doi:10.3969/j.issn.1000-436x.2014.11.011

# 抗隐蔽通道的网络隔离通信方案

李凤华[1]，谈苗苗[2]，樊凯[2]，耿魁[1,2]，赵甫[3]
（1. 中国科学院 信息工程研究所 信息安全国家重点实验室，北京 100093；
2. 西安电子科技大学 通信工程学院，陕西 西安 710071；3. 北京航天数控系统有限公司，北京 100854）

**摘 要**：随着网络技术的发展，广泛互联互通的异构网络间的信息交互越来越频繁。为有效保障信息跨网安全实时交换，提出了一种抗隐蔽通道的网络隔离通信方案(NICS, network isolation communication scheme)。建立了 NICS 理论模型，基于信息论理论证明了该方案的正确性，并给出了具体的实施方案。安全特性分析表明，NICS 可有效解决不同网络的通信协议均存在潜在的数据分组大小隐蔽通道与状态信息隐蔽通道的问题；在交互相同信息量的前提下，可实现与物理隔离等价的抗隐蔽通道的安全效果。

**关键词**：网络隔离；隐蔽通道；数据分组长度；状态信息
**中图分类号**：TN 929 **文献标识码**：A **文章编号**：1000-436X(2014)11-0096-11

# Network isolation communication scheme to resist against covert channel

LI Feng-hua[1], TAN Miao-miao[2], FAN Kai[2], GENG Kui[1,2], ZHAO Fu[3]
(1. State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Science, Beijing 100093, China;
2.School of Telecommunication Engineering, Xidian University, Xi'an 710071, China
3. Beijing Aerospace Numerical Control System Co.,Ltd, Beijing 100854, China)

**Abstract:** With the rapid development of network technologies, real-time information exchanging between heterogeneous networks becomes more frequently. To effectively guarantee the secure and real-time information exchanging crossing different networks, a network isolation communication scheme (NICS) is proposed to resist against covert channel. A newly theoretical model of NICS is designed and proved based on the information theory, and followed with a specific solution. Security analysis indicates that the NICS is able to effectively solve problems of the potential packet lengths' covert channel (PLCC) and the status covert channel (SCC) in most of the existing work; and, given similar amount of information for exchanging, the NICS can achieve equivalent security degree with the physical isolation in terms of resisting against the covert channel.

**Key words:** network isolation; covert channel; length of the data packet; status information

## 1 引言

随着网络技术和信息技术的快速发展与广泛应用以及服务模式的不断创新，推动了异构网络的广泛互联互通和信息实时交互的需求，在技术上已经达到了通过“网络之网络”访问“系统之系统”。

但在现实中，网络并没有形成大规模的广泛互联互通，其根本原因在于不同的网络之间通过物理隔离来防止网络攻击。因此，在安全隔离尚未有效解决之前，只能借助人工摆渡来实现不同网络之间

**收稿日期**：2014-08-26；**修回日期**：2014-11-03
**基金项目**：国家自然科学基金资助项目（61170251）；国家高技术研究发展计划（“863“计划）基金资助项目（2012AA013102, 2012AA01A401）；数字版权保护技术研发工程基金资助项目(1681300000119)
**Foundation Items:** The National Natural Science Foundation of China (61170251); The National High-Tech R&D Program of China (863 Program) (2012AA013102, 2012AA01A401); The Major Science and Technology Project of Press and Publication-Research and Development (1681300000119)

的信息安全交换，这并不能满足泛在网络环境下对信息广泛交互、实时交互、安全交互的应用需求，已制约信息化的广泛应用。随着各种云服务应用模式以及工业控制系统的自动化、智能化发展，尤其4G/5G 时代的到来，信息安全交互已渗透到人们日常生产、工作、学习和生活方式中，其信息交互的高实时、高可靠、高安全需求与尚未完善的隔离方案之间的矛盾日益突出，迫切需要一种达到与物理隔离安全效果等价的网络隔离通信方案。

依据内、外网络之间是否直接相连，隔离可划分为物理隔离和逻辑隔离。

传统的物理隔离方法通常需要借助人工操作方式(如采用 U 盘、光盘等介质)实现数据在 2 个网络间的交互。该方法实时性弱、效率低、成本高，不能防止病毒在不同网络间摆渡。

逻辑隔离主要通过逻辑隔离器实现。逻辑隔离器是一种介于不同网络间的隔离部件，被隔离的两端仍然存在物理上有数据通道连线，但通过技术手段保证被隔离的两端没有数据通道，一般使用协议转换[1]、数据格式剥离[2]和数据流控制[3]等方法在 2 个逻辑隔离区域间传输数据。

