系统安全

可信计算

黄杰 网络空间安全学院

本章学习内容

知识点

1、可信计算的基本概念

2、可信计算的基本功能

3、3种可信计算技术

4、信任链

9.0

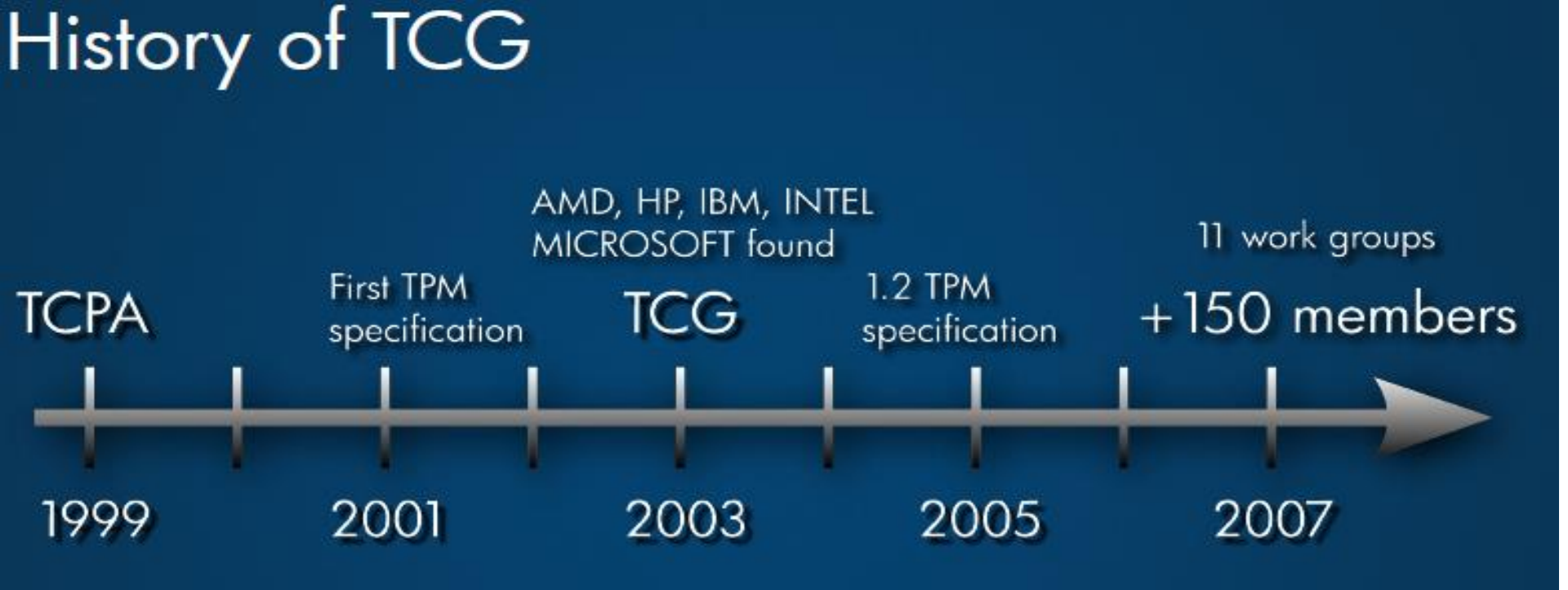
绪　论

9.0.1 可信计算的理念

- **可信计算解决的问题**
  - 解决人与程序之间、人与机器之间的信息安全传递。因此，“可信计算”成为信息安全发展的必由之路
  - 有别于传统的信息安全技术，可信计算的目标希望杜绝的是不可信代码的存在，包括有漏洞的，或者是恶意的
  - 对于微机，只有从芯片、主板、BIOS（Basic Input Output System，基本输入输出系统）和操作系统做起，采取综合措施，才能提高微机的安全性。正是这一思想推动了可信计算的产生和发展
- **可信计算的理念**
  - 从硬件到软件；从基础软件到应用系统；从PC到服务器；从移动设备到网络；从存储到外设。**无所不包**
  - 从进入可信计算环境开始，直到退出，**提供一个完整的解决方案**
  - 以标准的形式，**提供一个可伸缩的、模块化的体系架构**

9.0.2 可信计算在中国

- **紧跟美国的步伐**
  - 上世纪九十年代——PC机安全防护系统
  - 2004 年具有自主知识产权的可信计算机产品面世（瑞达）
  - 2005年联想“恒智”芯片和北京兆日公司TPM芯片
- **2002年中国信息产业商会信息安全产业分会提出了可信网络世界体系结构框架(Trusted Cyber Architecture Framework，TCAF)**
- **2005年1月，成立国家安全标准委员会WG1可信计算工作小组专门规划可信计算相关标准**
  - 2006年颁布了《可信计算平台密码技术方案》和《可信计算密码支撑平台功能与接口规范》
- **2007年，沈昌祥院士主导了“可信计算平台密码规范”、“可信计算基础支撑软件”、“可信平台主机规范”、“可信网络连接规范”等草案的制定**
- **2008年4月底，中国可信计算联盟 (CTCU)在国家信息中心成立**



