基于终端可信的分层可信平台模型研究

贾 佳[1] 周 毅[2] 董 慧[1] 张成岩[1]
(中国联通研究院 北京 100032)[1] (北京科技大学信息工程学院 北京 100083)[2]

摘 要 给出了基于终端可信的分层可信平台的研究,它是信息安全领域的一个新方向是一个结合平台硬件、安全操作系统、应用系统的完整的平台系统。基于终端可信的分层可信平台从客户终端系统的引导阶段出发构建完整的可信链,保证了可信计算平台的安全性,使信任的传递从用户到用户,进而实现真正的对于用户体验的可信计算。

关键词 可信度,可信计算,信任传递,信任分层
中图法分类号 TP309 **文献标识码** A

Research on Hierarchical Trusted Platform Based on Trusted Terminal

JIA Jia[1] ZHOU Yi[2] DONG Hui[1] ZHENG Cheng-yan[1]
(China Unicom Research Institute, Beijing 100032, China)[1]
(School of Information Engineering, University Science and Technology Beijing, Beijing 100083, China)[2]

Abstract This paper presented the research on hierarchical trusted platform based on trusted terminal, which is a new direction. The platform system is a combination of hardware, secure operating system and application system. This paper, through building a complete trusted chain from the start stage of the client terminal, undertaking the security of trusted computing platform which could make the trust transfer from user to user, realized the real trusted computing experience for users.

Keywords Credibility, Trusted computing, Trust transfer, Hierarchical trust

1 研究背景

从组成信息系统的服务器、网络、终端3个层面来看,现有的保护系统是逐层递减的,这说明人们往往把过多的注意力放在对服务器和网络的保护上,而勿略了对终端的保护,这显然是不合理的[1]。

据分析表明,80%以上的泄密和恶意攻击事件都是在内部客户端上发生的,黑客利用被攻击系统的漏洞窃取超级用户的访问权限,肆意进行破坏。注入病毒也是从终端发起的,病毒程序利用PC操作系统对执行代码不检查一致性的弱点,将病毒代码嵌入到执行代码程序,实现病毒的传播[2]。更为重要的是对合法的用户没有进行严格的访问控制,可以进行越权访问,造成不安全的事故[3]。现在的不安全问题多数都是PC机结构和操作系统不安全引起的,如果从终端操作平台实施高等级防范,保证每一个终端都是可信的,那么这些不安全因素将从终端源头被控制[4]。

面对这种局面,人们开始反思并进行了总结,当前的网络安全措施存在两个缺陷:一是只防外不防内;另一个是忽略了对终端的保护而终端往往是创建和存放重要数据的源头,绝大部分攻击事件都是从终端发起的事实[5]。要克服这两个缺陷,操作系统需要为终端提供一个可信的计算环境,确保终端在网络中的内容与行为的安全性。这是信息安全领域的新问题,也是最紧迫的问题[6]。

对于最常用的微机,只有从芯片、主板等硬件和BIOS、操作系统等底层软件综合采取措施,才能有效地提高其安全性[7]。正是基于这一思想催生了可信计算的迅速发展。可信计算的基本思想是在计算机系统中首先建立一个信任根,再建立一条信任链,一级测量认证一级,一级信任一级,把信任关系扩大到整个计算机系统,从而确保计算机系统的可信[8]。

只有从信息系统的硬件和软件的底层采取安全措施,从整体上采取措施,才能有效地确保信息系统的安全。以为信息系统提供可靠和安全的运行环境为主要目标,提出了一种能够超越预设安全规则,执行特殊行为的运行实体,就是可信计算。目前,体现系统与网络整体安全思想的可信赖计算与可信计算已经成为信息安全研究发展的必然趋势[9]。而我们要善于顺应和掌握产业发展的潮流,即对基于终端可信的可信计算的研究迫在眉睫。

2 分层可信计算平台

根据TCG的定义:如果实体按照预先设定的方式运行,则实体是可信的。如果网络中的行为与结果总是预期和可控的,则网络是可信的。TCG定义中网络是可信的主体即网络是值得信赖的[10]。

2.1 信任(可信)的定义

假设A为信任主体,B可信计算平台。

主体A对可信计算平台B的预期为一个集合$INTENTION=\{in\}$,可信计算平台B的运行实现结果为集合$IMPLEMENT=\{im\}$,主体A对于可信计算平台B的信任度为

贾 佳 硕士,E-mail:jia_jia@m165.com;周 毅 博士;董 慧 硕士;张成岩 高级工程师。

集合 $TRUST=\{t\}$，Ξ 表示可信计算平台 B 的对预期集合 $INTENTION$ 到实现结果集合 $IMPLEMENT$ 的运算规则，Ξ^{-1} 为 Ξ 的逆运算，$\circ$ 表示可信计算平台 B 的对实现结果集合 $IMPLEMENT$ 到预期集合 $INTENTION$ 差距的测量运算规则，$\circ^{-1}$ 为 $\circ$ 的逆运算。