现有防火墙、安全网关、隔离网闸等基于策略或专有协议的隔离措施可以提升网络环境的安全性，但这些技术手段还存在不足。例如，防火墙是网络层边界检查工具，可以设置规则对内部网络进行安全防护，却难以检测到来自内部的攻击；安全网关理论上基于逻辑机制，有被逻辑实体控制的威胁；安全隔离网闸重点保护内部网络，但并没有达到与物理隔离等价的安全特性。它们均不能避免隐蔽通道的存在，难以满足跨网、跨行业信息系统互联和数据共享所需的安全要求。Lampson 首先提出隐蔽通道的概念[4]，其后有多种定义从不同的角度来描述隐蔽通道的本质和特性。美国可信计算机安全评价标准指出，隐蔽通道是让一个进程以不受安全策略控制或以违反系统安全策略的方式传递消息的信息通道[5]。亦被定义为违背系统设计者原本意图，从系统的一个用户传送信息到另一个用户的机制[4]。前期大量研究者们均是基于特定环境进行分析与探讨[6~8]，难以满足当前泛在网络环境下的安全隔离通信需求。

在泛在网络环境下的信息交互系统中，隐蔽通道是多用户系统之间普遍存在的间接信息扩散通道，它的存在使攻击者能够绕过所有的安全保护机制来破坏系统的机密性，使信息系统的保密性受到严重威胁。前期已有通过各种方案对隐蔽通道进行的多种分类[9~12]。此处，依据其形成的原因，隐蔽通道可包括如下 3 种。

1) 消息内容自身隐蔽通道。主要是指借由消息内容及其大小的组合变化，而形成所传递消息之外的隐藏信息的通道，即通过改变交互消息内容自身来传递附加信息。

2) 数据分组大小隐蔽通道（PLCC, packet lengths' covert channel）。主要指借由通信双方控制传输过程中数据分组的大小，以形成所传递消息之外的附加信息[13]。

3) 状态信息隐蔽通道(SCC, status covert channel)。主要指通信双方控制纠错重发的时机、次序及次数等，以形成所传递消息之外的附加信息。

由于内容的长度大小在每次交互时都是基本确定的，在任何广泛自由的通信系统中都是无法控制的，故第一类隐蔽通道存在于所有的网络通信中，不可避免。第 2、3 类隐蔽通道可以通过适当方式予以避免，本文将针对第 2 和第 3 种情况，提出相应的解决方案。

旨在保障传输内容不被非法窃取、伪造、篡改等的安全通信协议工作在通信系统协议栈之上，安全通信协议并不能解决通信系统中潜在的隐蔽通道问题。隐蔽通道安全问题与常规安全通信协议解决的安全问题是 2 个不同的方向，不可相互替代。本文针对泛在网络环境中抗隐蔽通道的安全隔离需求，首先对通信系统进行形式化描述，并在充分考虑现有网络隔离方案中存在隐蔽通道可能被敌手利用的前提下，提出 NICS 方案。NICS 以硬件隔离装置为基础，确保拆分前后的数据分组长度之间不存在固定的映射关系，实现抗 PLCC 功能；为每个作业分组全程携带校验码及纠错码，可保障数据传输高可靠性，并实现抗 SCC 功能。然后，在理论模型上证明了 NICS 具有抗上述 2 类隐蔽通道的特性，安全性分析得出在交互相同信息量的前提下，可实现与物理隔离等价的抗隐蔽通道的安全效果。最后，给出的实例及在某项目中的具体实施均进一步验证了 NICS 方案的高安全、高可靠、高实时及高可行性。

## 2　相关工作

### 2.1　隔离技术

隔离概念是在为了保护高安全级网络的信息安

全的情况下提出，许多旨在保障网络安全的隔离方案也相继提出[14~16]。共经历了如下 5 代隔离技术。

1) 完全隔离[17]。该方法是基于空间的隔离技术，一般采用 2 套设备，分别连接内部网和外部网，通过人工摆渡、磁盘、光盘等交换内外网信息，使内网处于信息孤岛状态。这不仅带来信息交流的不便和成本的提高，也给维护和使用带来了极大的不便。

2) 硬件卡隔离。在客户端增加一块硬件卡，客户端硬盘或其他存储设备首先连接到该卡，然后再转接到主板上，通过该卡能控制客户端硬盘或其他存储设备。而在选择不同的硬盘时，同时选择了该卡上不同的网络接口，连接到不同的网络，故亦称为硬盘隔离技术。但该隔离方式通常具有兼容性问题，且由于两套系统共用内存，亦存在安全隐患。

