Open Platform Trust Services (OpenPTS)
User’s Guide
Version 0.2.3

Seiji Munetoh

Mar 07, 2011

Copyright © 2011 IBM Corporation. All rights reserved
Mailing list for comments: openpts-users@lists.sourceforge.jp
Web access (preferred): http://sourceforge.jp/projects/openpts
# Contents

## 1 Introduction
1.1 Purpose ........................................ 1  
1.2 Scope ........................................ 1  
1.3 Architecture .................................. 1  
1.4 Operations .................................... 2  
1.5 Limitation ..................................... 2  

## 2 Use case 1. Standalone Remote Attestation
2.1 Setup the Collector (target platform) .................. 4  
2.2 Setup the Verifier (at localhost) ...................... 5  
2.3 Setup verifier (at remote host) ....................... 5  
2.4 Update Manifests ................................ 6  
2.5 Check the status ................................ 6  

## 3 OpenPTS Commands Usage
3.1 ptscd ........................................... 8  
3.2 openpts ......................................... 8  
3.3 uml2dot .......................................... 8  
3.4 rm2dot ........................................... 9  
3.5 iml2text ......................................... 9  
3.6 iml2aide ......................................... 9  
3.7 ir2text .......................................... 9  

## 4 OpenPTS Configuration Files
4.1 Files ............................................ 11  
4.2 /etc/ptscd.conf .................................. 12  
4.3 /.openpts/openpts.conf ............................ 12  
4.4 /.openpts/UUID/target.conf ....................... 12  

## 5 Configuration of Trusted Platform
5.1 RHEL 6 .......................................... 14  
5.1.1 GRUB-IMA .................................... 14  
5.1.2 Linux IMA ..................................... 14  
5.1.3 TrouSetS(TSS) ................................ 15  
5.2 Fedora 12 ........................................ 15  
5.2.1 GRUB-IMA .................................... 15  
5.2.2 Linux IMA ..................................... 16  
5.2.3 TrouSetS(TSS) ................................ 16  
5.3 Fedora 14 - TBD ................................ 16  
5.4 Ubuntu 10.04 .................................... 16  

## 6 Build OpenPTS
6.1 Linux RPM package ................................ 17  
6.2 Linux DEB package ................................ 17  
6.3 User’s Guide .................................... 17  
6.4 Design document .................................. 17  
6.5 API document .................................... 17  

## 7 Common errors and problems
7.1 tpm_takeownership is fail (0x0008) .................... 18  
7.2 Key generation is fail ................................ 18  
7.3 validation fail - POLICY-L010 ....................... 18  
7.4 0x803 Error ..................................... 18
1 Introduction

1.1 Purpose
The purpose of this User’s Guide is to provide a description of the usage of Open Platform Trust Services (OpenPTS).

1.2 Scope
System administrator and developer of Trusted Platform.

1.3 Architecture
Figure 1 shows brief overview of OpenPTS architecture. OpenPTS is used by both collector (target platform) and verifier sides. Collector side, 'ptscd' command is a daemon process which manage the integrity of target platform. Verifier side, 'openpts' command is used to validate the target platform by remote attestation. The protocol between ptscd and openpts is based on TCG IF-M protocol. OpenPTS setup the SSH tunnel between collector and verifier to secure the remote attestation. This figure shows stand-alone operation mode. OpenPTS supports IMC and IMV interfaces for TNC (Trusted Network Connect).

![OpenPTS Architecture Diagram](image)

Figure 1: OpenPTS - Architecture (Standalone Mode)
1.4 Operations

Figure 2 shows how OpenPTS manage the integrity. OpenPTS uses a model which describe the behabior of transitive trust chain of target platform. The model is Finite State Machine (FSM) written by UML state diagram. OpenPTS uses this model to parse the integrity measurement log (IML) and generate the reference manifest (RM).

The behavior model just describe the general behavior of transitive trust chain and is used to generate RM and integrity report (IR). OpenPTS supports generic model of x86(PC) platform. The binary model contains actual digest value of target and used to validate the IML.

By using the model, we can translate the binary measurement (hash value) into security properties. Then, translated properties are validated by given policies to get the final result, VALID/INVALID/UNKNOWN.

1.5 Limitation

- AIDE and TNC integration is still under development.
- Need to apply the patch to TrouSerS (TSS) to handle eventlog properly.
2 Use case 1. Standalone Remote Attestation

In this use case, we use individual reference manifest and integrity database for each target platform. Thus, the reference manifest and integrity database are created by collector running at the target platform. Fig 3 shows the operation flow of OpenPTS.

This use case have three operation phases as follows.