信任

$\forall t\in TRUST; in\in INTENTION; im\in IMPLEMENT;$

$\exists im=\Xi(in), t=in\circ im, in=t\circ^{-1} im;$

$\Xi,\circ$ 由 A 决定；

则称信任主体 A 对于可信计算平台 B 信任，即 A 信任 B，记为 $A\{B\}_{trust}$。

推论 1

构成可信计算平台的 4 个要素就是：$in, im, \Xi, \circ$。

信任能控

如果信任主体 A 对可信计算平台 B 的预期 in 以及可信计算平台的对预期到实现结果运算规则 Ξ 由信任主体 A 所决定，定义信任主体 A 对于可信计算平台 B 信任能控[17]，即 A 信任 B 能控，记为 $A\{B\}_{contr}$。

推论 2

由 $im=\Xi(in)$ 可知，当 $A\{B\}_{contr}$ 时，im 可预期。

信任能观

如果可信计算平台 B 的实现结果到预期运算规则 $\circ$ 由信任主体 A 所决定，定义信任主体 A 对于可信计算平台 B 信任能观，即 A 信任 B 能观，记为 $A\{B\}_{obser}$。

推论 3

$A\{B\}_{contr}\cap A\{B\}_{obser}\Leftrightarrow A\{B\}_{trust}$ 则信任主体 A 对于可信计算平台 B 信任，即即 A 信任 B，记为 $A\{B\}_{trust}$。

证明：如果信任主体 A 对可信计算平台 B 的预期 in，实现结果 im，可信计算平台的对预期到实现结果运算规则 Ξ，以及可信计算平台的对实现结果到预期运算规则 $\circ$ 全部由信任主体 A 所决定，定义信任主体 A 对于可信计算平台 B 信任，即 A 信任 B。

推论 4

$B\{B\}_{trust}$

证明：如果可信计算平台 B 对自身的预期 in，实现结果 im，可信计算平台的对预期到实现结果运算规则 Ξ，以及可信计算平台的对实现结果到预期运算规则 $\circ$ 全部由可信计算平台 B 自身所决定。$B\{B\}_{trust}\Leftrightarrow B\{B\}_{contr}\cap B\{B\}_{obser}$

信任的属性：

假设主体对可信计算平台的预期实现程度的预期为集合 $DEGREE=\{de\}$，主体对可信计算平台的预期实现速度的预期为集合 $SPEED=\{SP\}$，可信计算平台的运行实现结果程度为集合 $SECURITY=\{se\}$，可信计算平台的运行实现效率为集合 $EFFICIENCY=\{ef\}$，

$\forall de\in DEGREE; sp\in SPEED; se\in SECURITY; ef\in EFFICIENCY;$

当 $in=(de,sp); im=(se,ef); t_{se}=de\circ se; t_{ef}=sp\circ ef$

$\exists t=in\circ im=(de,sp)\circ(se,ef)=((de\circ se),(sp\circ ef))=(t_{se},t_{ef})$

这里定义了信任的两大属性，即信任的安全属性 t_{se} 和效率属性 t_{ef}。

2.2 信任协商

假设对于一个可信计算平台的信任主体的预期为多个，那么主体信任预期为多个主体信任协商的结果：

$\forall in^k\in INTENTION, k=1,\cdots,l; in^k$ 为第 k 个信任主体对可信计算平台的预期；

$\forall de^k\in DEGREE, k=1,\cdots,l; de^k$ 为第 k 个信任主体对可信计算平台的实现程度预期；

$\forall sp^k\in SPEED, k=1,\cdots,l; sp^k$ 为第 k 个信任主体对可信计算平台的实现速度预期；

$$in=\mathop{\Delta}_{k=1}^{l} in^k=\mathop{\Delta}_{k=1}^{l}(de^k,sp^k)=(\mathop{\Delta}_{k=1}^{l} de^k,\mathop{\Delta}_{k=1}^{l} sp^k)=(de,sp)$$

式中，Δ 为信任协商的方法。

2.3 多可信计算平台信任链传递

假设可信计算平台集合 $TCP=\{tcp\}$ 中包含有限个元素有序，$\forall tcp^i\{tcp^{i+1}\}_{contr}, 1\leqslant i\leqslant n;$

A 为信任主体，$A\{TCP\}_{trust}$；

$\exists\circ^{-1}$ 为 $\circ$ 的逆运算；

$\forall in^i\in INTENTION, i=1,\cdots,n; in^i$ 为第 $i-1$ 个可信计算平台对为第 i 个可信计算平台的预期；in^1 表示主体对第一个可信计算平台的预期；

$\forall im^i\in IMPLEMENT, i=1,\cdots,n; im^i$ 为第 i 个可信计算平台对于第 $i-1$ 个可信计算平台的预期的实现结果；im^1 表示主体对第一个可信计算平台的预期实现结果；