- 1999年IEEE太平洋沿岸国家容错系统会议改名为“可信计算会议”
- 2000年12月11日，成立了可信计算联盟TCPA，2003年改为TCG
- TCPA于2001年9月制定了可信PC的详细实现规范V1.1
- 2003年9月TCG推出可信PC的新规范V1.2
- 2013年TCG发布TPM 2.0标准库，可信计算进入2.0时代。

9.1 可信计算概述

9.1.1 可信计算的概念

- **可信的概念**
  - 可信是指值得信任，一个系统可信是指系统的运行（或输入输出关系）符合预期的结果，没有出现未预期的结果或故障。
  - TCG的定义：如果一个实体的行为总是以预期的方式达到既定目标，那么它是可信的。
  - ISO / IEC15408的定义：一个组件、操作或过程的可信是指在任意操作条件下是可预测的，并能很好地抵抗应用程序软件、病毒以及一定物理干扰所造成的破坏。
- **信任的属性**
  - **信任具有二重性**，既具有主观性又具有客观性。
  - **信任不一定具有对称性**，即A信任B不一定就有B信任A。
  - **信任可度量**，也就是说信任的程度可以测量，可以划分等级。
  - **信任可传递**，但不绝对，而且在传播过程中有损失。
  - **信任具有动态性**，即信任与环境(上下文)和时间等因素相关。

9.1.1 可信计算的概念

- 针对信息系统，实现“可信计算”需要达到以下要求：
  - 验证用户的身份：验证使用者的合法身份；
  - 验证平台软硬件配置的正确性：使用者可以信任平台的运行环境，软硬件配置没有问题；
  - 验证应用程序的完整性和合法性：在平台上运行的应用程序是可信的，是正版软件且未受破坏；
  - 平台之间的可验证性：在网络环境下运行的多个平台之间是可以相互信任的，即这些平台本身各自可信，且可以合法地相互访问和相互通信不存在安全问题。

9.1.1 可信计算的概念

TCG的可信计算概念

- TCG联盟制定了可信计算规范，提出了基于可信计算平台模块（TPM）的可信计算平台（TCP）体系结构。
- 主要定义了可信计算的三个安全属性：
  - 可鉴别性（Authentication）：信息系统的用户可以认证与他们进行通信的对象身份。
  - 完整性（Integrity）：用户能够确保信息在传输和保存过程中不会被窜改或伪造。
  - 机密性（Privacy）：用户相信系统能保证其信息的私密性不被泄漏。

9.1.1 可信计算的概念

微软的可信计算概念

- 在2002年，微软发布的“可信计算”白皮书中，从实施（Execution）、方法（Means）、目标（Goals）三个角度对可信计算进行了概要性的阐释。其目标包括四个方面：
  - 安全性（Security）：用户希望系统受到攻击后具有恢复能力，而且能够保护系统及其数据的机密性、完整性和可用性；
  - 机密性（Privacy）：用户能够控制与自己相关的数据不会泄密，并按照信息平等原则使用数据；
  - 可靠性（Reliability）：用户可在任何需要服务的时刻获得服务；
  - 完整性（Integrity）：强调服务提供者以快速响应的方式提供负责任的服务，且服务在传输和保存过程中不会被窜改或伪造。

9.1.1 可信计算的概念

Intel的可信计算概念

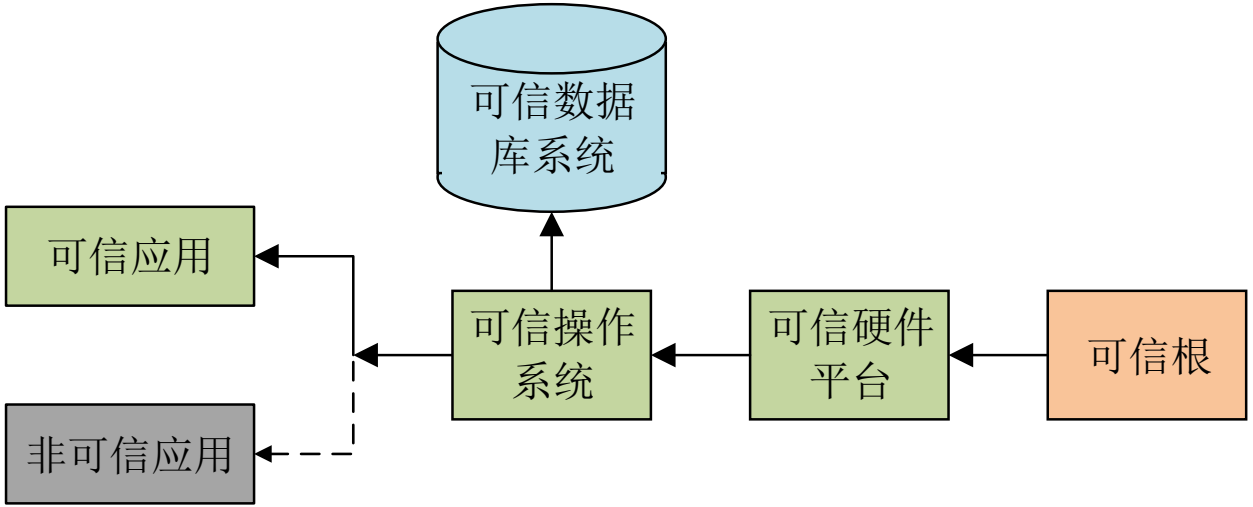
- **LT技术：**和TCG推出的PC实施规范有所区别，它用到了TCG定义的TPM，并基于此来构建自己的安全架构，但LT技术扩展了TCG所定义的可信计算的功能和范围，它能够保护处理器、芯片组和系统平台（包括内存、键盘、显示部件等）以抵御黑客软件对系统的恶意攻击。在这种采用了硬件级安全保护的环境中，用户的私密信息将得到很好的保护。