3) 数据转播隔离。利用转播系统分时复制文件的途径来实现隔离，切换时间久，甚至需要手工完成，不仅明显减缓了访问速度，更不支持常见的网络应用，失去了网络存在的意义。

4) 空气开关隔离。该方法是一种基于时间的隔离方法。通过使用单刀双掷开关，使内外部网络分时访问临时缓存器来完成数据交换，在安全和性能上仍存在许多问题。

5) 安全通道隔离。此技术通过专用通信硬件和专有安全协议等安全机制来实现内外部网络的隔离和数据交换，不仅解决了传统隔离技术存在的问题，有效地把内外部网络隔离开来，而且高效地实现了数据的安全交换，可支持多种网络应用[18]。

以上隔离技术各有利弊，虽然能满足人们的一定事务需求，但均没有考虑隐蔽通道所带来的信息泄露、恶意信息传播等威胁。为此需要研究一种能够抵抗 PLCC 和 SCC 2 种隐蔽通道的网络隔离通信方案。

### 2.2 隐蔽通道

自 Lampson 首次提出了用于秘密通信的隐蔽通道以来，已有大量研究者对各种类型的隐蔽通道进行分析。如基于处理器的隐蔽通道[12]，基于 DNS 的隐蔽通道[19]等。其中一个很重要的分支就是上述 PLCC。

该概念由 Padlipsky 在文献[13]中提出，他将链路层的数据帧长度与符号联系起来隐藏秘密信息。在文献[20]中 Girling 通过用 256 个字符表示链路层的 256 种数据帧长度，每种数据帧长代表一个字节的消息；文献[9]中 Liping 提出了另一种模型，需要发送方和接收方一起协商将要发送的数据分组长度，并将其设为默认参考，每发送一次都对其进行相应的更新，从而保证了发送数据分组长度分布非常类似于正常网络数据分组。文献[21]给出了一种基于数据分组长度的隐蔽通信模型，并分析了该模型下的隐蔽信道容量。文献[22]针对原有隐蔽通道易于被通过长度分析监测出的问题，提出了一种基于相关矩阵的数据分组长度分布正常，所形成的 PLCC 难以被监测到。

上述文献均为 PLCC 的相关研究，并未涉及 SCC，且多以构建模型为主，缺乏系统有效的解决方案。本文提出的 NICS，通过采取相应的措施可具备同时抵抗 PLCC 和 SCC 的功能。

## 3 模型与方案构造

### 3.1 理论模型

如图 1 所示，$A$、$B$ 为 2 个需要进行数据交互却不可直接相连的网络，$C_1$、$C_2$、$C_3$、…、$C_p$ 为隔离控制单元，组成隔离控制装置 $M$。

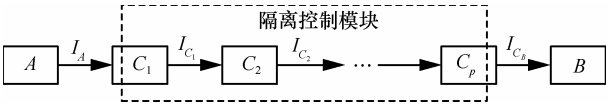


图 1 理论模型

网络 $A$ 与控制单元 $C_1$ 之间的协议遵从网络 $A$ 的通信协议，控制单元 $C_p$ 与网络 $B$ 之间的协议遵从网络 $B$ 的通信协议，任意 2 个相邻的控制单元 $C_i$ 与 $C_{i+1}(i=1,\cdots,p-1)$之间均为自定义的私有协议。

### 3.2 攻击模型

如图 1 所示，网络 $A$ 与控制单元 $C_1$ 直接相连，网络 $B$ 与控制单元 $C_p$ 直接相连。假设网络 $A$、网络 $B$ 均不可信，且为了保证整个系统更好的适应性和兼容性，允许对控制单元 $C_1$ 和 $C_3$ 进行软件更新，这将带来潜在的安全威胁，控制单元 $C_1$ 与网络 $A$ 直接相连导致理论上控制单元 $C_1$ 有被来自网络 $A$ 的敌手攻破的威胁；控制单元 $C_p$ 与网络 $B$ 直接相连导致理论上控制单元 $C_p$ 有被来自网络 $B$ 的敌手攻破的威胁；已知各控制单元 $C_i$ 与 $C_{i+1}(i=1,\cdots,p-1)$ 之间均为结构化的自定义私有协议，只传内容不传信令，且控制单元 $C_i(i=2,\cdots,p-1)$的代码是固化且不允许远程在线更新的，故不会存在在线升级或代码更换时潜在的病毒入侵或恶意攻击等安全隐患。且