$\forall t^i\in TRUST, i=1,\cdots,n; t^i$ 为第 $i-1$ 个可信计算平台对为第 i 个可信计算平台的信任度；t^1 表示主体对第一个可信计算平台的信任度；

假设对于一个可信计算平台是由多个相对独立的计算平台连接而成，那么主体对于可信计算平台整体的信任度：

$A\{TCP\}_{trust}\Rightarrow in=in^1, im=im^n;$

$\forall tcp^i\{tcp^{i+1}\}_{contr}\Rightarrow in^{i+1}=im^i, 1\leqslant i\leqslant n;$

$t=in^1\circ im^n=t^1\circ^{-1} im^1\circ im^n=t^1\circ^{-1} in^2\circ im^n$

$=t^1\circ^{-1} t^2\circ^{-1} im^2\circ im^n$

由递推可知

$$=\mathop{\circ^{-1}}_{i=1}^{n} t_i=\mathop{\circ^{-1}}_{i=1}^{n}(t_{se}^i,t_{ef}^i)=(\mathop{\circ^{-1}}_{i=1}^{n} t_{se}^i,\mathop{\circ^{-1}}_{i=1}^{n} t_{ef}^i)=(t_{se},t_{ef})$$

2.4 信任的分层

假设可信计算平台层集合 $STRATUM=\{stratum\}$ 中包含有限个元素有序，$\forall stratum^j\{stratum^{j+1}\}_{contr}, 1\leqslant j\leqslant m;$

A 为信任主体，$A\{STRATUM\}_{trust}$；

$\exists\circ^{-1}$ 为 $\circ$ 的逆运算；

$\forall in^j\in INTENTION, j=1,\cdots,m; in^j$ 为主体对可信计算平台第 j 层的预期；

$\forall im^j\in IMPLEMENT, j=1,\cdots,m; im^j$ 为主体对可信计算平台第 j 层的预期的实现结果；

$\forall t^j\in TRUST, j=1,\cdots,m; t^j$ 为主体对可信计算平台第 j 层的信任度；

$A\{STRATUM\}_{trust}\Rightarrow in=in^1, im=im^m;$

$\forall stratum^j\{stratum^{j+1}\}_{contr}\Rightarrow in^{j+1}=im^j, 1\leqslant j\leqslant m;$

$t=in^1\circ im^m=t^1\circ^{-1} im^1\circ im^m=t^1\circ^{-1} in^2\circ im^m$

$=t^1\circ^{-1} t^2\circ^{-1} im^2\circ im^m$

由递推可知

$$=\mathop{\circ^{-1}}_{j=1}^{m} t_j=\mathop{\circ^{-1}}_{j=1}^{m}(t_{se}^j,t_{ef}^j)=(\mathop{\circ^{-1}}_{j=1}^{m} t_{se}^j,\mathop{\circ^{-1}}_{j=1}^{m} t_{ef}^j)=(t_{se},t_{ef})$$

2.5 信任度(可信度)状态矩阵

假设可信计算平台集合 TCP=｛tcp｝中包含有限个元素

有序，$\forall tcp^i\{tcp^{i+1}\}_{contr}$，$1\leqslant i\leqslant n$；

对于$\forall tcp^i$的层集合$STRATUM=\{stratum\}$中包含有限个元素有序，$\forall stratum^j\{stratum^{j+1}\}_{contr}$，$1\leqslant j\leqslant m$；

A为信任主体，$A\{TCP\}_{trust}$，$A\{STRATUM\}_{trust}$；

$\forall in^{i,j}\in INTENTION, i=1,\cdots,n, j=1,\cdots,m$；$in^{i,j}$为主体对第$i$个可信计算平台第$j$层的预期；

$\forall im^{i,j}\in IMPLEMENT, i=1,\cdots,n, j=1,\cdots,m$；$im^{i,j}$为主体对第$i$个可信计算平台第$j$层的预期的实现结果；

$\forall t^{i,j}\in TRUST, i=1,\cdots,n, j=1,\cdots,m$；$t^{i,j}$为主体对第$i$个可信计算平台第$j$层的信任度；