- **TXT技术：**使用硬件密钥和子系统来控制系统内部的资源，并决定谁或什么进程能够访问这些资源。

9.1.1 可信计算的概念

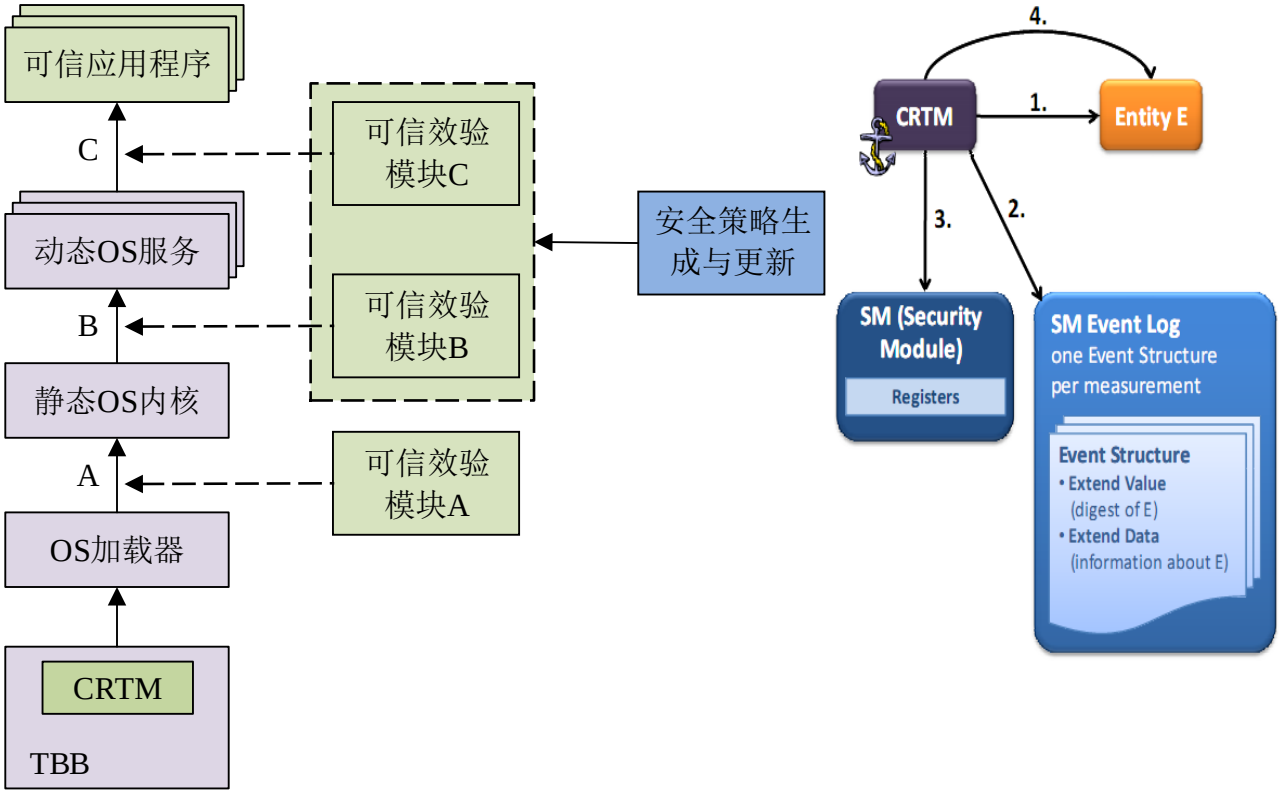
可信计算机系统的结构

- 可信根是系统的安全基础也是安全起点，在可信网络环境中所有安全设备都信任该可信根。可信应用将会从下层获得安全支撑，而非可信应用可以运行于可信系统之上，但不能获得安全支撑。



