全国大学生信息安全创新能力赛初赛赛题

——平台可信认证

## 一 题目描述：

某单位为自己的文件服务器提供了远程登录功能，员工可以通过远程登录过程访问服务器。因工作需要，服务器上存放了敏感文件，因此，公司要求员工的远程访问操作必须在一台装有TCM1.0模块，具备可信启动功能，安装了安全操作系统，且安全策略符合公司规定的可信终端上进行。单位的信息安全官小Z决定在登录过程中，添加平台可信验证功能，通过在终端上获取可信启动机制、操作系统内核、系统安全机制、系统安全初始策略的度量值并生成可信报告，在登录过程中，由服务器验证可信报告来实现一要求。

为此，小Z首先研究了可信终端自身的安全机制，发现可信终端在启动过程中，将可信启动机制度量结果写入了PCR7寄存器，将操作系统内核与系统安全机制的度量结果写入了PCR11寄存器，将系统安全策略的度量结果写入了PCR14寄存器。小Z在终端上部署了可信报告模块，使用TCM的quote机制对PCR7、PCR11和PCR14三个寄存器合并生成可信报告，并准备在服务端使用预先存储的可信终端鉴别密钥pik.key验证可信报告的可信性，再通过验证pcr寄存器与对可信终端预期的一致性来确认终端是否为可信终端。

在服务端开发时，小Z构造了程序框架，然后发现自己的开发基础知识不够，难以完成可信报告的验证工作。你能否在小Z工作的基础上，完成可信报告验证与PCR验证部分代码，实现对终端的可信验证

请在platform\_verify/src/trust\_verify目录下，找到可信验证的代码位置，编写代码，完成可信报告验证与pcr值验证的功能，使得系统可以验证合法Quote报告，并验证PCR值是否与对可信终端的预期一致。两项验证均正确即可获得flag。

**二 题目环境：**

本题目为“兵棋“模式新型可信计算赛题，示例在docker环境下运行，基于可信软件基原型Cube-1.3、可信tcm模拟器cube-tcm和主动免疫安全可信组件集合cube\_tcmplugin搭建，通过分布式消息驱动机制模拟应用流程与攻防行为。

可信报告过程分两阶段，第一阶段由用户发出报告申请，申请经终端发送给server，server返回防重放攻击的随机数，终端收到随机数后完成第一阶段。第二阶段则由终端获取PCR7,PCR11和PCR14的值，使用随机数生成可信报告并发送给server，server验证报告签名以及报告中PCR取值，并返回验证结果信息。下面是两个阶段的示意图：