$$
\begin{aligned}
t&=\begin{bmatrix} t^{1,1} & \cdots & t^{1,m} \\ \vdots & t^{i,j} & \vdots \\ t^{n,1} & \cdots & t^{n,m} \end{bmatrix} \\
&=\begin{bmatrix} in^{1,1} & \cdots & in^{1,m} \\ \vdots & in^{i,j} & \vdots \\ in^{n,1} & \cdots & in^{n,m} \end{bmatrix}\circ\begin{bmatrix} im^{1,1} & \cdots & im^{1,m} \\ \vdots & im^{i,j} & \vdots \\ im^{n,1} & \cdots & im^{n,m} \end{bmatrix} \\
&=\begin{bmatrix} in^{1,1}\circ im^{1,1} & \cdots & in^{1,m}\circ im^{1,m} \\ \vdots & in^{i,j}\circ im^{i,j} & \vdots \\ in^{n,1}\circ im^{n,1} & \cdots & in^{n,m}\circ im^{n,m} \end{bmatrix} \\
&=\begin{bmatrix} (t_{se}^{1,1},t_{ef}^{1,1}) & \cdots & (t_{se}^{1,m},t_{ef}^{1,m}) \\ \vdots & (t_{se}^{i,j},t_{ef}^{i,j}) & \vdots \\ (t_{se}^{n,1},t_{ef}^{n,1}) & \cdots & (t_{se}^{n,m},t_{ef}^{n,m}) \end{bmatrix} \\
&=\left(\begin{bmatrix} t_{se}^{1,1} & \cdots & t_{se}^{1,m} \\ \vdots & t_{se}^{i,j} & \vdots \\ t_{se}^{n,1} & \cdots & t_{se}^{n,m} \end{bmatrix},\begin{bmatrix} t_{ef}^{1,1} & \cdots & t_{ef}^{1,m} \\ \vdots & t_{ef}^{i,j} & \vdots \\ t_{ef}^{n,1} & \cdots & t_{ef}^{n,m} \end{bmatrix}\right)=(t_{se},t_{ef})
\end{aligned}
$$

3 基于终端可信的分层可信平台的实现

TCG提出的可信计算平台的体系结构相对简单，它仅仅从硬件上引入了TPM模块，因此要在其提出的可信计算平台上建立完整的“可信计算环境”还需要诸多技术手段的支持。而其中的关键是操作系统的支持。而当前主流操作系统，Windows操作系统和类Unix操作系统，在支持可信计算上主要存在以下的问题：

- 没有良好的进程隔离机制；
- 缺少强制访问控制机制；
- 忽视最小特权策略；
- 没有为输入输出提供可信路径。

要克服主流操作系统存在以上的问题，必须站在体系结构的角度，研究以操作系统为中心的支持可信计算的终端安全体系结构。

3.1 终端基本信任链

终端基本信任链（见图1）中，信任主体为具有相同INTENTION的USER。可信计算环境的基本信任链可以抽象为由服务请求（Service Request）的可信计算实体SERE和服务提供（Service Supply）的可信计算实体SESU这两种可信计算实体的集合。

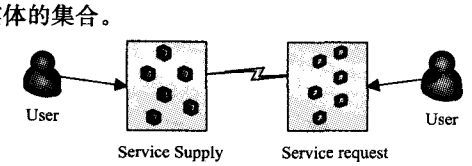

图1 终端基本信任链

其中$SERE=\{sere\}$，$SESU=\{server\}$，$USER=\{user\}$；

Ξ_{sere}为对应可信计算实体$SERE$信任实现运算规则；

$\circ_{sere}$为对应可信计算实体$SERE$信任度量运算规则；

Ξ_{sesu}为对应可信计算实体$SESU$信任实现运算规则；

$\circ_{sesu}$为对应可信计算实体$SESU$信任度量运算规则；

$USER$，$SESU$，$SERE$按顺序构成基本信任链，

$SESU\{SERE\}_{trust}$；

可信计算环境的可信度：

$t=im\circ in=t_{sesu}+t_{sere}+t_{sesu\to sere}$

3.2 基本可信层

可信计算环境的层是可信计算实体相同运算规则的抽象。计算环境经过长期的发展形成了相对稳定的基本可信层。图2给出可信计算环境的层结构，分为基本可信层（Basic layer）和自定义可信任（User-defined layer）。可信计算环境的基本可信层可以抽象为统一可扩展固件接口层UEFI（Unified Extensible Firmware Interface）、操作系统层OS（Operating System）和应用层App（Application）集合。