9.1.1 可信计算的概念

可信计算平台的启动流程



1. 可信度量根（RTM）测量实体E
2. RTM在TPM事件日志中，生成事件结构。SML包含 Event Structures，SM Event Log能存储在任意存储设备如磁盘上
3. RTM将值扩展到寄存器中
4. 执行/传递控制给实体E

9.1.2 可信计算的基本功能

完整性度量、存储和报告

- 完整性度量是一个过程，是在可信平台启动过程中，组件（固件或软件）加载和执行之前，其度量散列值被“扩展”到了TPM内部的平台配置寄存器（PCR）中，通过计算该组件的散列值同期望值的比较，就可以维护它们的完整性。
- 一次度量就是一个度量事件，每个度量事件由两类数据组成：
  - 被度量的值：嵌入式数据或程序代码的特征值。
  - 度量散列值：被度量值的散列值。
- 完整性报告则是用于验证平台的当前配置，证明完整性存储的过程，展示保护区域内存储的完整性度量值，依靠可信平台的鉴定能力判断存储值的正确性。

9.1.2 可信计算的基本功能

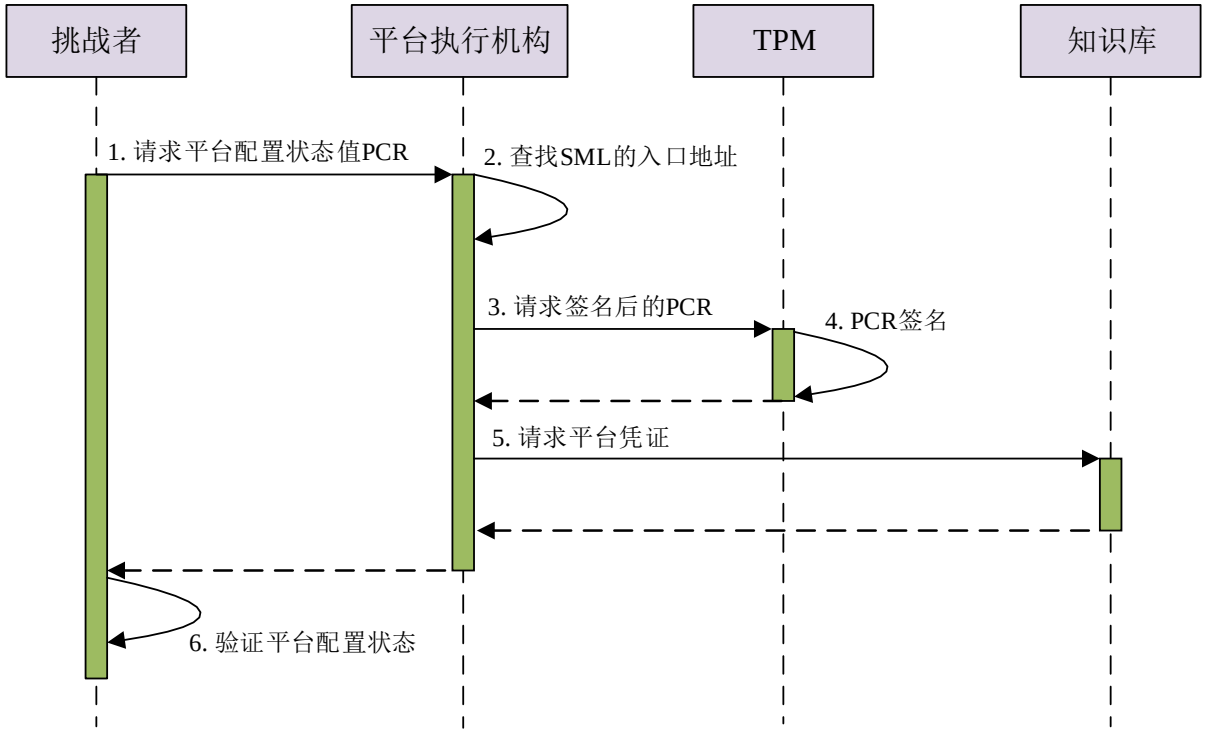
完整性度量、存储和报告

度量日志（SML）作用：

- 各阶段代码的详细配置信息和对PCR值扩展操作的历史记录是保存在度量日志中
- 度量日志存储在磁盘上，而磁盘属于不可信的外存，因此度量日志是可能被攻击者篡改的
- 攻击者即使篡改了度量日志，但由于PCR中记录的度量值是不可伪造的，用户对度量日志进行摘要就会发现与度量值不匹配

问题：SML需要额外的保护吗？

一个度量事件的完整性验证过程：



9.1.2 可信计算的基本功能

平台证明

- 外部实体能够确认被保护区域、受保护能力和信任根，而本地调用不需要证明。
- 通过证明，完成了远程实体对平台身份的认证。
- 由于引入了AIK对PCR值和随机数N在TPM的控制下的签名，保证了平台配置信息的完整性和新鲜性，从而大大提高了通信的安全性。

