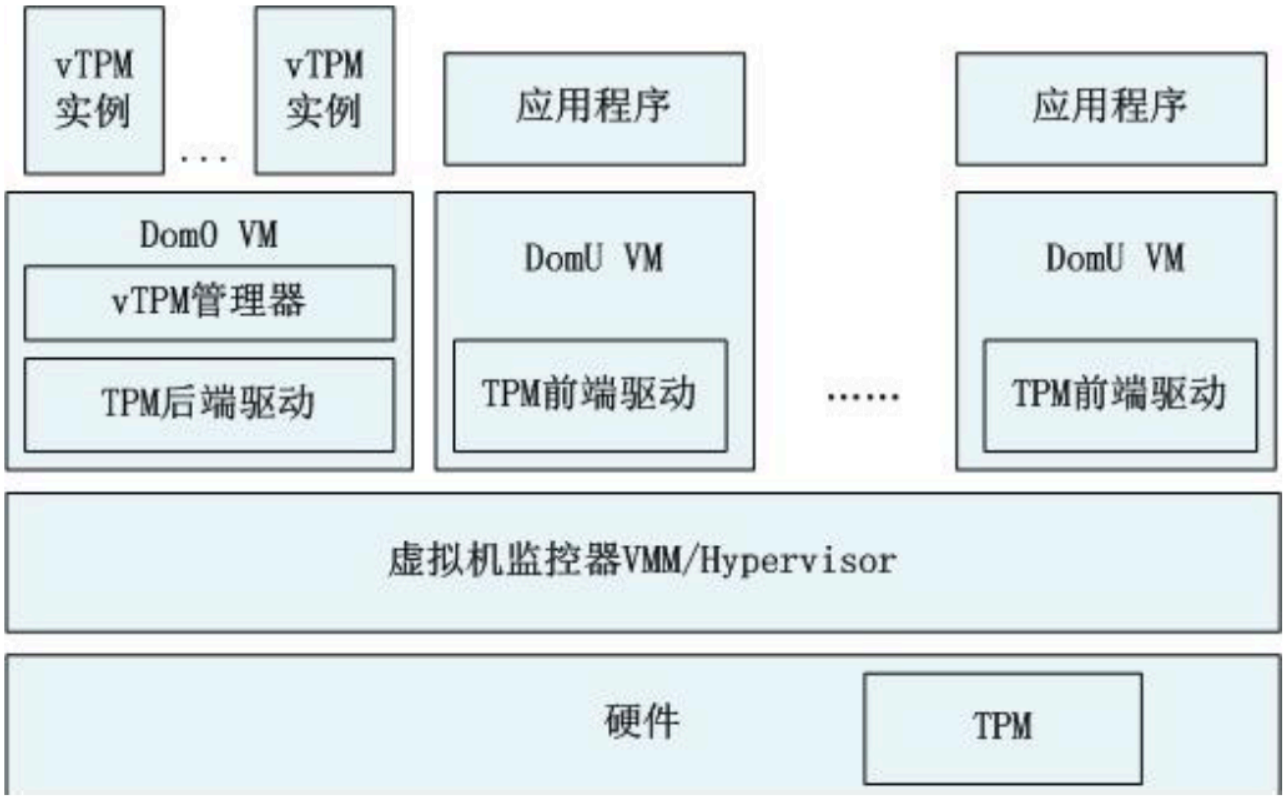
vTPM架构分析与环境部署

[TOC]

概述

可信平台模块（Trusted Platform Module，TPM）是可信计算的基石。可信计算是一种基于硬件的平台保护方案，能够记录平台（PC）从上电开始到bios、到grub、到操作系统及至应用程序的整个链式过程，并且通过密码学的机制使得这些记录能够完整地发送给远程端，由远程端来与预期值对比判断平台是否可信，这个过程称为远程证实（Remote Attestation）。