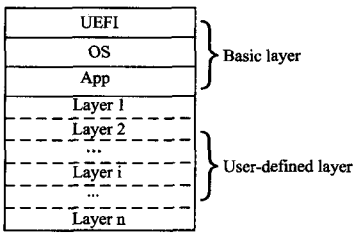

图2 基本可信层

其中$UEFI=\{uefi\}$，$OS=\{os\}$，$App=\{app\}$；

App，OS，$UEFI$按顺序构成基本可信层信任链；

$App\{OS\}_{contr}$，$OS\{UEFI\}_{contr}$；

$App\{OS\}_{obser}$，$OS\{UEFI\}_{obser}$；

$App\{OS\}_{trust}$，$OS\{UEFI\}_{trust}$；

Ξ_{uefi}为对应可信层$UEFI$信任实现运算规则；

$\circ_{uefi}$为对应可信层$UEFI$信任度量运算规则；

Ξ_{os}为对应可信层OS信任实现运算规则；

$\circ_{os}$为对应可信层OS信任度量运算规则；

Ξ_{app}为对应可信层App信任实现运算规则；

$\circ_{app}$为对应可信层App信任度量运算规则；

可信计算环境的可信度：

$t=im\circ in=(t_{app},t_{os},t_{uefi})$

3.3 可信计算环境的可信状态

可信状态（见图3）中描述了可信计算实体SERE和SESU中可信层的对应情况。其中有：

$\Xi_{(sere,uefi)}$为对应可信计算实体$SERE$可信层$UEFI$信任实现运算规则；

$\circ_{(sere,uefi)}$为对应可信计算实体$SERE$可信层$UEFI$信任度量运算规则；

$\Xi_{(sere,os)}$为对应可信计算实体$SERE$可信层OS信任实现运算规则；

$\circ_{(sere,os)}$为对应可信计算实体$SERE$可信层OS信任度量运算规则；

$\Xi_{(sere,app)}$为对应可信计算实体$SERE$可信层App信任实现运算规则；

$\circ_{(sere,app)}$ 为对应可信计算实体 *SERE* 可信层 *App* 信任度量运算规则；

$\Xi_{(sesu,uefi)}$ 为对应可信计算实体 *SESU* 可信层 *UEFI* 信任实现运算规则；

$\circ_{(sesu,uefi)}$ 为对应可信计算实体 *SESU* 可信层 *UEFI* 信任度量运算规则；

$\Xi_{(sesu,os)}$ 为对应可信计算实体 *SESU* 可信层 *OS* 信任实现运算规则；

$\circ_{(sesu,os)}$ 为对应可信计算实体 *SESU* 可信层 *OS* 信任度量运算规则；

$\Xi_{(sesu,app)}$ 为对应可信计算实体 *SESU* 可信层 *App* 信任实现运算规则；

$\circ_{(sesu,app)}$ 为对应可信计算实体 *SESU* 可信层 *App* 信任度量运算规则。

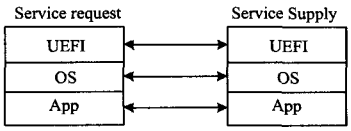

图 3　可信状态

可信计算环境的可信度：

$$t = im \circ in = \begin{bmatrix} t_{sesu,app} & t_{sere,app} \\ t_{sesu,os} & t_{sere,os} \\ t_{sesu,ufei} & t_{sere,ufei} \end{bmatrix}$$

3.4　信任的主要属性

由定义可知信任托付和信任托付实现的差距表示可信度。可信度的差距主要由信任托付实现的程度、信任托付实现的速度、信任托付实现的成本 3 个方面来衡量(见图 4)。

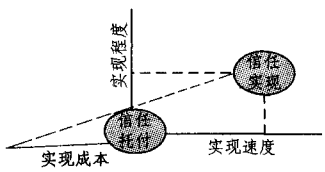

图 4　信任的主要属性

主体对可信计算实体的信任托付实现程度的预期为集合 $DEGREE=\{de\}$；

主体对可信计算实体的信任托付实现速度的预期为集合 $SPEED=\{sp\}$；

主体对可信计算实体的预期实现价值的预期为集合 $VALUE=\{va\}$；

可信计算实体的运行实现结果程度为集合 $SECURITY=\{se\}$；

可信计算实体的运行实现结果速度为集合 $EFFICIENCY=\{ef\}$；

可信计算实体的运行实现结果成本为集合 $COST=\{co\}$；

$\forall de \in DEGREE, sp \in SPEED, se \in SECURITY, ef \in EFFICIENCY, va \in VALUE, co \in COST$；

当 $in=(de,sp), im=(se,ef), t_{se}=de \circ se, t_{ef}=sp \circ ef, t_{co}=va \circ co$；

$\exists t=in \circ im=(de,sp,va) \circ (se,ef,co)=(de \circ se),(sp \circ ef),(va \circ co)=(t_{se},t_{ef},t_{co})$。

这里定义了信任的 3 大属性，即信任的安全属性 t_{se}、效率属性 t_{ef} 和成本属性 t_{co}。

信任的主体偏好为 $ORDINAL=\{or\}=\{(se,ef,co),(se,co,ef),(ef,co,se),(ef,se,co),(co,se,ef),(co,ef,se)\}$。

3.5　基于终端可信的分层可信计算平台

可信计算平台(见图 5)由网络融合接口卡 NCIC(Network Convergence Interface Card)、人机信任接口 MMTI(Man-Machine Trust Interface)、可信存储根 TRS(Trusted Root Storage)、传统的计算单元中央处理器 CPU、快速存储器 Memory、控制单元 Control、显示设备 Display、内部设备(Embdded Devices)、可移动设备(Removable Devices)构成。