完全依照等保四级中结构化设计来设计 $C_2$，可对流经数据进行内容检查，因此理论上可以假设其安全可靠。

基于以上安全假设，就文中重点讨论的 2 种隐蔽通道存在的攻击模型进行如下分析。

针对 PLCC 的攻击模型：发送方通过将要传输的数据拆分为不同大小的数据分组，假设来自网络 *A* 和网络 *B* 的敌手可以通过带外信息来合谋进行相关约定，例如约定当数据分组长度大于某一阈值时表示 1，否则表示 0，则可向收方传递一系列 01 编码的二进制比特串，从而利用数据分组大小隐蔽通道传递附加信息。

针对 SCC 的攻击模型：接收方通过收到作业分组后发送的状态反馈码，可直接控制发送端的重传次数、重传内容、重传的数据分组次序等，从而利用状态信息隐蔽通道传递附加信息。

### 3.3　方案构造

**定义 1**　源端网络 *A*

源端网络 *A* 向系统注入数据流为

$$I_A=\{X_{A,k}^{s_i}+X_{A,k}^{d}+X_{A,k}^{l}\mid k=1,2,\cdots;i=1,2,\cdots\}$$

其中，$k$ 表示数据分组序号，$X_{A,k}^{s_i}$ 表示源端网络 *A* 发出第 $k$ 个数据分组的控制信令集合，$s_i$ 是纠错重发状态信息的集合，$X_{A,k}^{d}$ 表示源端网络 *A* 发出第 $k$ 个数据分组的数据部分，$X_{A,k}^{l}$ 表示第 $k$ 个数据分组的长度。

**定义 2**　终端网络 *B*

终端网络 *B* 收到的数据流为

$$I_B=\{X_{B,m}^{s_i}+X_{B,m}^{d}+X_{B,m}^{l}\mid m=1,2,\cdots;i=1,2,\cdots\}$$

其中，$m$ 表示数据分组序号，$X_{B,m}^{s_i}$ 表示终端网络 *B* 收到的第 $m$ 个数据分组的控制信令集合，$s_i$ 是纠错重发状态信息的集合，$X_{B,m}^{d}$ 表示终端网络 *B* 发出第 $m$ 个数据分组的数据部分，$X_{B,m}^{l}$ 表示第 $m$ 个数据分组的长度。

**定义 3**　控制单元 *C*

*C* 指的是系统中隔离控制装置中的一系列控制单元，记为 $C=\{C_i|i=1,2,\cdots,p\}$。在控制单元 $C_i(i=1,2,\cdots,p-1)$ 上转发信息流时向下一级 $C_{i+1}(i=1,2,\cdots,p-1)$发送的格式为

$$I_{C_i}=\left\{X_{C_i,h}^{s_i}+X_{C_i,h}^{d}+X_{C_i,h}^{l}\mid h=1,2,\cdots;i=1,2,\cdots\right\}$$

其中，$h$ 表示数据分组序号，$X_{C_i,h}^{s_i}$ 表示控制单元 $C_i$ 发出第 $h$ 个数据分组的控制信令集合，$s_i$ 是纠错重发状态信息的集合，$X_{C_i,h}^{d}$ 表示控制单元 $C_i$ 发出第 $h$ 个数据分组的数据部分，$X_{C_i,h}^{l}$ 表示第 $h$ 个数据分组的长度。

控制单元 $C_1$ 收到的数据分组取决于与源端网络 *A* 之间通信时的协议类型。控制单元 $C_p$ 发送的数据分组取决于与终端网络 *B* 之间通信时的协议类型。

**假设 1**　在一个级联的通信系统中，通常假设各个级联模块共同组成一个马尔科夫链。因此在本系统中，存在 $I_A$、$I_{C_i}$ ($i$=1,2,⋯, $p$) 和 $I_B$，即 $P\left(I_B\mid I_A,I_{C_i}\right)=P\left(I_B\mid I_{C_i}\right)$。

**定理 1**　进入隔离控制装置的各数据分组长度的集合与离开该装置的各数据分组长度的集合之间不存在固定的映射关系时，可抵抗 PLCC 攻击。

**证明**　如图 2 所示，记来自源端网络 *A* 待传输的数据为 $D$，共拆分为 $n$ 个分组长为 $l_{D_i}$ ($i$=1,2,⋯,$n$) 的数据分组{$D_1$,$D_2$,⋯,$D_n$}，总长度为 $\sum_{i=1}^{n}l_{D_i}$。