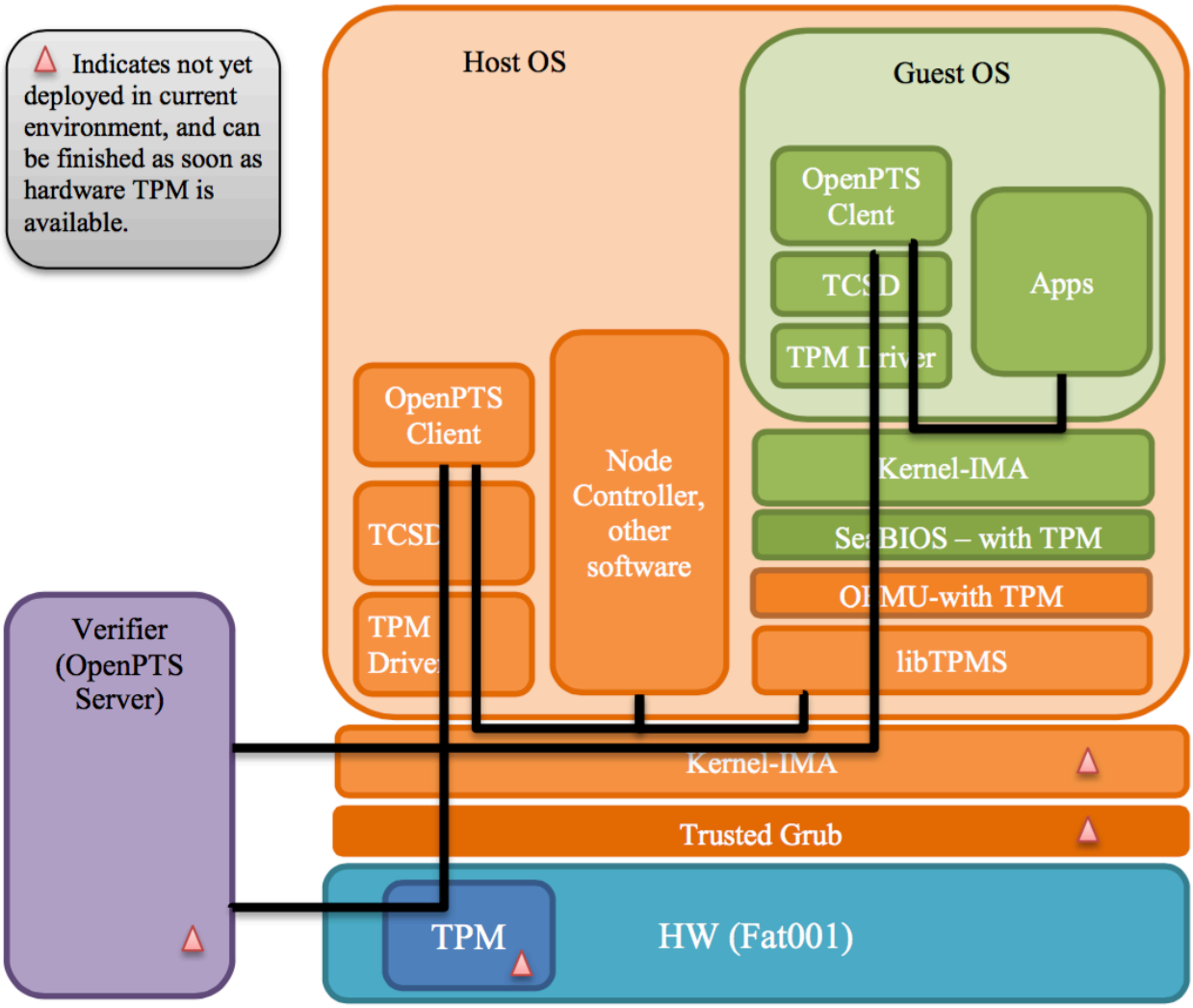
vTPM是对TPM的虚拟化，使得TPM能够应用在云计算等虚拟化的环境中。对于xen来说，物理TPM的驱动存在于Domain0中，同时利用vTPM管理器创建多个vTPM实例，这些实例与Domain U进行交互，使得虚拟机的可信服务成为可能。



xen虽然很好的支持了vTPM，但是在源码中我并没有找到完整性度量架构（Integrity Measurement Architecture，IMA）的实现代码，而且在使用vmware部署xen时发现，grub进入xen后会黑屏。因此，本文着重介绍vTPM在qemu-kvm中的架构及其部署过程。

qemu-kvm与vTPM

在Trusted Virtualization Platform Deployment（google学术可搜索到）中给出了使用qemu-kvm部署vTPM的架构图。