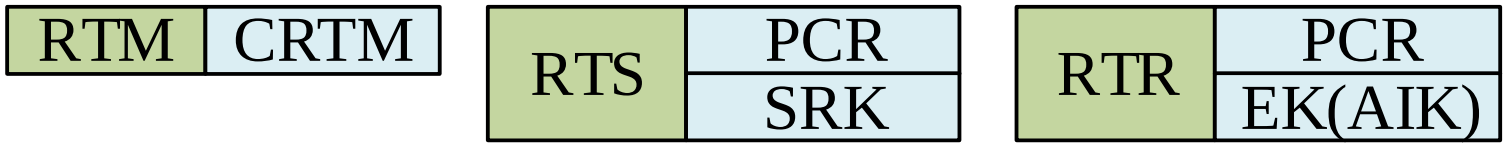
受保护能力

- 受保护能力就是唯一被许可具有访问被保护区域的特权命令集，而被保护区域就是能够安全操作敏感数据的地方，如：内存、寄存器等。
- TPM通过实现受保护能力和被保护区域，来保护和报告完整性度量值。

9.2 可信计算技术

9.2.1 可信平台的信任根

- TCG认为一个可信平台必须包含三个可信根：
  - 可信度量根（ Root of Trust for Measurement，简称RTM）
  - 可信存储根（Root of Trust for Storage，简称RTS）
  - 可信报告根（Root of Trust for Report，简称RTR）



9.2.1 可信平台的信任根

- RTM是平台启动时首先执行的一段程序，它是由CRTM控制的计算引擎。在理想状态下，CRTM存储在TPM内部，但根据实现的需要，它可能需要加载到其他固件中，如：BIOS。

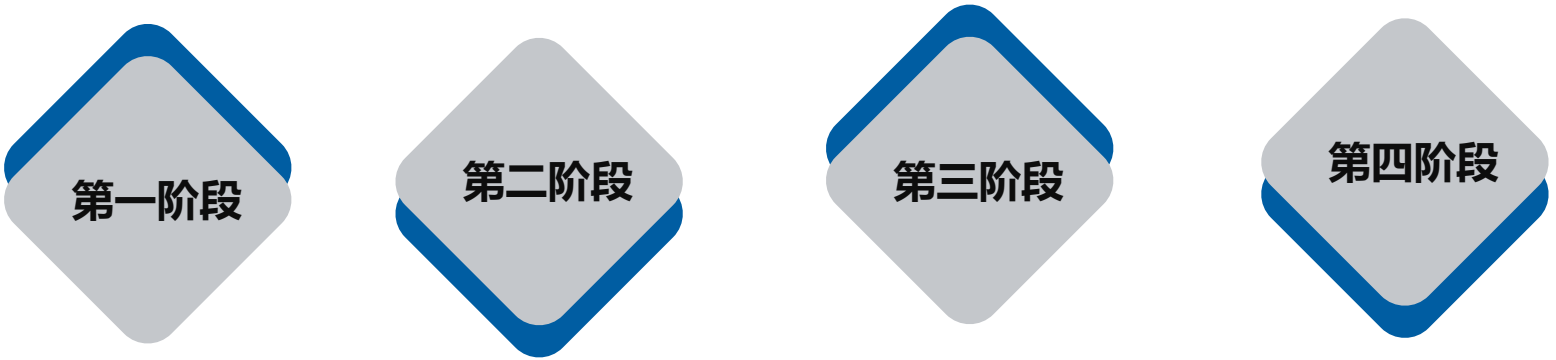
- RTS由TPM芯片中的PCR和存储根密钥（Storage Root Key，简称SRK）组成。处于密钥树根部的密钥是最高级存储密钥，即存储根密钥SRK，它是2048位的RSA密钥对，主要用于对由TPM使用，但存储在TPM之外（如：硬盘）的密钥进行保护。同时，它作为父密钥对其子密钥进行加密保护。

- RTR由TPM芯片中的PCR和背书密钥（Endorsement Key，EK）组成。EK仅用于以下两种操作：一是创建TPM的拥有者；二是创建AIK及其授权数据。