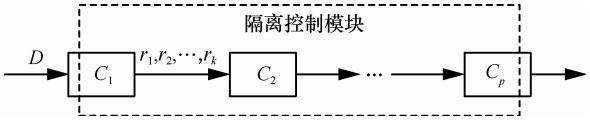


图 2　隔离控制模块

记其中最小的数据分组长度为 $l_{\min}$，最大的数据分组长度为 $l_{\max}$。对于拆分后的每一个数据分组 $D_i(i=1,2,\cdots,n)$，其长度可以有 $l_{\min}$,$l_{\min}$+1,1,$l_{\max}$ 共 $l_{\max}-l_{\min}$ 种可能，则发端 *A* 所能包含的最大 PLCC 信道容量为 $\log(l_{\max}-l_{\min})$。由于一次传输共 $n$ 个数据分组，则其总的最大信道容量为 $C_{\max}=n\log(l_{\max}-l_{\min})$。

这些数据分组进入控制单元 $C_1$，经重组后再重新拆分成 $k=\left\lceil\dfrac{\sum_{i=1}^{n}L_{D_i}}{t}\right\rceil$ 个长度均为 $t$ 的数据分组{$r_1$,⋯,$r_k$}。

以上数据分组经过控制单元 $C_1$，重新组分组后仅有 1 种固定长度为 $t$ 的分组，则此时的最大信道容量为 $C_{\max}=\log(1)=0$。对于一条级联信道来说，信道容量取决于所有部分中的最小值，所以 $A\rightarrow C_1\rightarrow\cdots\rightarrow C_p\rightarrow B$ 这条信道上的最大 PLCC 信道容量为 0，从而避免了分组长度类型的隐蔽通道的存在。

**定理 2** 由接收节点反馈回来的数据分组接收状态信息仅与当前发送节点有关，而与前一发送节点无关，且该状态信息不会传递至下一条节点，可抵抗 SCC 攻击。

**证明** 已知在本隔离装置中，网络 $A$ 与控制单元 $C_1$ 之间传输的数据依照网络 $A$ 的通信协议来保障正确性，确保正确后由接口 $D_1$ 接收，并拆分为作业分组添加校验码后存入 $C_1$。经由中间转发模块转发至 $C_2$，若 $C_2$ 校验未通过，则仅向 $C_1$ 请求重传，$C_1$ 知道自己存储的作业分组绝对正确，故而不会将这一重传请求继续向 $A$ 转发，这就保障了不会追溯至作业分组源处去请求正确的作业分组，确保状态信息不会跨级传递，从而实现了将状态信息隔离开来。

已知在本隔离装置中，仅当通过校验码及纠错码保证收到的数据分组正确无误后，才将其转发至下一跳。控制单元 $C_1$ 是否能正确接受来自网络 $A$ 的数据仅取决于 $A$ 与 $C_1$ 之间的信道；控制单元 $C_2$ 能否正确接收来自于控制单元 $C_1$ 只取决于 $C_2$ 与 $C_1$ 之间的信道，与 $A$ 与 $C_1$ 之间的信道无关。同理，$B$ 能否正确接收只取决于 $B$ 与 $C_p$ 之间的信道，与 $A$ 与 $C_1$ 之间、$C_i$ 与 $C_{i+1}$( $i$ =1,2,⋯,$p$−1)之间的信道均无关。因此 $A, C_1, C_2, \cdots, C_p, B$ 行成马尔科夫链，即 $B$ 与 $A$ 独立。

$C_1$ 对反馈给 $A$ 的接收状态与 $B$ 对 $A$ 的接收状态无关，即 $I(A;B)=H(A)-H(A|B)=0$。

通过状态来传输附加信息的该隐蔽通道的信道容量为 0。对于一条级联信道而言，信道容量取决于所有部分中的最小值，所以 $A\rightarrow C_1\rightarrow\cdots\rightarrow C_p\rightarrow B$ 这条信道上的最大状态信息隐蔽通道信道容量为 0，从而具备抵抗状态信息类型的隐蔽通道攻击的能力。

综上，针对第 2、3 类隐蔽通道，可通过确保控制单元间传递的数据的长度不存在映射关系，状态不存在传递关系，从而具备相应的抗隐蔽通道的功能。

**定理 3** 通过本隔离控制系统最少需要 3 个控制单元能够实现隐蔽通道。

**证明** 当 $p$=1 时，图 3 中存在 $A$、$C_1$ 和 $B$ 3 个实体。由于实体 $C_1$ 与网络 $A$、网络 $B$ 均连通，当有数据需要自网络 $A$ 传至网络 $B$ 时，相当于软件转发，在控制单元 $C_1$ 中既会有待返回给 $A$ 的状态信息，也会有来自 $B$ 的状态信息，一旦被来自 $A$、$B$ 网络任意一方的敌手攻破，状态信息会传递至另一网络，从而导致 SCC 的存在。