其中：

- kernel-IMA是在平台加载应用程序的时候，将应用程序的二进制值、加载的动态链接库与模块进行度量，度量值扩展写入PCR10，度量记录写入度量日志中；
- TCSD为TPM的软件栈；
- openPTS（Open Platform Trust Service）是远程证实的软件实现；
- libtpms为每个虚拟机提供了基于软件的TPM实现。

qemu-tpm从qemu中fork出来，以支持可信计算，其中包含一个后端驱动用来调用每个虚拟机的libtpms，以及将前端驱动暴露给每个虚拟机。

qemu-kvm vTPM环境部署

按照上述架构，qemu-kvm vTPM环境部署包括安装libtpms、qemu-tpm等。同时，

- 由于我对OpenAttestation更加熟悉一些，所以暂且先不考虑部署openPTS。
- 在安装中发现ubuntu14.04的nss包中找不到blapi.h, ubuntu15.04中没有AES_CreateContext函数（有可能是我安装的版本不对），因此最终部署环境选择为centos7。
- 我的计算机没有TPM物理芯片，因此在host os上还得安装tpm的软件实现swtpm。
- 最终需要安装的包为seabios-tpm、swtpm、qemu-tpm以及libtpms。

安装之前

安装依赖包：
```
yum install glibc-headers openssl-devel nss-softokn-freebl-devel nss-softokn-devel gmp-devel libtool nss-devel

yum install automake autoconf bash coreutils expect libtool sed fuse fuse-devel glib2 glib2-devel gmp gmp-devel nss-devel net-tools selinux-policy-devel gnutls gnutls-devel libtasn1 libtasn1-tools libtasn1-devel rpm-build iasl socat

yum groupinstall "Development Tools"

yum install pixman pixman-devel libuuid-devel libaio-devel spice-server-devel SDL SDL-devel
```

下载安装包：

1. libtpms: https://github.com/stefanberger/libtpms
2. swtpm: https://github.com/stefanberger/swtpm
3. seabios-tpm: https://github.com/stefanberger/seabios-tpm
4. qemu-tpm: https://github.com/stefanberger/qemu-tpm

安装seabios-tpm与libtpms

seabios：直接make即可，记住out/bios.bin路径，最好写入环境变量。

```
make
```

libtpms:

```
# ./bootstrap.sh
# ./configure --prefix=/usr
# make
# sudo make install
```

swtpm安装

```
./bootstrap.sh
./configure --prefix=/usr
make
make check
sudo make install
```

出现错误：

```
configure: error: "Is libtpms-devel installed? -- could not get libs for libtpms"

[luowu@localhost swtpm]$ pkg-config --libs libtpms
Package libtpms was not found in the pkg-config search path.
Perhaps you should add the directory containing `libtpms.pc'
to the PKG_CONFIG_PATH environment variable
No package 'libtpms' found

[luowu@localhost swtpm]$ sudo find / -name libtpms.pc
[sudo] password for luowu:
find: ‘/run/user/1000/gvfs’: Permission denied
/usr/lib/pkgconfig/libtpms.pc

[luowu@localhost swtpm]$ sudo vim /etc/profile
export PKG_CONFIG_PATH=/usr/lib/pkgconfig:$PKG_CONFIG_PATH

[luowu@localhost swtpm-master]$ source vim /etc/profile
```

安装qemu-tpm

```
./configure --enable-kvm --enable-tpm --enable-sdl
make
sudo make install
```

出现错误：

```
ERROR: DTC (libfdt) version >= 1.4.0 not present. Your options:
         (1) Preferred: Install the DTC (libfdt) devel package
         (2) Fetch the DTC submodule, using:
             git submodule update --init dtc
```

解决方案：

1. 下载dtc-1760e7c.tar.gz
2. 解压后执行make
3. 将所有文件复制到qemu-tpm/dtc下

启动vTPM

创建/dev/vtpm*:

```
sudo modprobe cuse
mkdir /tmp/myvtpm0
chown -R tss:root  /tmp/myvtpm0
swtpm_setup --tpm-state /tmp/myvtpm0  --createek
```

出现错误：

```
Error: Cannot access config file /etc/swtpm_setup.conf.

[luowu@localhost swtpm]$ sudo find / -name swtpm_setup.conf
find: ‘/run/user/1000/gvfs’: Permission denied
/usr/etc/swtpm_setup.conf

[luowu@localhost swtpm]$ sudo cp /usr/etc/swtpm_setup.conf /etc/swtpm_setup.conf
```

