OSDP Test VM Operating Guide

2016 Edition Revised 29Feb16

Rodney Thayer rodney@smithee.us

Table of Contents

Overview

- it's a 64-bit debian 7 vm.
- it's shipped as an ova file, exported from virtualbox. It uses the "1.0" format so it should import into VMWare.
- it's set up to just run. You don't need to operate it. It was meant to be useful to share as a fixed-function VM e.g. useful for any dev team, linux or not.
- all the sources are there, you can reconfigure it to run as a CP or PD, client or server, TLS or TCP or 485

Installation

TCP Configuration

- runs on port 10001
- uses passphrase for authentication

TLS Configuration

- runs on port 10443
- fixed certs used, files shipped on the vm. test root included. See below to generate certificates.
- uses gnutls latest configuration, supports TLS 1.2, AES-128-GCM
- configured out of the box with the PD as the server.

OSDP Configuration

assumes pd is at address 0.
no secure channel (secured instead by TLS. TCP implemented for diagnostic use only.)

vm set-up

```
ip address 10.0.0.200
gateway 10.0.0.1
dns: none
login: user osdp, password osdp.
  You have sudo.  Root password is also osdp.
```

vm configuration:

512meg
32gig hard disk (dynamic, expands to about 2.5 gig as released)
2 ethernet interfaces
various other vestigial vm configuration items e.g. floppy may be present but are not used.

vmware installation

(tested with vmware 12)

- make sure your vmware configuration (using the virtual network editor) is set up to have an appropriate interface (which might be bridged to an external ethernet connection.)
- open .ova file.
- click "retry" when it reports it fails compliance checks
- make sure there are 2 network interfaces and the first one is the one you'll use for OSDP and the second one is the NAT interface

virtualbox installation:

1. make sure virtualbox is running on your host platform
2. import the vm
3. make sure the vm's first ethernet interface is bridged to a real ethernet
4. start the vm
5. log into the vm and confirm the interface name is correct. it likely changes upon vm import. use procedure in appendix B below ("change ethernet interface name") to fix if necessary.
6. ping 10.0.0.200 from elsewhere on the network to confirm connectivity. It does respond to ping. Note: by default it's got ports 22,80 and 10001 open if you nmap scan it.

vm operation

- use a browser to access http://10.0.0.200/open-osdp-control.html. There is a primitive html/cgi-based UI on an apache web server on port 80. No authentication, this is an un-insulated engineering testbed.
- you can control the PD or the CP.
- the vm as shipped is set up to be the PD, and to answer network connections
- the "display status" link will spawn a second page with current status. This is updated approximately every time there is a pol/response sequence. Not all the counters work as visible.
- OSDP is started at boot time by /etc/rc.local calling /opt/open-osdp/bin/up.open-osdp. You can change the scripts to change the IP address, PD/CP role, TLS or TCP.

Demonstration

- start the vm
- ping it to confirm it's alive
- go to the web UI
- start the status page
- from a CP, connect to the PD.
    - console access can show the incoming connection
    - use network trace tool to show the TCP or TLS traffic
- from the PD, present a badge (raw format, 26 bit) - use the web UI and click "present card". it uses the card data in /opt/open-osdp/run/PD/open-osdp-params.json
- if you run two instances back to back you can demo CP to PD communications, in which case the CP status page will show incoming cardholder data.

Contents

- ova file, in 1.0 format, with a manifest (should work with vmware workstation 10 or later)
- /home/osdp/setup is where everything is built
- includes open source
- includes platform set-up (apache config, etc.)
- libosdp ("1.0 build 6")
- runs out of /opt/open-osdp/run/PD
- started at boot time from /etc/rc.local (calls file /opt/open-osdp/bin/up.osdp)
- uses open source components: gnutls, nettle, libtasn, jansson
- the whole thing is there so you could run a second VM as the CP

Appendix

A. Colophon

B. Linux Shell Cheatsheet

Powering off the VM:

log in as user osdp (password osdp)

issue the power off command (requires password again)

```
sudo poweroff
```

Testing the network:

log in (username osdp, password osdp)

list the interfaces (prompt is for your password, osdp)

```
sudo ifconfig -a
```

ping a test destination (remember linux ping is continuous, limit it to 4 in this command)

```
ping -c 4 10.0.0.1
```

Change ethernet interface name:

log in as user osdp (password osdp)

sudo to bash (requires password again)

```
sudo bash
```

list current ethernet interfaces (you should have a "#" prompt as you're sudo to root)

```
ip link
```

It should show 3 lines (1: lo: ... 2: eth0: ... 3:eth1:...) If the second entry isn't "eth0" you need to edit the start-up script. That file is /opt/open-osdp/bin/up.osdp. Change the "eth0" to whatever ethernet interface you have set as the bridged interface.

Test Certificates

In /home/osdp/setup/test-ca there's a copy of the relevant bits of the openssl tool for creating certificates. Two shell scripts generate a CA and then several test certificates.

Certificate use:

- root.pem is copied to /opt/open-osdp/etc/ca_files.pem
- a matched pair of ..._key.pem and ..._cert.pem files is copied to /opt/open-osdp/etc/key.pem and cert.pem
- use separate key/certificate pairs for the cp and the pd
- the root certificate has to go on both CP and PD so they can each check the other's certificate.

- note that at least in libosdp's TLS implementation it checks the "common name" field against the name in the certificate. It doesn't do any DNS lookups, you do not need to have a DNS running for this to work.
- a CRL is available for certificate status checking.