当 $p$=2 时，图 3 中则存在 $A$、$C_1$、$C_3$ 和 $B$ 4 个实体。已知 $A$ 与 $C_1$ 之间的协议遵从 $A$ 的通信协议，$C_3$ 与 $B$ 之间的协议遵从 $B$ 的通信协议，$C_1$ 与 $C_3$ 之间为结构化的私有协议。理论上 $C_1$ 与 $C_3$ 间不会有状态传递，但是 $C_1$ 会有来自网络 $A$ 的攻击，$C_3$ 会有来自网络 $B$ 攻击，故存在一定安全隐患。一旦 $C_1$、$C_3$ 中有任何一方被攻破，则同上述 $p$=1 时情况相同，将难以抵抗 SCC 隐蔽通道攻击。

当 $p$=3 时，图 3 中则存在 $A$、$C_1$、$C_2$、$C_3$ 和 $B$ 5 个实体。已知 $A$ 与 $C_1$ 之间的协议遵从 $A$ 的通信协议，$C_3$ 与 $B$ 之间的协议遵从 $B$ 的通信协议，$C_1$ 与 $C_2$、$C_2$ 与 $C_3$ 之间为私有协议。$C_1$ 会有来自网络 $A$ 的攻击威胁，$C_3$ 会有来自网络 $B$ 的攻击威胁，但 $C_2$ 理论上安全可靠，作为 $C_1$ 与 $C_3$ 之间的隔离，可以确保 $C_1$ 与 $C_3$ 状态不级联，由定理 2 知，可避免状态信息隐蔽通道的存在。

同理可知，当 $p\geqslant 3$ 时，均可具备抗隐蔽通道的功能。

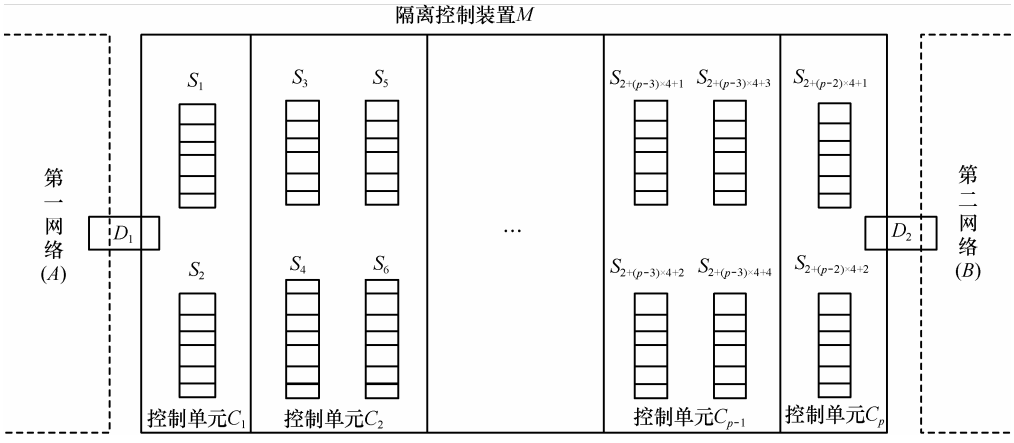


图 3 系统架构(含 $p$ 个控制单元时)

因此至少需要 3 个控制单元才能够实现抗隐蔽通道的功能。

## 4　NICS 方案具体实现

### 4.1　系统架构

NICS 系统架构如图 3 所示。以自网络 $A$ 至网络 $B$ 的上行数据传输过程为例对系统中各实体进行解释。

网络 $A$、网络 $B$：需要进行信息交互但又不可直接相连的 2 个网络。

隔离控制装置 $M$：由 $D_1$、$D_2$ 2 个接口模块，$p$ 个控制单元 $C_i(i=1,2,\cdots,p)$组成，作业分组只能在被控制单元正确接收后，再在相邻的 $C_i$ 和 $C_{i+1}(i=1,2,\cdots,p-1)$之间接力传递，任意非相邻的 2 个控制单元之间不能进行直接通信，控制单元 $C_i$ 包括中间转发模块、上行输入/输出缓冲区及下行输入/输出缓冲区。