**Engagement phase** We trust an installation process\(^1\). The collector generate the new UUID to identify the target and reference manifest based on the measurement of initial boot. Thus, the reference manifests are based on actual BIOS\(^2\) and Operating System measurement at this phase. Verifier get the UUID and manifests from the Collector and securely stored them.

**Operation phase** Verifier validate the target (remote attestation).

**Update phase** After the BIOS or OS update, manifest must be updated. The OpenPTS collector (ptscd daemon) do selftest at the boot. If validation was faild due to the change, it generates the new manifest.

If the update was expected, Verifier update the manifest too.

---
\(^1\)If we have the EK credential of TPM, we can trust the remote platform.

\(^2\)OpenPTS generate manifest of actual measurement since there are no PC and BIOS vendors which disclose integrity information.
Engagement  Initial setup (Trusted environment)

Operation  Status check by Remote Attestation (Untrusted environment)

Update  Update the SW status (Trusted environment)

2.1  Setup the Collector (target platform)

Step 1, Take the TPM ownership. (with well known secret)

```
# tpm_takeownership -y -z
```

Step 2, Install openpts. (see the section X.X.X how to build)

```
# rpm -ivh openpts-0.2.3-1.x86_64.rpm
```

After the installation, adjust the configuration file `/etc/ptscd.conf` If you are using GRUB-IMA,

```
rm .num=2
runtime.model.pcr.4=grub_pcr4hdd.uml
runtime.model.pcr.5=grub_pcr5.uml
runtime.model.pcr.8=grub_pcr8.uml
```

If you enabled Linux-IMA

```
rm .num=2
runtime.model.pcr.10=ima_pcr10.uml
```

Also set the platform information. e.g.

```
platform.system.manufacturer=LENOVO
platform.system.productname=745749J
platform.system.version=ThinkPad X200
platform.bios.version=0DE1S8WW
```

Step 3a, Setup the AIDE database (OPTION)

You can create the sample AIDE DB from current IML. (It takes long time).

```
# iml2aide -c /etc/ptscd.conf -o /var/lib/aide/aide.db.gz
```

Step 3b, Setup the AIDE database (OPTION)

Or create the AIDE DB. (It takes long time too).

```
# cp /usr/share/openpts/aide.conf /etc/aide.conf
# aide --i
```

Step 4, Init ptscd. ()

e.g.

```
# /usr/sbin/ptscd --i
Generate uuid : 186bebba-2781-11e0-bcdb-001f160c9c28
Sign key location : SYSTEM
Generate UUID (for RM) : 19566e16-2780-11e0-bf2e-001f160c9c28
level 0 Reference Manifest : /var/lib/openpts/19566e16-...9c28/rm0.xml
level 1 Reference Manifest : /var/lib/openpts/19566e16-...9c28/rm1.xml
```
Step 5, Start ptscd daemon. ()

<table>
<thead>
<tr>
<th># service ptscd start</th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting ptscd: [ OK ]</td>
</tr>
</tbody>
</table>

2.2 Setup the Verifier (at localhost)

Step 6a, Engagement with Collector (at localhost) ()

First, you attest the local platform. Engagement with the local collector.

```
$ openpts
```

<table>
<thead>
<tr>
<th>localhost</th>
</tr>
</thead>
<tbody>
<tr>
<td>/usr/bin/openpts --localhost</td>
</tr>
<tr>
<td>/home/f0o/.openpts is missing. create [Y/n]: Y</td>
</tr>
<tr>
<td>Target : localhost</td>
</tr>
<tr>
<td>Collector UUID : 186bebba-2781-11e0-bcdb-001f160c9c28</td>
</tr>
<tr>
<td>Manifest UUID : 19566e16-2780-11e0-bf2e-001f160c9c28</td>
</tr>
<tr>
<td>manifest[0] : /home/f0o/.openpts/186bebba-...9c28/19566e16-...9c28/rm0.xml</td>
</tr>
<tr>
<td>manifest[1] : /home/f0o/.openpts/186bebba-...9c28/19566e16-...9c28/rm1.xml</td>
</tr>
<tr>
<td>configuration : /home/f0o/.openpts/186bebba-...9c28/target.conf</td>
</tr>
<tr>
<td>validation policy : /home/f0o/.openpts/186bebba-...9c28/policy.conf</td>
</tr>
</tbody>
</table>

```
target.conf, policy.conf and aide.ignore is automatically generated. You can modify them. rm0.xml, rm1.xml and aide.db.gz are received from collector. To override existing setting, use "-f" option.
```

```
$ openpts
```

See the Table 2 about the file used by openpts command.