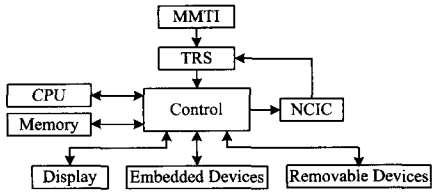

图 5　分层可信计算平台

可信计算平台是构成可信计算环境的基本物理单元。TCG 方案中信任的基础在于可信计算平台自身内部的 TPM。然而 TPM 在可信计算平台的物理结构中还处于从属的位置，在可信计算平台加电的过程中还是要先由系统 BIOS 提供引导后移交控制权。此外，TPM 的安全防护能力是基本的密码学保护和校验，这与计算环境中具有多种安全功能的安全服务器的安全防御和校验能力不可相提并论。

本文提出的基于 TRS 和 NCIC 的可信计算平台，其核心思想是将计算环境中最可信的计算模块通过 NCIC 引入可信计算平台的 TRS 中，从而建立基本可信根、基本信任链、基本可信层、基本可信状态。具体过程是由可信平台的使用者通过 MMTI 确定信任预期的权利义务关系，从而决定 TRS 中对应的存储单元的可信根。而可信根由 NCIC 从计算环境中引入。

其中可信存储根 TRS 由 4 个物理上隔离(Physical Isolation)的存储空间构成。分别对应基于自然权利义务关系(Nature Relation)、基于道德权利义务关系(Moral Relation)、法定权利义务关系(Rule Relation)的可信根和独立的数据单元。采用物理隔离实现较为简单，同时免去了数据交叉带来的数据校验开销。其中每一种权利义务关系中由逻辑上隔离(Logical Isolation)的基本可信层构成。基于自然权利义务关系存储根由统一可扩展固件接口 UEFI、操作系统 OS、应用程序 App(Applications)构成。基于道德权利义务关系存储根由统一可扩展固件接口 UEFI、操作系统 OS、多主体应用程序 GApp(Groupware Applications)构成。基于道德权利义务关系存储根由统一可扩展固件接口 UEFI、操作系统 OS、公众应用程序 GApp(PApp Public Applications)构成(见图 6)。

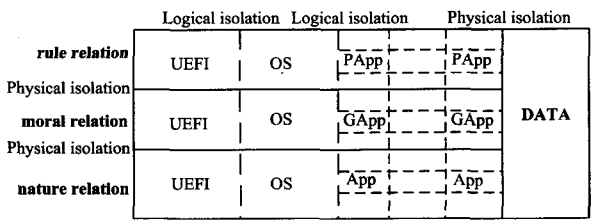

图 6　可信存储根

TRS 中每一种权利义务关系的可信根与对应的网络通过 NCIC 直接连接(见图 7)。其中，基于自然权利义务关系的

可信存储根与广电网直接相连，基于道德权利义务关系的可信存储根与电信网络相连，法定权利义务关系的可信存储根与计算机网络相连。广电网以广播的方式在物理信道中直接传输法定权利义务关系可信根。当前电信网和计算机网络的融合已经比较充分，其对应的基于自然权利义务关系的可信存储根和基于道德权利义务关的可信存储根传输，在法定权利义务关系可信根引导下由控制器完成。

信任层次	可信存储	可信网络
法定权利义务关系	可信UEFI 可信OS 可信应用	广电网
道德权利义务关系	可信UEFI 可信OS 可信应用	电信网
自然权利义务关系	可信UEFI 可信OS 可信应用	计算机网络

图 7 可信网络连接

3.6 基于终端可信的分层可信计算平台与当前计算平台的结合

当前计算平台(见图 8)由基本输入输出 BIOS(Basic Input-Output System)、光盘驱动器 CD(Compact Disc)、硬盘存储器 HD(Hard Disk)、网络接口卡 NIC(Network Interface Card)和计算单元中央处理器 CPU、快速存储器 Memory、控制单元 Control、显示设备 Display、内部设备(Embdded Devices)、可移动设备(Removable Devices)构成。

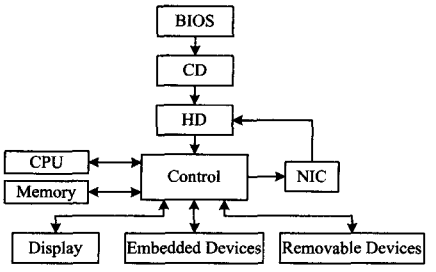

图 8 分层可信计算平台与当前计算平台的结合

BIOS 设置中将系统基本输入输出后计算平台的控制权交给光盘驱动器。CD 中可以存储 3 种权力义务关系可信根的任何一种，具体由计算系统的使用者决定。HD 对计算平台的控制权接管由 CD 决定。这样在当前计算平台中就可以建立可信存储根，从而完成当前计算平台和可信计算平台的结合。