接口模块 $D_1$：用于当收到网络 $A$ 发送给网络 $B$ 的上行通信数据时，将该上行通信数据拆分为多个数据分组，对于拆分得到的各数据分组，分别根据该作业分组中预定位置的多个字节计算校验值，将计算结果作为校验码加入该数据分组尾部，并对每个数据分组中的有效数据段计算其纠错码，将纠错码加入该数据分组尾部，将组装好的作业分组发送给控制单元 $C_1$ 的上行数据缓冲区；或用于收到控制单元 $C_1$ 的中间转发模块转发来的作业分组后，提取有效数据重组成下行通信数据，发送至网络 $A$。

接口模块 $D_2$：用于当收到网络 $B$ 发送给网络 $A$ 的下行通信数据时，将该下行通信数据拆分为多个作业分组，对于拆分得到的各作业分组，分别根据该作业分组中预定位置的多个字节计算校验值，将计算结果作为校验码加入该作业分组，并对每个作业分组中的数据段计算其纠错码，将纠错码加入该作业分组，将作业分组发送给控制单元 $C_p$ 的下行数据缓冲区；或用于收到控制单元 $C_p$ 的中间转发模块转发来的作业分组后，提取有效数据重组成上行通信数据，发送至网络 $B$。

中间转发模块：当控制单元 $C_i$ 的上行数据缓冲区里有作业分组时，$i$ 不为 $p$ 时将该作业分组转发给控制单元 $C_{i+1}$ 的上行数据缓冲区，$i$ 为 $p$ 时将该作业分组转发给所述第二接口 $D_2$；当本控制单元 $C_i$ 的下行数据缓冲区里有作业分组时，$i$ 不为 1 时将该作业分组转发给控制单元 $C_{i-1}$ 的下行数据缓冲区，当 $i$ 为 1 时将该作业分组转发给所述第一接口 $D_1$。

上行输入缓冲区：用于存储由中间转发模块转发的来自第一接口 $D_1$ 或来自控制单元 $C_{i-1}$(当 $i=2$ 时)的上行输入缓冲区的作业分组，或来自控制单元 $C_{i-1}$(当 $i=3,4,\cdots,p$ 时)的上行输出缓冲区的作业分组。

上行输出缓冲区：用于存储经由中间转发模块转发的来自本控制单元(当 $i\neq p$)中上行输入缓冲区中存储的作业分组、或来自控制单元 $C_{i-1}$(当 $i=p$ 时)的上行输出缓冲区的作业分组。

下行输入缓冲区：用于存储由中间转发模块转发的来自第二接口 $D_2$ 或来自控制单元 $C_{i+1}$(当 $i=p-1$ 时)的下行输入缓冲区的作业分组、或来自控制单元 $C_{i+1}$(当 $i=1,2,\cdots,p-2$ 时)的下行输出缓冲区的作业分组。

下行输出缓冲区：用于存储由中间转发模块转发的来自本控制单元(当 $i\neq 1$)中下行输入缓冲区中存储的作业分组、或来自控制单元 $C_{i+1}$(当 $i=1$ 时)的下行输出缓冲区的作业分组。

### 4.2　通信流程

在本示例中，设定上述理论模型中的 $p=3$，如图 4 和图 5 所示。上行数据缓冲区 $S_1$、$S_3$、$S_6$、$S_8$ 与下行数据缓冲区 $S_7$、$S_5$、$S_4$、$S_2$ 各为一组，分别处理各自流向的作业分组。这 2 条完全隔离的读写通道，可同时工作，互不干扰，将 $A$、$B$ 2 个网络在硬件上完全隔离开。