Step 7a, Remote Attestation (at localhost) ()

```
$ openpts
```

<table>
<thead>
<tr>
<th>localhost</th>
</tr>
</thead>
<tbody>
<tr>
<td>/usr/bin/openpts --localhost</td>
</tr>
<tr>
<td>/home/f0o/.openpts is missing. create [Y/n]: Y</td>
</tr>
<tr>
<td>Target : localhost</td>
</tr>
<tr>
<td>Collector UUID : 186bebba-2781-11e0-bcdb-001f160c9c28</td>
</tr>
<tr>
<td>Manifest UUID : 19566e16-2780-11e0-bf2e-001f160c9c28</td>
</tr>
<tr>
<td>manifest[0] : /home/f0o/.openpts/186bebba-...9c28/19566e16-...9c28/rm0.xml</td>
</tr>
<tr>
<td>manifest[1] : /home/f0o/.openpts/186bebba-...9c28/19566e16-...9c28/rm1.xml</td>
</tr>
<tr>
<td>configuration : /home/f0o/.openpts/186bebba-...9c28/target.conf</td>
</tr>
<tr>
<td>validation policy : /home/f0o/.openpts/186bebba-...9c28/policy.conf</td>
</tr>
</tbody>
</table>

2.3 Setup verifier (at remote host)

Install openpts to the verifier box.

Step 2b, Install openpts. ()

```
# rpm --ivh openpts-0.2.3-1.x86_64.rpm
```

Step 6b, Engagement with Collector (at remote host) ()

```
$ openpts
```

Engagement with the target collector. It uses SSH portforward. you have to provide your username of remote host.

```
$ openpts --s --S 5568 --username hostname
```

Step 7a, Remote Attestation (at remote host) ()

```
$ openpts hostname
```
2.4 Update Manifests

Collector selftest the measurement and manifests. If validation based on the manifest was faild, the collector generate the new manifest for current measurements. This happen if you update any relevent components, such as the BIOS or kernel image.

Step 8, Accept the Collector change ()

```bash
$ openpts localhost
Target : localhost
Collector UUID : 1dbac28e–2787–11e0–b84a–001f160c9c28
Manifest UUID : 1df210fe–2787–11e0–b84a–001f160c9c28
port : 6678 (localhost)
policy file : /home/foo/.openpts/1dbac28e–2787–11e0–b84a–001f160c9c28/policy.conf
property file : /home/foo/.openpts/1dbac28e–2787–11e0–b84a–001f160c9c28/vr.properties
integrity : unknown (INTERNAL ERROR) rc=35

Reasons
0 Missing Reference Manifest (RM)
1 Collector hostname = localhost
2 Collector UUID = 1dbac28e–2787–11e0–b84a–001f160c9c28
3 Collector RM UUID = 33b88c38–2787–11e0–ad0–001f160c9c28
New reference manifest exist. If this is expected change, update the manifest by openpts --i --f

$ openpts --i --f --localhost
Target : localhost
Collector UUID : 1dbac28e–2787–11e0–b84a–001f160c9c28
Manifest UUID : 33b88c38–2787–11e0–ad0–001f160c9c28
manifest[0] : /home/foo/.openpts/1dbac28e–2787–11e0–b84a–001f160c9c28/rm0.xml
manifest[1] : /home/foo/.openpts/1dbac28e–2787–11e0–b84a–001f160c9c28/rm1.xml
configuration : /home/foo/.openpts/1dbac28e–2787–11e0–b84a–001f160c9c28/target.conf
validation policy : /home/foo/.openpts/1dbac28e–2787–11e0–b84a–001f160c9c28/policy.conf

$ openpts -p 5557 localhost
Target : localhost
Collector UUID : 1dbac28e–2787–11e0–b84a–001f160c9c28
Manifest UUID : 33b88c38–2787–11e0–ad0–001f160c9c28
port : 6678 (localhost)
policy file : /home/foo/.openpts/1dbac28e–2787–11e0–b84a–001f160c9c28/policy.conf
property file : /home/foo/.openpts/1dbac28e–2787–11e0–b84a–001f160c9c28/vr.properties
integrity : valid
```

2.5 Check the status

Collector side ()

```bash
# /usr/sbin/ptscd -D
openpts version 0.2.2.svn
config file : /etc/ptscd.conf
poer : 6678
UUID : 186bebba–...9c28 (/var/lib/openpts/uuid)
IML access mode : TSS
Runtime IML type : IMA (kernel 2.6.32)
RM UUID (current) : 19566e16–2780–11e0–bf2e–001f160c9c28
RM UUID (for next boot) : (null)
List of RM set
ID UUID date(UTC) status
0 d5086d88–...9c28 2011–01–24–05:50:21 state=UNKNOWN

Integrity Report
Model dir : /var/lib/openpts/ir.xml
Model dir : /usr/share/openpts/models
Behavior Models
```
Verifier side ()