如图 9 所示，可信计算平台之间的信任关系的转化对于可信计算环境的输入信任预期而言是不断变化的。由信任预期的使用率决定了可信计算实体的可信计算关系，基于自然权利义务关系(Nature Relation)、基于道德权利义务关系(Moral Relation)、法定权利义务关系(Rule Relation)。当这种信任预期产生外部性时，就必须通过信任协商来解决。如果是无损信任协商，可信计算实体在基于人选择的可信计算模型下通过层扩张开始信任收敛。如果是正信任协商，可信计算实体在信任预期完全实现的层扩张后，在基于人选择的可信计算模型下通过层扩张开始信任收敛。如果是负信任协商，可信计算实体在信任预期只能部分实现的层扩张后，在基于人选择的可信计算模型下通过层扩张开始信任收敛。在这种动态平衡中基于 3 种权利义务关系的预期和实现越趋于一致可信计算环境越公正。

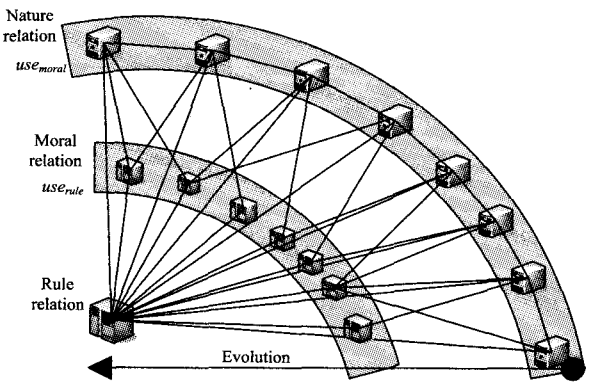

图 9 可信计算环境

4 仿真实验

构建现实网络计算环境中网络节点的信任关系，采用查询周期模型进行仿真。现实网络计算环境中，点对点构建网络节点的信任关系为多对多的对应关系。这种对应关系相对于其它对应关系而言是最为复杂的，对于各种对应关系将其进行仿真实验具有代表性。

实验中，每次仿真由若干个仿真周期组成，在每个仿真周期中，网络中的节点可以发起文件查询并对查询进行响应。每个周期中，节点可能出于活动状态或离线状态，在活动状态节点可能发出请求，并对收到的请求进行处理。查询请求进行广播，通过 TTL 控制查询的规模。接收到查询消息的节点查看是否对其应答，发起查询的节点等待接收响应并从响应列表中选择信任值最高的节点下载文件，直到交易成功或者所有响应均告失败。在仿真中，网络中的节点分为两大类：正常节点和恶意节点。其中，正常节点提供真实的服务和评价，而恶意节点提供虚假的服务和反馈。仿真实验描述了 0%、10%、20%、30%、40%、50%恶意节点的 3 种实验场景，对每 20 个查询周期选一个进行比较。

4.1 仿真结果

(1)下载成功率对比结果

描述了 0%、10%、20%、30%、40%、50%恶意节点随即下载过程中对于下载成功率对比的仿真结果。对每 20 个查询周期选一个进行比较，如图 10 所示。

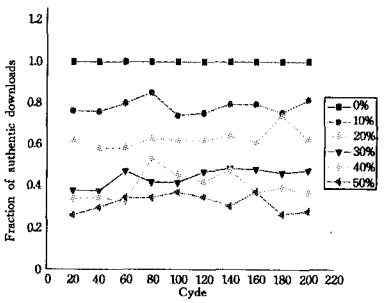

图 10 下载成功率对比

(2)成功下载次数对比结果

描述了 0%、10%、20%、30%、40%、50%恶意节点随即下载过程中对于成功下载次数对比的仿真结果。对每 20 个查询周期选一个进行比较，如图 11 所示。

(下转第 123 页)

EKE协议要求用户发送额外的消息来证实其知道口令 P 的明文，使得攻击者在获取了用于加密密钥交换的口令，哈希值在 $H(P)$ 的情况下也无法向主机冒充用户。用户选择的口令通常是一些便于记忆的弱口令，本文构建的基于环 Z_n 上圆锥曲线用 ElGamal 签名算法实现的 A-EKE 协议，不需要协议双方保存口令明文，能抵抗主动攻击、字典攻击和破坏口令文件的攻击，使得即使在用户选用弱口令的情况下也能保证身份认证的有效性和密钥协商的安全性，从而保证信息传输的安全性，并且能够有效减少计算机处理开销，在加密处理速度、通信带宽和对硬件的要求方面都有更大的优势，对于在网络通信中实现高效的身份认证和安全信息传递有一定的现实意义。

参 考 文 献

[1] 孙琦，张起帆，彭国华. 计算群元的整数倍的一种算法及其在公钥密码体制中的应用[A]//密码学进展——ChinaCrypt'2002：第七届中国密码学学术会议论文集[C]. 北京：电子工业出版社，2002

[2] 孙琦，朱文余，王标. 环 Z_n 上圆锥曲线和公钥密码协议[J]. 四川大学学报：自然科学版，2005，42(3)

[3] 王标，朱文余，孙琦. 基于剩余类环 Z_n 圆锥曲线的公钥密码体制[J]. 四川大学学报：工程科学版，2005，37(5)