9.2.2 TPM

- TPM是平台可信的起点，可信信息系统以它为信任根构建可信的计算环境。

- 由底层固件依次验证BIOS和操作系统的完整性，如正确则正常运行操作系统，否则运行停止。

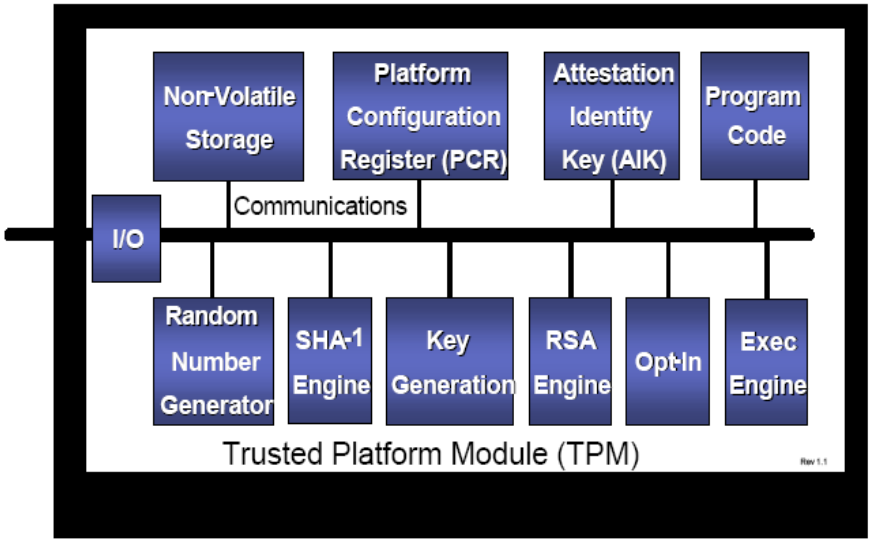


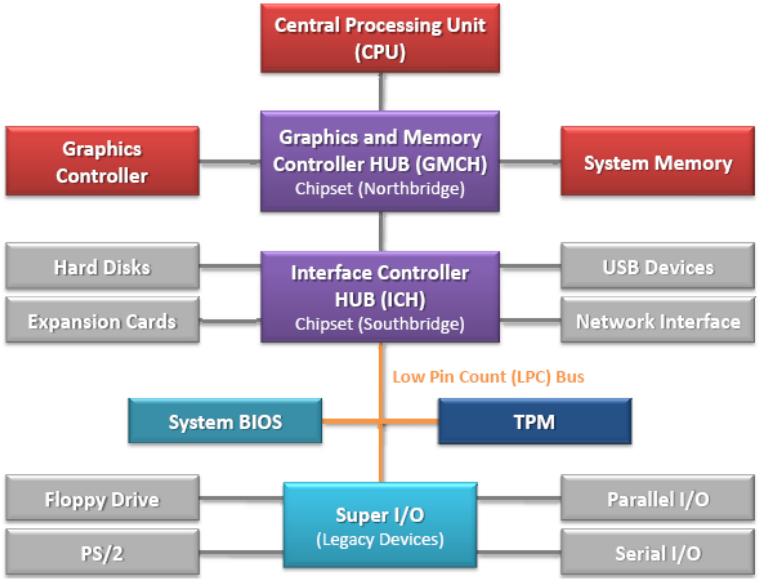
- 平台上电后，首先TPM芯片验证当前底层固件的完整性，如正确则完成正常的系统初始化。

- 利用TPM内置的加密模块生成系统所需要的各种密钥。

9.2.2 TPM

组成结构



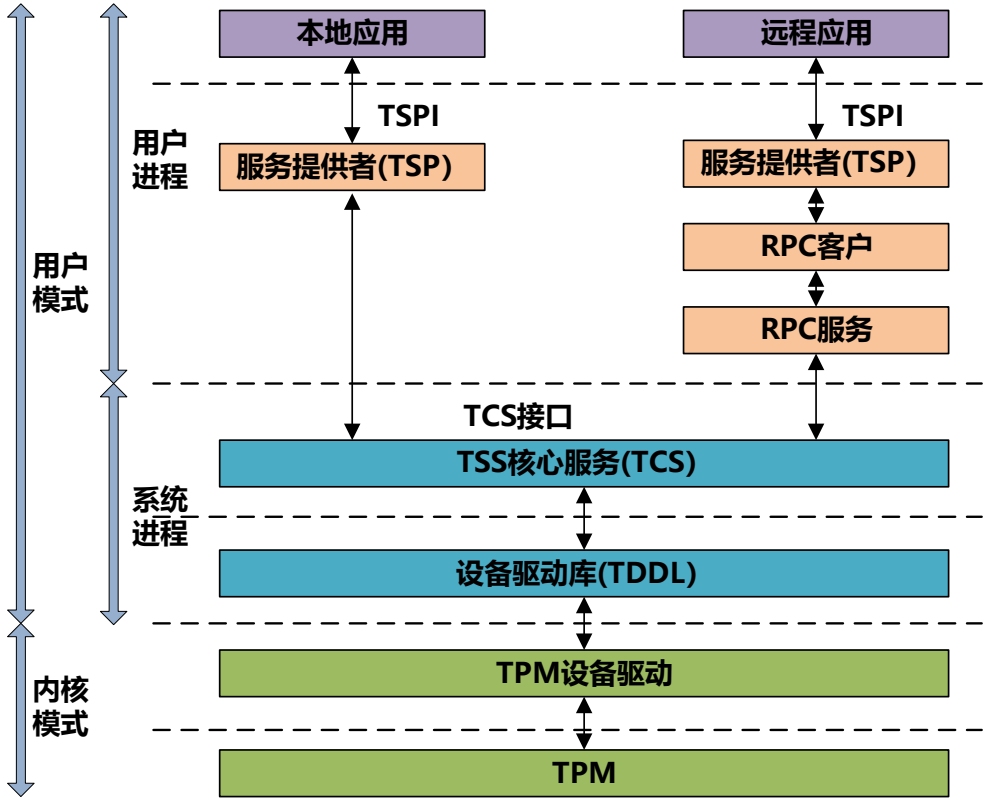


TPM至少需要具备四个主要功能：对称/非对称加密、安全存储、完整性度量和签名认证。

9.2.3 可信计算平台

体系架构

- 内核模式
  - 运行TPM设备驱动和TPM的核心组件
  - 只有在管理员授权下才能修改其中运行的代码
- 用户模式
  - 根据用户的要求来加载和执行应用程序和服务
  - 用户模式有两类进程：**系统进程**和**用户进程**