$ /usr/bin/openpts --D
Show openpts config

<table>
<thead>
<tr>
<th>config file</th>
<th>: /home/foo/.openpts/openpts.conf</th>
</tr>
</thead>
<tbody>
<tr>
<td>uuid</td>
<td>: 69c9e458-2781-11e0-9b86-001f160c9c28</td>
</tr>
<tr>
<td>target [0]</td>
<td>uuid : 186bebba-2781-11e0-bcdb-001f160c9c28</td>
</tr>
<tr>
<td>target [0]</td>
<td>config : /home/foo/.openpts/186bebba-...9c28/target.conf</td>
</tr>
<tr>
<td>target [0]</td>
<td>hostname : localhost</td>
</tr>
<tr>
<td>target [0]</td>
<td>port : 6678</td>
</tr>
<tr>
<td>target [0]</td>
<td>SSH : off</td>
</tr>
</tbody>
</table>
3 OpenPTS Commands Usage

3.1 ptsd

PTS collector daemon.

Usage: ptsd [options] [command]

Commands: (for grand)
-  
  -i Initialize PTS collector
  -u Use HUP signal if the ptsd is running to update the RM
  -D Display the configuration

Miscellaneous:
-  
  -h Show this help message
  -v Verbose mode. Multiple -v options increase the verbosity.

Options:
-  
  -p port Set port number. default is 8078
  -c configfile Set configuration file. default is /etc/ptsd.conf
  -f Force init, delete existing target (collector) info
  -t Sort the output by timestamp
  -l username ssh username

3.2 openpts

PTS validation utility.

Usage: openpts [options] [command] target

Commands:
-  
  -i Initialize PTS verifier with target (collector)
  -u Update PTS verifier with target (collector)
  -D Display the configuration (ALL)

Miscellaneous:
-  
  -h Show this help message
  -v Verbose mode. Multiple -v options increase the verbosity.

Options:
-  
  -p port Set port number. default is 6678
  -c configfile Set configuration file. default is /etc/ptsd.conf
  -f Force init, delete existing target (collector) info
  -s ssh username

3.3 uml2dot

Generate dot file from the UML State Diagram model.

Usage: uml2dot [options] umlfile

Options:
-  
  -o output Set output file (default is stdout)

Example:
$ uml2dot -o pcr0.dot pcr0.uml
$ dot -Tpng pcr0.dot -o pcr0.png
$ eog pcr0.png
3.4 rm2dot

Generate dot file from Reference Manifest (RM). Select pcr index since the RM may contain multiple FSMs for each PCRs.

Usage: rm2dot [options] rmfile

Options
- `-o output` set output file (default is stdout)
- `-p pcrindex` set PCR index
- `-l level` set snapshot level (0 or 1)

Example:
$ rm2dot -p 0 -o pcr0.dot rm.uml
$ dot -Tpng pcr0.dot -o pcr0.png
$ eog pcr0.png

3.5 iml2text

Dump the eventlog in text. It take out the eventlog from TSS or securityfs file directly.

Usage: iml2text [options]

Options:
- `-i filename` Set binary eventlog file (at securityfs)
- `-p pcrindex` Select PCR (TSS)
- `-E` Enable endian conversion (BE->LE or LE->BE)
- `-h` Show this help message

Example:
$ iml2text
Idx PCR Type Digest EventData
0 0 0x00000008 1dfce7de0cf13cfff102b1eb01875f752d5090c [BIOS:EV...
1 0 0x00000001 1c41801dd329198e50a3d98040230095693e49b3 [BIOS:EV...
2 0 0x00000001 16f111792cb98a3de12f3adb0406fc04e7e5fca [BIOS:EV...
3 0 0x00000001 dd261ca7511a7daf9e16cb572318e8e5fbd22963 [BIOS:EV...

3.6 iml2aide

Convert IML to AIDE database.

Usage: iml2aide [option]

Options:
- `-c filename` Set config file
- `-i filename` Set IMA IML file. default, get IML via TSS
- `-r filename` Set AIDE DB file as reference of fullpathname
- `-o filename` Set output file (AIDE DB format, gzipped)
- `-w filename` Set output file (Ignore name list, plain text format)
- `-h` Show this help message

Example:
$ src/iml2aide -c /etc/ptsconf.conf -r /var/lib/aide/aide.db.new.gz \
-o /tmp/aide.db.gz
AIDE DB (ref) : 241826 entries ((/var/lib/aide/aide.db.new.gz) \
IML : 5681 events (TSS)
AIDE DB : 3986 entries (tests/data/Fedora12/aide.db.gz)

3.7 ir2text

Convert Integrity Report (IR) to text format or binary format (=IML).
OpenPTS command

Usage: ir2text [options]

Options:
- `i filename` Set IR file
- `o filename` Set output file, else stdout
- `P filename` Set PCR output file (option)
- `b` Binary, (Convert IR to IML)
- `E` Enable endian conversion (BE->LE or LE->BE)
- `h` Show this help message
4 OpenPTS Configuration Files

4.1 Files
OpenPTS generates and uses many files as described below. Table 6 lists the files used by collector (ptscd daemon). Table 2 lists the files used by verifier (openpts command). The verifier store the target information at the user’s home directory.

