proto udp
dev tun
cert /usr/local/etc/openvpn/openvpn-server.crt
dh /usr/local/etc/openvpn/dh1024.pem
server 10.60.1.0 255.255.255.0
push "route 192.168.1EVEN.0 255.255.255.0"
push "route 192.168.1EVEN.0 255.255.255.0"
push "route 10.60.EVEN.0 255.255.255.0"
topology net30
script-security 2
client-to-client
crl-verify /usr/local/etc/ssl-admin/prog/crl.pem
keepalive 10 120
verb 5
management 127.0.0.1 1194
plugin /usr/local/lib/openvpn-auth-pam.so "login login USERNAME password PASSWORD"
LAB 3: PAM Authentication

REGULAR VPN CLIENTS: client.ovpn

- client
dev tun
proto udp
remote srv.v101.example.org
resolv-retry infinite
nobind
persist-key
persist-tun
remote-cert-tls server
cert client.crt
key client.key
verb 3
auth-user-pass

SERVER/ROUTER VPN CLIENTS: client.ovpn

- client
dev tun
proto udp
remote srv.v101.example.org
resolv-retry infinite
nobind
persist-key
persist-tun
remote-cert-tls server
cert client.crt
key client.key
verb 3
auth-user-pass pw.txt

SERVER/ROUTER VPN CLIENTS: pw.txt

- vpnuser
password

LAB 3: PAM Authentication

- Restart OpenVPN server
- re-connect OpenVPN clients (with updated config)
  - should be asked for user/pass to connect
  - User: vpnuser  Password: password
  - User: root will fail (secure-tty)
  - EVEN server will send contents of pw.txt
- Verify connectivity (same as end of Lab 2)
LAB 3: PAM Authentication

QUESTIONS?

/Lusr/local/etc/openvpn/cci/DEFAULT
push "redirect-gateway def1"

• DEFAULT applies to all clients WITHOUT entry in client-config-dir
• Generally, do NOT want to push redirect-gateway to remote LAN systems
• If no client-config-dir directive, put in server.conf
• Verify by going to http://control.example.org - IP should be that of your VLAN server

LAB 4: Default Gateway & PF

• use pf to NAT traffic from VPN to internet – don’t forget to /etc/rc.d/pf reload

PF Macros:

wan_if="em0"
lan_if="em1"
stor_if="em2"
vpn_if="tun0"
stor_srv="172.16.16.3"
ctrl_srv="172.16.16.2"

PF Tables:

table <self> {self}

PF Options:

no block/policy errors
no skip on lo

set block-policy return
set skip on lo

nat on $wan_if from 10.60.160.0/24 -> $wan_if:0

## Filtering

pass all

#block log all

# Only traffic on storage should be to NFS server or DHCP.

#block in log on $stor_if all

#pass on $stor_if from {$stor_srv $ctrl_srv} to <self>

pass in inet proto tcp from any to <self> port 22

pass inet proto icmp

# Block connections from the Internet

block in log on $wan_if from any to $lan_if:network

push "redirect-gateway def1"

## Macros

wan_if="em0"
lan_if="em1"
stor_if="em2"
vpn_if="tun0"
stor_srv="172.16.16.3"
ctrl_srv="172.16.16.2"

PF Tables:

table <self> {self}

PF Options:

no block/policy errors
no skip on lo

set block-policy return
set skip on lo

nat on $wan_if from 10.60.160.0/24 -> $wan_if:0

## Filtering

pass all

#block log all

# Only traffic on storage should be to NFS server or DHCP.

#block in log on $stor_if all

#pass on $stor_if from {$stor_srv $ctrl_srv} to <self>

pass in inet proto tcp from any to <self> port 22

pass inet proto icmp

# Block connections from the Internet

block in log on $wan_if from any to $lan_if:network
LAB 4: Default Gateway & PF

Questions?

LAB 5: Auto-Start OpenVPN at Boot

- rc script supports multiple instances of OpenVPN
- for each additional instance beyond the first, symlink the /usr/local/etc/rc.d/openvpn script to openvpn_foo, openvpn_bar, etc
- rc.conf options are named to match:
  - openvpn_foo_enable="NO"
  - openvpn_foo_flags=
  - openvpn_foo_configfile="/usr/local/etc/openvpn/NAME.conf"
  - openvpn_foo_dir="/usr/local/etc/openvpn"

OpenVPN Management Interface

- designed for programmatic control/information from other programs/scripts
- CAN connect via telnet
- type ‘help’ at prompt for commands and options
- short list of commands/functions:
  1. kill specific client instances
  2. statistics
  3. modify running logging verbosity
  4. traffic bytes by client id
  5. display log real-time past N lines
Conclusion

Covered Topics/Labs:
1. ss1-admin for CA/Certificate Management
2. Client to Server VPNs
3. Connecting multiple networks with OpenVPN
4. PAM authentication with OpenVPN for clients
5. Using OpenVPN as a default gateway for clients
6. Auto-start OpenVPN on boot
7. Overview of OpenVPN management interface