9.2.3 可信计算平台

TPM和TCS的比较

TPM的缺陷	TCS的优点
1、一次只有一个操作可以进行 2、处理速度很慢 3、有限的资源，包括密钥槽、授权槽等 4、只能通过一个驱动程序与其进行串行通信 5、本地软件与之通信是有限制的	1、可以对多个TPM待处理的操作进行排队 2、对于不需要TPM处理的操作，TCS可以自行作出响应 3、对TPM有限资源进行管理，可看作是无限的资源 4、将输入输出的数据进行相应的转换 5、可以对资源提供本地的或者远程的调用方式

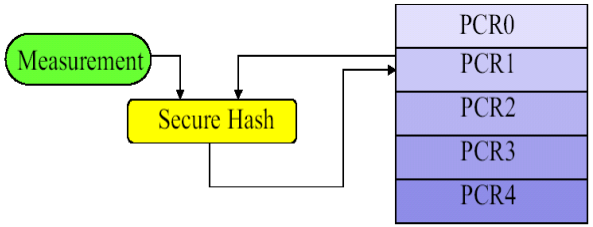
9.3 信任链技术

9.3.1 信任链

- 问题：信息系统在执行各种任务时，其控制权在不同的实体之间传递，很难保证某个实体不会因为恶意的原因，取得系统的控制权后，对系统进行破坏。
- 解决问题的方法：可信度量、存储和报告机制
  - 在信息系统顺序启动时，TCG对启动的所有固件或软件进行可信性度量，然后将度量的值进行安全存储，一旦访问者需要即可提交报告。
  - 由于PCR的存储空间有限，无法单独存储启动过程中所有的度量值，因此，采用了一种“扩展”操作。

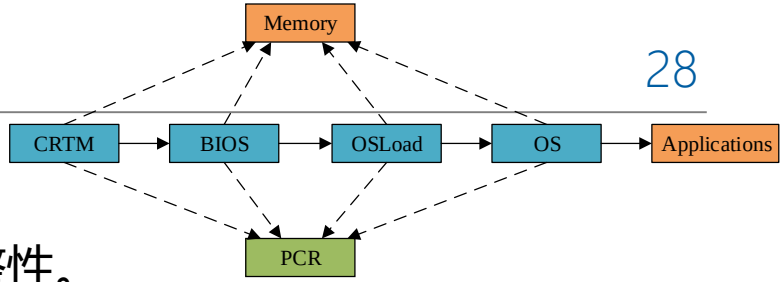
    PCR[n] = SHA-1{PCR[n-1] || newMeasurement}
  - “扩展”操作需要记录平台的启动顺序和在启动过程中的可信度量结果
- 信任链

CRTM→BIOS→OSLoader→OS→Applicatons



$PCR_n = Hash(PCR_{n-1} || Measurement_n)$

9.3.1 信任链



信任链的度量过程

- 当系统加电以后，CRTM度量BIOS的完整性。
- 如果BIOS可信，则可信的边界将从CRTM扩展到CRTM+BIOS。于是执行BIOS。
- BIOS度量OSLoader。
- 如果OSLoader可信，则可信的边界扩展到CRTM+BIOS+OSLoader，执行操作系统的加载程序。
- OSLoader在加载操作系统之前，首先度量操作系统的完整性。
- 如果操作系统可信，则可信边界扩展到CRTM+BIOS+OSLoader+OS，加载并执行操作系统。
- 操作系统启动后，由操作系统度量应用程序的完整性。
- 如果应用程序可信，则可信边界扩展到CRTM+BIOS+OSLoader+OS+Applications，操作系统将加载并执行应用程序。

9.3.1 信任链

PCR的安全性

- TPM只允许两种操作来修改PCR的值：重置操作和扩展操作。
- **重置操作**发生在机器断电或者重新启动之后，PCR的值自动重新清零（但TCG 1.2新引入的寄存器除外）。
- 只能通过**扩展操作**来改变PCR的内容。扩展操作是不可交换的，即先扩展度量值A再扩展度量值B所得到的PCR值与先扩展B再扩展A的结果是不同的。
- 理论上PCR能够记录一个无限长的度量值序列，这个度量值序列反映了系统状态的变迁。
- 如果扩展序列中的某个度量值被改变了，那么后续的度量序列都会受到影响。