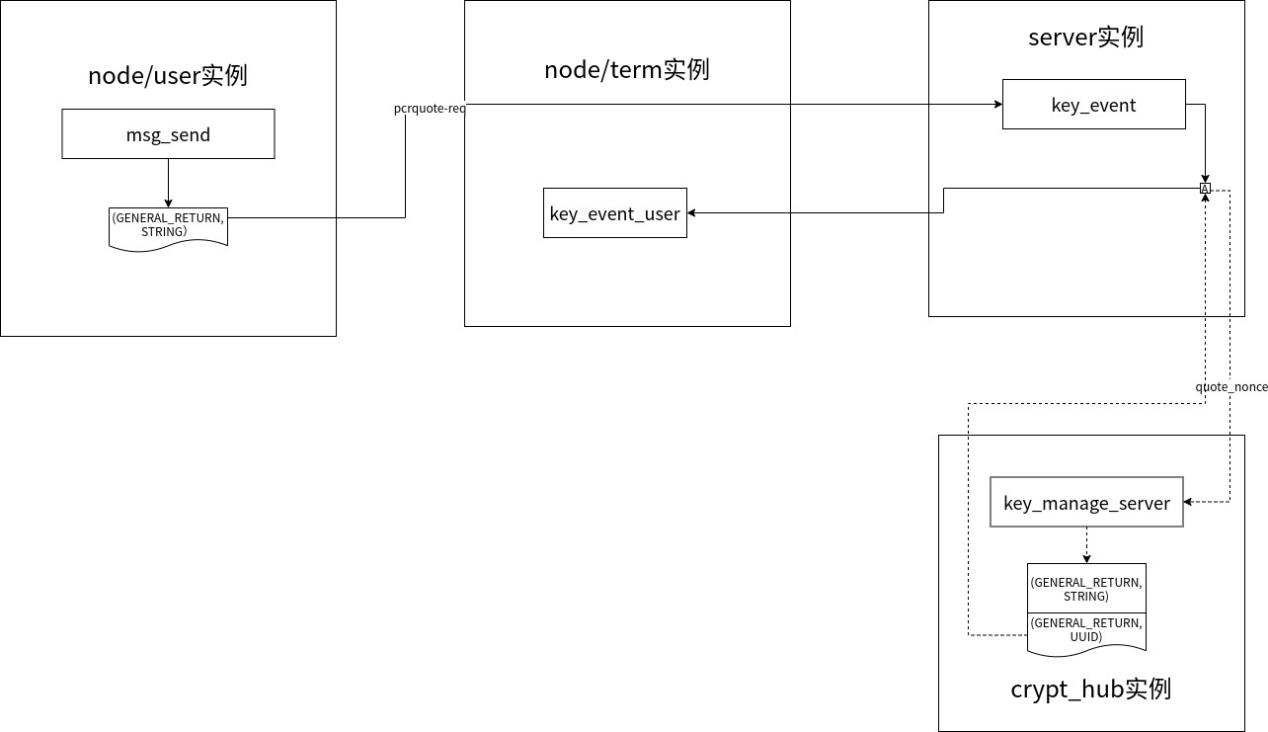


图1 可信报告过程（第一阶段）

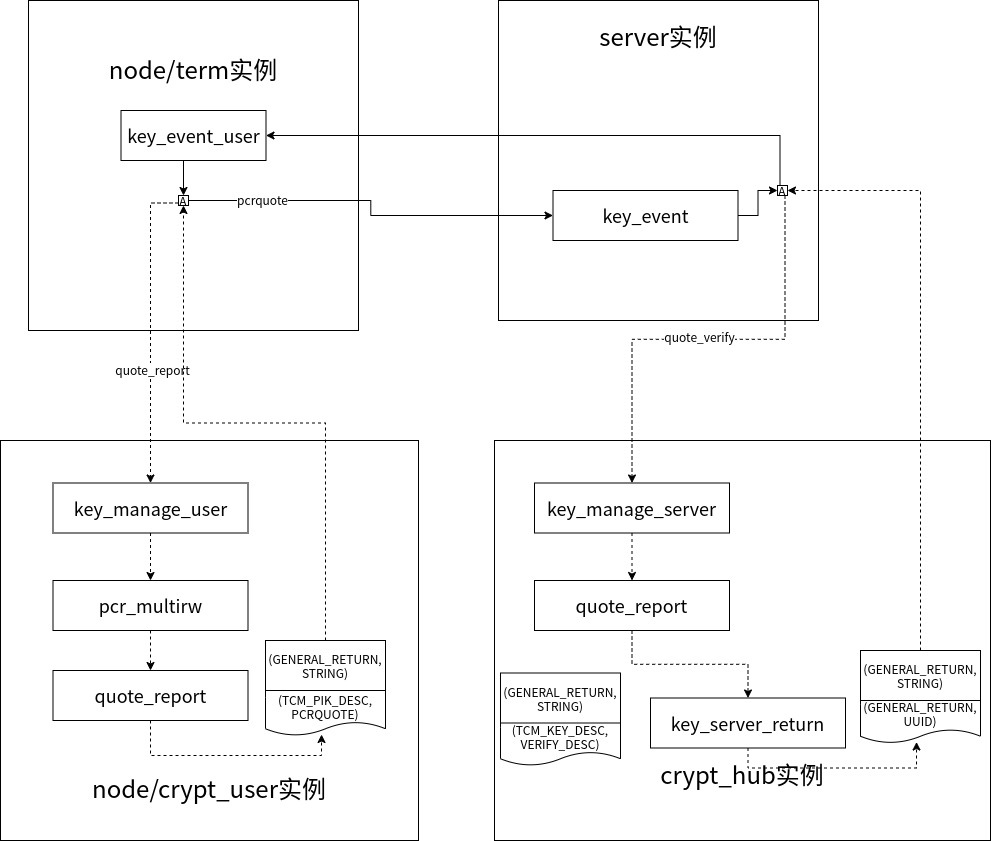


图2 可信报告过程（第二阶段）

第一阶段用户发起可信报告请求后，通过term，server到crypt\_hub模块获取随机数，以抵御重放攻击威胁。第二阶段则在终端term和服务器server间进行，终端通过pcr\_multirw模块，获取了7,11和14号PCR寄存器的值，并交由qoute\_report模块，作为报告中描述可信属性的部分，在server端，则由key\_manage\_server将报告存到磁盘数据库中，quote\_report完成报告验证，输出格式为（TCM\_KEY\_DESC,VERIFY\_DESC）的验证结果，再由key\_server\_return模块验证PCR取值与预期的一致性，完成验证后，在报告消息上添加格式为（GENERAL\_RETURN,UUID）的扩展项作为验证结论输出。