Table 1: Files - collector side, (ptscd command)

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/etc/ptscd.conf</td>
<td>configuration file of collector</td>
</tr>
<tr>
<td>/var/lib/openpts/uuid</td>
<td>uuid of this platform</td>
</tr>
<tr>
<td>/var/lib/openpts/rm_uuid</td>
<td>uuid of current manifest (=RM_UUID)</td>
</tr>
<tr>
<td>/var/lib/openpts/newrm_uuid</td>
<td>uuid of next boot-cycle manifest (=NEWRM_UUID) TBD</td>
</tr>
<tr>
<td>/var/lib/openpts/[RM_UUID]/rm0.xml</td>
<td>Reference Manifest (BIOS)</td>
</tr>
<tr>
<td>/var/lib/openpts/[RM_UUID]/rm1.xml</td>
<td>Reference Manifest (IPL and OS)</td>
</tr>
<tr>
<td>/var/lib/openpts/ir.xml</td>
<td>Integrity Report</td>
</tr>
<tr>
<td>/var/lib/aide/aide.db.gz</td>
<td>AIDE database file</td>
</tr>
</tbody>
</table>

Table 2: Files - verifier side (openpts command)

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOME/.openpts/openpts.conf</td>
<td>configuration file of verifier</td>
</tr>
<tr>
<td>HOME/.openpts/[COLLECTOR_UUID]/target.conf</td>
<td>configuration file of each target</td>
</tr>
<tr>
<td>HOME/.openpts/[COLLECTOR_UUID]/policy.conf</td>
<td>validation policy</td>
</tr>
<tr>
<td>HOME/.openpts/[COLLECTOR_UUID]/ir.xml</td>
<td>Integrity Report (XML)</td>
</tr>
<tr>
<td>HOME/.openpts/[COLLECTOR_UUID]/vr.properties</td>
<td>target properties</td>
</tr>
<tr>
<td>HOME/.openpts/[COLLECTOR_UUID]/[RM_UUID]/rm0.xml</td>
<td>Reference Manifest (BIOS) (XML)</td>
</tr>
<tr>
<td>HOME/.openpts/[COLLECTOR_UUID]/[RM_UUID]/rm1.xml</td>
<td>Reference Manifest (IPL and OS) (XML)</td>
</tr>
<tr>
<td>HOME/.openpts/[COLLECTOR_UUID]/aide.db.gz</td>
<td>AIDE database as Integrity Database</td>
</tr>
<tr>
<td>HOME/.openpts/[COLLECTOR_UUID]/aide.ignore</td>
<td>list of valid components not listed on AIDE database</td>
</tr>
</tbody>
</table>
### 4.2 /etc/ptscd.conf

Table 3: /etc/ptscd.conf

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>config.dir</td>
<td>/var/lib/openpts</td>
<td>Set location of ptscd data</td>
</tr>
<tr>
<td>srk.password.mode</td>
<td>known</td>
<td>SRK password is well known secret (20 bytes of zeros)</td>
</tr>
<tr>
<td></td>
<td>null</td>
<td>SRK password is null, SHA1(&quot;&quot;)</td>
</tr>
<tr>
<td>iml.mode</td>
<td>tss</td>
<td>Get IML via TSS</td>
</tr>
<tr>
<td></td>
<td>securityfs</td>
<td>Get IML from securityfs filesystem</td>
</tr>
<tr>
<td>runtime.iml.type</td>
<td>IMA32</td>
<td>kernel 2.6.32</td>
</tr>
<tr>
<td>rm.num</td>
<td>1</td>
<td>Number of manifest. 1: Platform only, 2: Platform and Runtime</td>
</tr>
<tr>
<td>rm.basedir</td>
<td>/var/lib/openpts/</td>
<td></td>
</tr>
<tr>
<td>ir.file</td>
<td>/var/lib/openpts/ir.xml</td>
<td></td>
</tr>
<tr>
<td>uuid.file</td>
<td>/var/lib/openpts/uuid</td>
<td></td>
</tr>
<tr>
<td>rm.uuid.file</td>
<td>/var/lib/openpts/rm_uuid</td>
<td></td>
</tr>
<tr>
<td>newrm.uuid.file</td>
<td>/var/lib/openpts/newrm_uuid</td>
<td></td>
</tr>
<tr>
<td>model.dir</td>
<td>/usr/share/openpts/models</td>
<td></td>
</tr>
<tr>
<td>platform.model.pcr.0</td>
<td>bios_pcr0.uml</td>
<td></td>
</tr>
<tr>
<td>runtime.model.pcr.4</td>
<td>grub_pcr4hdd.uml</td>
<td></td>
</tr>
<tr>
<td>platform.system.manufacturer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>platform.system.productname</td>
<td></td>
<td></td>
</tr>
<tr>
<td>platform.system.version</td>
<td></td>
<td></td>
</tr>
<tr>
<td>platform.bios.version</td>
<td></td>
<td></td>
</tr>
<tr>
<td>runtime.vendor.name</td>
<td>redhat</td>
<td></td>
</tr>
<tr>
<td>runtime.distro.name</td>
<td>rhel</td>
<td></td>
</tr>
<tr>
<td>runtime.distro.version</td>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>