9.3.2 动态可信度量

- 静态度量：没有度量运行过程中加载的软件，因此无法保证系统运行时的安全。
- 动态度量：TPM1.2及其之后的版本，增加了PCR[16]~PCR[23]用于支持动态可信度量技术。
- 动态可信根(DRTM)： 使TPM可以在任何时候执行度量，重新构建平台的信任链，而不需要重启整个平台。

PCR	初始值	PCRReset T/OSPresent=FALSE	TPM_HASH_START	PCRReset T/OSPresent=TRUE
0～15	0	NC	NC	NC
16	0	0	NC	0
17～22	-1	-1	-1	0
23	0	0	NC	0

9.3.2 动态可信度量

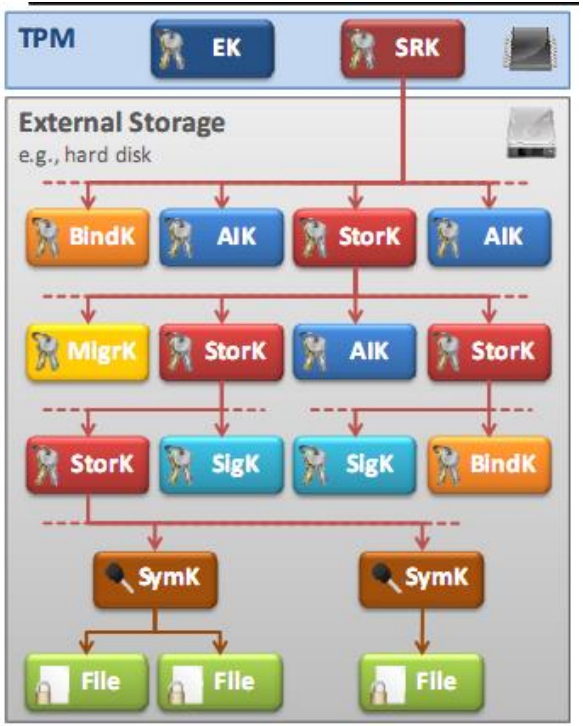
- 由于动态度量是在平台运行的任何时候进行的，因此，TPM需要区分其被动接受的度量值和指令是来源于可信操作系统，还是其他。
  - Locality0：用于向下兼容TPM1.1规范。
  - Locality1：当可信操作系统为应用程序提供执行环境时使用。
  - Locality2：可信操作系统运行时使用。
  - Locality3：辅助组件，它的使用是可选的，目前Intel已经使用，AMD没有使用。
  - Locality4：可信硬件（如：CPU），用于动态可信根。专门用于鉴别CPU发送的指令（如：SKINIT），是动态可信根的基础。

PCR	所属对象	PCRReset	PCR是否可以被Locality4,3,2,1,0重置	PCR是否可以被Locality4,3,2,1,0扩展
0～15	SRTM	0	0,0,0,0,0	1,1,1,1,1
16	Debug	1	1,1,1,1,1	1,1,1,1,1
17	Locality4	1	1,0,0,0,0	1,1,1,0,0
18	Locality3	1	1,0,0,0,0	1,1,1,0,0
19	Locality2	1	1,0,0,0,0	0,1,1,0,0
20	Locality1	1	1,0,1,0,0	0,1,1,1,0
21	可信OS控制	1	0,0,1,0,0	0,0,1,0,0
22	可信OS控制	1	0,0,1,0,0	0,0,1,0,0
23	应用程序	1	1,1,1,1,1	1,1,1,1,1

9.4 密钥管理

9.4.1密钥类型

- 不同类型密钥具有不同的安全需求，体现在密钥的产生、传输、存储、备份、销毁等主要环节
- 可迁移密钥与不可迁移密钥
- 对密钥按照功能划分并限制它们的使用，很大程度上可以增强系统的安全性
  - 存储密钥、签名密钥、平台身份认证密钥、绑定密钥、密封密钥、派生密钥、鉴别密钥
  - 这7类密钥可以粗略的分类为签名密钥和存储密钥



9.4.1密钥类型

可迁移密钥与不可迁移密钥

- 可迁移存储密钥并不局限于某个特定平台，可以由平台用户的控制下在平台之间迁移
- 不可迁移密钥则永久地与某个指定平台关联
- 不可迁移密钥能够用来加密保护可迁移密钥，反之则不行

9.4.2 密钥安全管理

- 由于TPM的空间有限，有些密钥以加密的形式存放到外部存储区中
- 当要使用这些密钥时，首先通过TSS在密钥缓冲池中查找并判断此密钥是否已经存在。若存在，则说明此密钥信息已经存在于TPM内部，不需要重新加载，直接可以在TPM中使用；否则就需要进行加载

问题：

- 1、可信计算的基本功能?
- 2、基于TPM的可信计算的作用是什么？能否保护系统的安全性?
- 3、信任度量时，组件度量的顺序是否可以改变？为什么?
- 4、为什么需要用AIK替代EK签名?

谢 谢