（TCM\_KEY\_DESC,VERIFY\_DESC）数据格式如下：

表1 quote\_report输出结果格式

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 记录项名称 | 英文名 | 数据格式 | 长度 | 内容 | 备注 |
| 验证结果 | result | INT | 4 | 表示完整报告验证结果，0为验证成功，1为失败 |  |
| 对象编号 | object\_uuid | UUID | 32 | 验证证书的编号 |  |
| 密钥编号 | verifykey\_uuid | UUID | 32 | 验证密钥编号 | 暂不使用 |

（GENERAL\_RETURN,UUID）数据格式如下：

表1 key\_server\_return模块验证结果数据格式

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 记录项名称 | 英文名 | 数据格式 | 长度 | 内容 | 备注 |
| 名称 | name | ESTRING | <=32 | 用来提供相关的信息 | 这里用来表示验证结果 |
| 取值 | return\_value | UUID | 32 | 报告的摘要值，与object\_uuid一致 |  |

name的取值表如下所示：

|  |  |
| --- | --- |
| 验证情况 | name取值 |
| 签名验证未通过 | “sign verify fail” |
| PCR值不符合预期 | “trust verify fail” |
| 两项验证均通过 | “trust verify succeed” |

pcr基准值存储在内存数据库（GENERAL\_RETURN,UUID）中，name项用来描述不同度量结果，如下所示：

|  |  |  |
| --- | --- | --- |
| 度量内容 | name取值 | 预期写入寄存器 |
| 可信启动机制 | “trust\_boot” | 7 |
| Linux内核 | “linux\_kernel” | 11 |
| 系统安全机制 | “secure\_mechanism” | 11 |
| 安全策略 | “secure\_policy” | 14 |

请在key\_server\_return模块下编写verify\_output函数，按上述需求，输出验证结果。

**三 解题要求**

可以直接使用vi 编写src/key\_server\_return/key\_server\_return.c中的代码，代码相关位置已有提示。

编码完成后，可执行make进行编译，编译成功后回到platform\_verify目录，执行./tcm\_quote.sh进行单次可信报告测试，可通过观察调试信息或直接观察instance/crypt\_hub的message.log结尾部分来检查输出结果,以确认本地测试结果正确。

系统本地测试通过后，可执行sh player.sh启动评分过程，评分过程会转入root环境进行测试，并提示进行测试的内容。完成测试后，如root环境输出正确，则评分过程会向用户发送flag，用户获取后提交flag完成解题。

**四 题目提示**

代码相关位置已提供了需要实现相关功能的提示，请参考编写。另外编写过程中可插入多条printf语句来进行跟踪调试，建议printf时添加一些特殊标记来区分它与其它调试信息。

题目中涉及大量TCM数据结构和自定义数据结构，除TCM\_PCR\_COMPOSITE结构外，不建议通过查询头文件来了解其结构，建议测试运行后，观察instance/crypt\_hub/message.log，既可以追踪整个过程，也可以看到相关数据结构的内容。

TCM\_PCR\_COMPOSITE结构如下：

typedef struct tdTCM\_PCR\_COMPOSITE {

TCM\_PCR\_SELECTION select; //pcr寄存器的选择，用位图表示，

uint32\_t valueSize; //pcr寄存器数值的总长度，

//为寄存器数目\*32

BYTE \*pcrValue; //pcr寄存器内容

}\_\_attribute\_\_((packed)) TCM\_PCR\_COMPOSITE;

结构中的子结构TCM\_PCR\_SELECTION定义如下：

typedef struct tdTCM\_PCR\_SELECTION {

uint16\_t sizeOfSelect; //本题中未使用

BYTE pcrSelect[TCM\_NUM\_PCR/CHAR\_BIT];

//寄存器选择的位图，长度为3字节

}\_\_attribute\_\_((packed))TCM\_PCR\_SELECTION;

**五 备注**

因为平台权限限制问题，我们在crypt\_user报告生成部分并没有执行可信报告流程，而是改用实际过程中采集到的可信报告取值进行了模拟，这个不影响crypt\_hub这边的解题过程。今后我们将提供可支持完整可信功能的实训平台，让同学们能够获得更真实的体验。