成功界面为：

```
[root@localhost swtpm]# swtpm_setup --tpm-state /tmp/myvtpm0  --createek
Starting vTPM manufacturing as tss:tss @ Fri 22 Jan 2016 01:39:43 PM CST
TPM is listening on TCP port 44121.
Ending vTPM manufacturing @ Fri 22 Jan 2016 01:39:44 PM CST
```

再执行下述命令，能够看到文件/dev/vtpm0。
`export TPM_PATH=/tmp/myvtpm0 swtpm_cuse -n vtpm0` 创建虚拟机～：

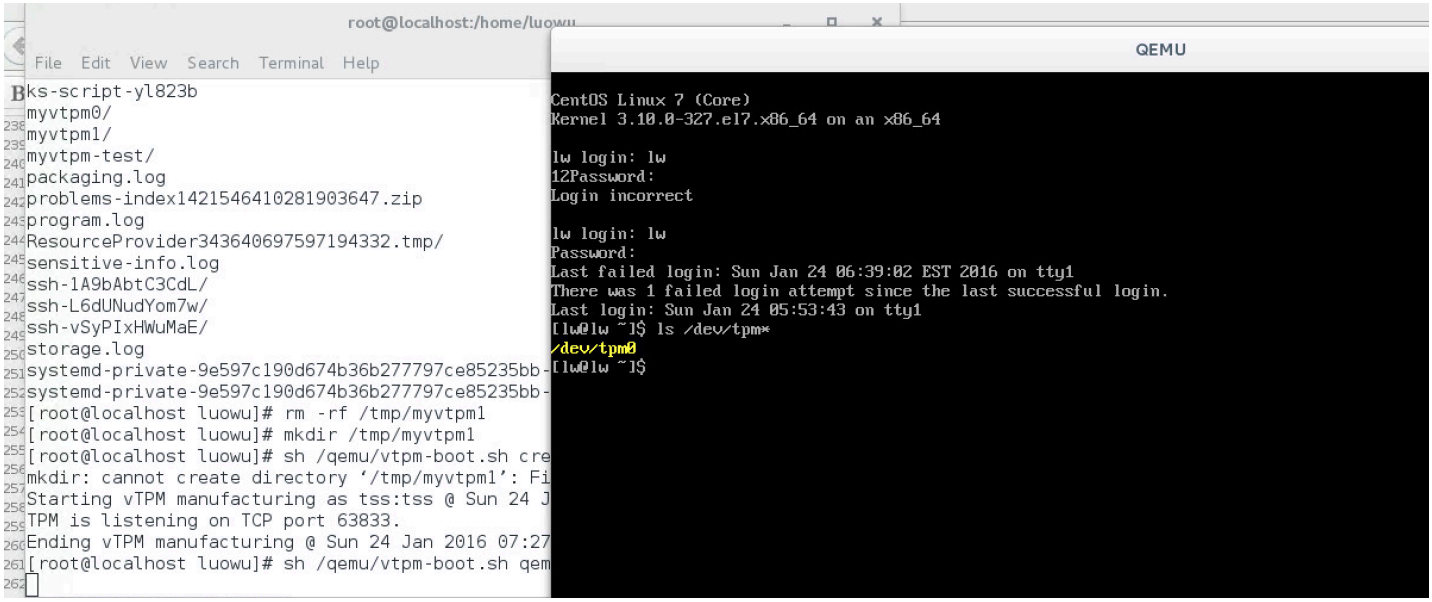
```
qemu-img create -f qcow2 <YOUR IMG PATH> 30G

qemu-system-x86_64 -display sdl -enable-kvm -cdrom <YOUR ISO PATH> \
    -m 1024 -boot d -bios $SEABIOS/bios.bin -boot menu=on -tpmdev \
    cuse-tpm,id=tpm0,path=/dev/vtpm0 \
    -device tpm-tis,tpmdev=tpm0 <YOUR IMG PATH>
```

安装虚拟机就和普通安装系统一样，这里不再介绍（我的iso文件是centos7）。

安装成功后执行（若出现错误，重新执行生成/dev/vtpm0的命令）：

```
qemu-system-x86_64 -display sdl -enable-kvm  \
    -m 1024 -boot c -bios $SEABIOS/bios.bin -boot menu=on -tpmdev \
    cuse-tpm,id=tpm0,path=/dev/vtpm0 \
    -device tpm-tis,tpmdev=tpm0 <YOUR IMG PATH>
```



至此，qemu虚拟机里已经能够看到/dev/tpm0了，可以愉快地进行下一步工作了～