### 4.3 /.openpts/openpts.conf

Table 4: /.openpts/openpts.conf

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>uuid.file</td>
<td>./uuid</td>
<td></td>
</tr>
<tr>
<td>verifier.logging.dir</td>
<td>/</td>
<td></td>
</tr>
</tbody>
</table>

### 4.4 /.openpts/UUID/target.conf
Table 5: /openpts/UUID/target.conf

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname</td>
<td>(hostname)</td>
<td>Target hostname</td>
</tr>
<tr>
<td>port</td>
<td>6678</td>
<td>Target port</td>
</tr>
<tr>
<td>ssh.mode</td>
<td>on</td>
<td>Use SSH tunnel</td>
</tr>
<tr>
<td></td>
<td>off</td>
<td>direct access (localhost)</td>
</tr>
<tr>
<td>ssh.username</td>
<td>(foo)</td>
<td>SSH account name</td>
</tr>
<tr>
<td>ssh.port</td>
<td>(6680)</td>
<td>SSH tunneling port</td>
</tr>
<tr>
<td>target.uuid</td>
<td></td>
<td>UUID string</td>
</tr>
<tr>
<td>target.pubkey</td>
<td>(base64)</td>
<td>Public Key</td>
</tr>
<tr>
<td>ima.validation.mode</td>
<td>none</td>
<td></td>
</tr>
<tr>
<td>rm.num</td>
<td>1 or 2</td>
<td>Number of Manifest</td>
</tr>
<tr>
<td>rm.basedir</td>
<td>./</td>
<td></td>
</tr>
<tr>
<td>rm.uuid.file</td>
<td>./rm_uuid</td>
<td></td>
</tr>
<tr>
<td>newrm.uuid.file</td>
<td>./newrm_uuid</td>
<td></td>
</tr>
<tr>
<td>oldrm.uuid.file</td>
<td>./oldrm_uuid</td>
<td></td>
</tr>
<tr>
<td>ir.file</td>
<td>./ir.xml</td>
<td></td>
</tr>
<tr>
<td>prop.file</td>
<td>./vr.properties</td>
<td></td>
</tr>
<tr>
<td>policy.file</td>
<td>./policy.conf</td>
<td></td>
</tr>
<tr>
<td>verifier.logging.dir</td>
<td>./</td>
<td></td>
</tr>
</tbody>
</table>
5 Configuration of Trusted Platform

Unfortunately, there is no Linux distribution which configure the Trusted Platform well.

Table 6: Linux distribution and TC support

<table>
<thead>
<tr>
<th>OS</th>
<th>Kernel</th>
<th>CONFIG_IMA</th>
<th>IPL</th>
<th>SRTM</th>
<th>DRTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fedora 12</td>
<td>2.6.32</td>
<td>Yes</td>
<td>Grub-0.97</td>
<td>patch</td>
<td>NA</td>
</tr>
<tr>
<td>Fedora 13</td>
<td>2.6.34</td>
<td>Yes</td>
<td>Grub-0.97</td>
<td>patch</td>
<td>NA</td>
</tr>
<tr>
<td>Fedora 14</td>
<td>2.6.35</td>
<td>Yes</td>
<td></td>
<td>TBD</td>
<td></td>
</tr>
<tr>
<td>Fedora 15</td>
<td>2.6.3X</td>
<td>Yes</td>
<td></td>
<td>TBD</td>
<td></td>
</tr>
<tr>
<td>RHEL 6.0</td>
<td>2.6.32</td>
<td>Yes</td>
<td>Grub-0.97</td>
<td>NA</td>
<td></td>
</tr>
<tr>
<td>Ubuntu 10.04LTS</td>
<td>2.6.32</td>
<td>No</td>
<td>Grub2</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Ubuntu 10.10</td>
<td>2.6.35</td>
<td>No</td>
<td>Grub2</td>
<td>NA</td>
<td>OK</td>
</tr>
</tbody>
</table>

5.1 RHEL 6

SRTM based Trusted Boot (BIOS, no UEFI) and IMA could be enabled.