[4] Bellovin S M，Merritt M. Encrypted Key Exchange：Password-Base Protocols Secure Against Dictionary Attacks[C]// Proceedings of the 1992 IEEE Computer Society Conference on Research in Security and Privacy. 1992

[5] Morris R H，Thompson K. Unix password security[J]. Communications of the ACM，1979，22

[6] A proposed Federal Information Processing Standard for digital signature standard (DSS)[S]. Docket No. 910807- 1207，RIN 0693-AA86

[7] Bellovin S M，Merritt M. Agumented Encrypted Key Exchange：a Password-Based Protocol Secure Against Dictionary Attacks and Password File Comprise[C]// Proceedings of the 1st ACM Conference on Computer and Communications Security. 1993

[8] 张明志. 用圆锥曲线分解整数[J]. 四川大学：自然科学版，1996，33(4)

(上接第 113 页)

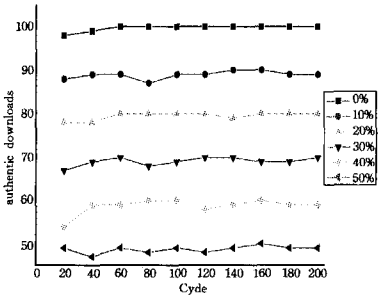

图 11 成功下载次数对比

(3)不成功下载次数对比结果

描述了 0%、10%、20%、30%、40%、50%恶意节点随即下载过程中对于不成功下载次数对比的仿真结果。对每 20 个查询周期选一个进行比较，如图 12 所示。

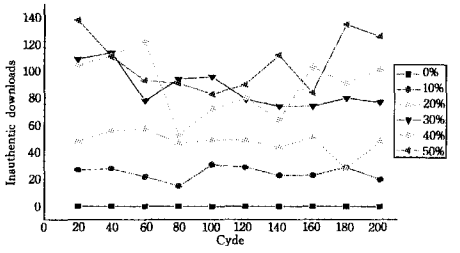

图 12 不成功下载次数对比

4.2 结果分析

由上述仿真结果可知，基于终端可信的分层可信平台通过对点对点文件共享网络中的节点可信度直接作用，对下载成功率、成功下载次数均有明显提高，对不成功下载次数显著下降。点对点文件共享网络计算环境下，下载成功率、成功下载次数、不成功下载次数都可以代表其可信度。仿真系统数据结果说明，提高可信节点的比例对于提高可信计算环境整体的可信度有显著效果。

结束语 现有的 TCG 规范的可信计算模型是一种基于终端可信的通用模型，由于可信根以硬件的方式进行实现，因此要求对现有的计算机硬件体系结构以及硬件驱动软件做出较大改动，模型实现存在较大困难。所以，研究与现行的计算机网络相适应的可信计算模型，具有很强的理论和实践意义。

本文给出的基于可信终端的分层可信计算平台，对 TCG 所定义的可信计算模型进行了结构上的改动，通过把原有的 TPM 模块逻辑化为一个信任度量的状态矩阵，较大地降低了系统实施的复杂性。由于该模型中信任的度量可以灵活地实施在任意相对独立的可信计算平台的任意逻辑层，因此可以从信任根的完整度量节点出发构建完整的可信链，保证可信计算平台的安全性。同时，广义的可信计算平台包括了所有的人机接口，进而实现真正的用户对用户可信。

参 考 文 献

[1] Shangyuan G，et al. Trust management and service selection in pervasive computing environments[C]// International Conference on Computational Intelligence and Security Workshops. Dec. 2008：15-19

[2] Feng D，Qin Y. Research on attestation method for trust computing environment[J]. Jisuanji Xuebao/Chinese Journal of Computers，2008，31(9)：1640-1652

[3] TCG. TCG Specification Architecture Overview. Ver1. 4 [EB/OL]. (2007-8-2) [2008-11-24]

[4] Brizek J，Khan M，Seifert J P，et al. A Platform-level Trust-Architecture for Hand-held Devices [C]// 2005 Workshop on Cryptographic Advances in Secure Hardware (CRASH). Belgium，2005

[5] Eisenbarth T，Güneysu T，Paar C，et al. Reconfigurable Trusted Computing in Hardware [C]// Proceedings of the 2007 ACM workshop on Scalable trusted computing. VA，USA，2007：15-20

[6] 林闯，田立勤，王元卓. 可信网络中用户行为可信的研究[J]. 计算机研究与发展，2008(12)

[7] 邓晓衡. iVCE 中基于可信评价的资源调度研究[J]. 计算机学报，2007(10)

[8] 章勤. 基于网格环境的可信计算平台共享模型[J]. 华中科技大学学报：自然科学版，2007(12)

[9] 林闯. 可信网络研究[J]. 计算机学报，2005，28(5)：751-758

[10] TCG. TCG TNC(Trusted Network Connect) Architecture for Interoperability. Ver1. 2 [EB/OL]. (2007-5-2) [2008-11-24]