5.1.1 GRUB-IMA

Download source code "grub-0.97-68.el6.src.rpm" and patch.

```bash
$ su
$ c 'yum install ncurses-devel ncurses-static glibc-static glibc-devel-2.12-1.7.el6_0.3.i686 glibc-static-2.12-1.7.el6_0.3.i686
$ cd ~/rpmbuild/SOURCES
$ wget http://osdn.dl.sourceforge.jp/openpts/40294/grub-0.97-68.el6 ima-1.1.0.0.patch
$ cd ~/rpmbuild/SPECS

Modify grub.spec file as follows.

```
Release: 68%{dist}.ima
<snap>
%patch32: grub-0.97-68.el6.ima-1.1.0.0.patch
</snap>
%configure --sbindir=/sbin --disable-auto-link --mem-opt
<snip>
-enable-ima --datarootdir=%{_datarootdir}
</snip>
```

Build the RPM and install.

```bash
$ rpmbuild -ba grub.spec
$ su
c 'rpm -ivh ./RPMS/x86_64/grub-0.97-68.el6.ima.x86_64.rpm'
$ grep TCG /boot/grub/
Binary file /boot/grub/stage1 matches
Binary file /boot/grub/stage2 matches
```

5.1.2 Linux IMA

Add option "ima=on" at the kernel line in /boot/grub/grub.conf file.

If you have Intel TPM (Thinkpad X200, T400 etc), you also need additional options.

Add tpm.tis.ittmp=1 tpm.tis.force=1 tpm.tis.interrupts=0 ima=on at the kernel line

Set SELinux to permissive mode. System-\$Admin-\$SELinux management

If you don’t have /sys/kernel/security/ directory, please add following line to /etc/fstab

```
* securityfs /sys/kernel/security security securityfs rw 0 0
```
5.1.3 TrouSetS(TSS)

TrouSerS provided by RedHat, trousers-0.3.4-4.el6.x86_64 is old and can’t parse the eventlog created by Linux-IMA. Thus, you have to use the latest TrouSerS.

```
$ git clone git://trousers.git.sourceforge.net/gitroot/trousers/trousers
trouser --git
$ cd trousers
$ sh bootstrap.sh
$ ./configure
$ cd ..
$ ln -s trousers--git trousers-0.3.6.git
$ tar zcvf ~/rpmbuild/SOURCES/trousers-0.3.6.git.tar.gz ./trousers-0.3.6.git/*
```

# rpm --ivh --force trousers-0.3.6.git-1.x86_64.rpm

**notes** You may need to fix the package dependencies in trousers.spec

Modify /etc/tcsd.conf file as follows

```
firmware_log_file = /sys/kernel/security/tpm0/binary_bios_measurements
kernel_log_file = /sys/kernel/security/ima/binary_runtime_measurements
firmware_pcrs = 0,1,2,3,4,5,6,7,8
kernel_pcrs = 10
```

**notes** if you already taken the ownership and the system.data is missing, please copy the dummy system.data.

```
cp .. /dist/dummy\_tss\_system.data /var/lib/tpm/system.data
```

Ok, enable tcsd daemon.

```
chkconfig tcsd on
service tcsd start
```

5.2 Fedora 12

SRTM based Trusted Boot and IMA could be enabled.

5.2.1 GRUB-IMA

Download source code and patch.

```
$ su -c 'yumdownloader --source grub'
$ su -c 'yum-builddep grub-0.97-62.fc12.src.rpm'
$ rpm --ivh grub-0.97-62.fc12.src.rpm
$ cd ~/rpmbuild/SOURCES
$ wget http://osdn.dl.sourceforge.jp/openpts/40294/grub-0.97-62.fc12.ima-1.1.0.0.patch
$ cd ~/rpmbuild/SPRCS
```

Modify grub.spec file as follows.

```
+Release: 62%{?dist}.ima
+Patch2: grub-0.97-62.fc12.ima-1.1.0.0.patch
+%patch2 -p1
+%configure --bindir=/sbin --disable-auto-linux-mem-opt
   --enable-ima --datarootdir=%{datarootdir}
```

Build the RPM and install.

```
$ rpmbuild -ba grub.spec
$ su -c 'rpm --ivh .. /RPMS/x86_64/grub-0.97-62.fc12.ima.x86_64.rpm'
$ su -c 'grub-install /dev/sda'
```
5.2.2 Linux IMA

Add option “ima_tcb=1” at the kernel line in /boot/grub/grub.conf file.

If you have Intel TPM (Thinkpad X200, T400 etc), you also need additional options.
Add tpm_tis.itpm=1 tpm_tis.force=1 tpm_tis.interrupts=0 ima_tcb=1 at the kernel line
Set SELinux to permissive mode. System-Admin-SELinux management
if you don’t have /sys/kernel/security/ directory, please add following line to /etc/fstab

```
* securityfs /sys/kernel/security securityfs rw 0 0
```

5.2.3 TrouSetS(TSS)

Modify /etc/tcsd.conf file as follows

```
firmware_log_file = /sys/kernel/security/tpm0/binary_bios_measurements
kernel_log_file = /sys/kernel/security/ima/binary_runtime_measurements
firmware_pcrs = 0,1,2,3,4,5,6,7,8
kernel_pcrs = 10
```

5.3 Fedora 14 - TBD

TBD. try tboot (DRTM)

5.4 Ubuntu 10.04

SRTM based Trusted Boot covers BIOS only. You need to recompile the kernel to use the IMA.
6 Build OpenPTS

6.1 Linux RPM package
Install required packages to build.

```
# yum install libtool trousers-devel openssl-devel libxml2-devel libuuid-devel sqlite-devel
```

Build RPM package od OpenPTS.

```
$ sh bootstrap.sh
$ ./configure
$ make rpmbuild-ba
$ rpm -qpl /rpmbuild/RPMS/x86_64/openpts-0.2.2-1.x86_64.rpm
/etc/ptscd.conf
/etc/rc.d/init.d/ptscd
/usr/bin/imal2aide
/usr/bin/imal2text
/usr/bin/openpts
/usr/bin/rm2dot
/usr/bin/tpm_createkey
/usr/bin/uml2dot
<snip>
```

6.2 Linux DEB package
Ubuntu does not support IMA.

```
$ sh bootstrap.sh
$ ./configure
$ make dpkg-buildpackage
$ dpkg -i contents ../openpts_0.2.2_i386.deb
<snip>
```

6.3 User’s Guide
User’s guide is written in Latex. Install the latex environments before generate the document. (yum install tetex* for RPM)

```
$ cd doc
$ make ug
$ evince userguide.pdf
```

6.4 Design document

```
$ cd models
$ make png
$ cd ..
$ cd doc
$ make hldd
$ evince design.pdf
```

6.5 API document

```
$ cd doc
$ make ldd
$ firefox apidoc_html/index.html
```

17
7 Common errors and problems

7.1 tpm_takeownership is fail (0x0008)

Your TPM already taken the ownership. If you don’t know the owner password, you have to clear the TPM. To clear the TPM, Your PC needs cold boot, then enter the BIOS menu and clear the TPM.

7.2 Key generation is fail

Check the key storage file “/var/lib/tpm/system.data” If the size is zero, your install TSS after someone take the ownership. If you know the owner password. you can recover the storage file.

7.3 validation fail - POLICY-L010

PCR10 is changeed by IMA, comment out the policy file, ” ~/.openpts/{UUID}/policy.conf”

7.4 0x803 Error

This is TPM_DEFEND_LOCK_RUNNING error. Your TPM is defending against dictionaly attacks. And can be cleared by ’tpm_resedalock’ command with owner secret. However some TPM assert this flag without attack. the workaround is,

• take TPM owenweship with -y (known-secret) option.

• add ’tpm.resedtalock=on’ in /etc/ptscd.conf
Open Edition does not, however, process user identifiers or group identifiers even if they are set in Active Directory. For more information, visit the BeyondTrust website. Components. With one-way trusts, the authentication service uses RPC to look up domain users, groups, and security identifiers. With two-way trusts, lookup takes place through LDAP, not RPC. Installation Guide. The PowerBroker Identity Services Samba Guide describes how to use the tool to integrate Samba 3.0.25, 3.2.X, or 3.5.X with Enterprise or Open editions. Installation Guide. 20 Â© 2016. Open Platform Trust Services is a proof-of-concept (PoC) and reference implementation of Platform Trust Services (PTS) which is defined by the Trusted Computing Group, https://www.trustedcomputinggroup.org/home. Install. TBD Show How to Install. Download. UNIX openpts-0.2.4.tgz (Date: 2011-05-06, Size: 1.09 MB). Latest Release. openpts-0.2 openpts-0.2.6 (Date: 2012-01-05). openpts-0.2 openpts-0.2.5 (Date: 2011-07-20). openpts-0.2 openpts-0.2.4 (Date: 2011-05-06). openpts-0.2 openpts-0.2.3 (Date: 2011-03-10). GRUB-IMA 1.1.0.0 Fedora 10&12 (Date: 2009-06-01). Download File List.