当来自网络 $A$ 的数据要经由隔离装置 $M$ 进入网络 $B$ 时，如图 4 所示，具体步骤如下（记文中所有长度单位均为字节）。

**Step1**　接口 $D_1$ 接收到来自第一网络 $A$ 的输入数据流，拆分为固定长度的作业分组并存入上行输入缓冲区 $S_1$。

**Step1-1**　接口 $D_1$ 对接收到的数据依据网络 $A$ 遵循的通信协议进行校验。记该数据为 $R$，其总长度为 $L_R$。

**Step1-2**　若校验未通过，则依据该通信协议请求重传。

**Step1-3**　若校验通过，则将校验通过的数据拆分为固定长度为 $l_s$ 的 $n$ 个作业分组$(r_1,r_2,\cdots,r_n)$。

其中，$n=\lceil L_R/l_s\rceil$。

由于最后一分组数据有可能小于 $l_s$，为保证固定数据分组长，须进行填充。

假设最后一个分组数据长度为 $l_{\text{last}}$，即

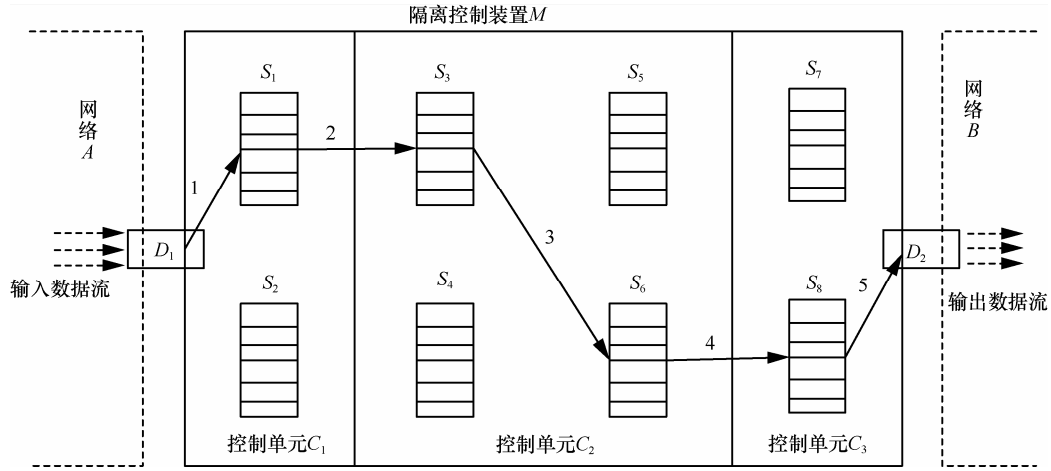


图 4 隔离控制装置内部上行数据处理过程

$$l_{\text{last}} = L_R \bmod (l_s)$$

需对最后一个分组填充 $l_s - l_{\text{last}}$ 个字节。填充值为 $l_s - l_{\text{last}}$，表示填充的数据长度，便于当接收到作业分组提取数据时，可直接根据最后一个分组的内容截掉相应长度的填充字节。

**Step1-4** 为填充后的各等长数据分组添加分组头信息，包括作业分组序号 *ID*(4 byte)、作业分组有效数据长度 *Length*(4 byte)、状态标志(1 byte，1 表示当前作业分组后续还有作业分组，0 表示当前作业分组为本次拆分的最后一个分组作业)。构成的作业分组数据 *pkt* 为

$$ID \,|\, Length \,|\, state \,|\, sockID \,|\, data$$

**Step1-5** 对上述添加分组头信息后的作业分组数据计算纠错码，添加至作业分组尾部。构成的作业分组数据为

$$ID \,|\, Length \,|\, state \,|\, sockID \,|\, data \,|\, ECC\left(pkt\right)$$

对上述添加分组头信息及纠错码后的作业分组数据计算校验码，添加至作业分组尾部。最终构成的作业分组数据为

$$ID \,|\, Length \,|\, state \,|\, sockID \,|\, data \,|\, ECC\left(pkt\right) \,|\, CRC\left(pkt \,|\, ECC\left(pkt\right)\right)$$

最后将组合完毕的上述作业分组存入控制单元 $C_1$ 的上行输入数据缓冲区 $S_1$。

**Step2** 控制单元 $C_1$ 将上行输入数据缓冲区 $S_1$ 中存储的作业分组转发至控制单元 $C_2$ 的上行输入数据缓冲区 $S_3$。

**Step2-1** $C_1$ 先从要进行转发的作业分组中提取固定长度(5+*Length*+纠错码长度)字节的数据。计算其校验码。将计算出的校验码与作业分组尾部的校验码对比，如果相同，则校验通过，将该作业分组直接存入控制单元 $C_2$ 的上行输入缓冲区 $S_3$。否则，校验不通过。

**Step2-2** 当校验不通过时，对从作业分组中提取的 5+*Length* 字节的数据进行纠错，若在纠错能力范围内，则计算出正确的原始作业分组数据，并将其存入控制单元 $C_2$ 的上行输入缓冲区 $S_3$。否则，请求重传当前数据分组。并重复 Step2，直至将正确的当前序号作业分组存入控制单元 $C_2$ 的上行输入缓冲区 $S_3$ 为止。

**Step3** $C_2$ 将 $S_3$ 中存储的作业分组转发至 $S_6$。同 Step2。

**Step4** $C_2$ 将 $S_6$ 中存储的作业分组转发至 $S_8$。同 Step2。

**Step5** $C_3$ 将 $S_8$ 中存储的作业分组转发至接口 $D_2$。$D_2$ 经同样的步骤确保收到的作业分组正确后，依照作业分组头部的作业分组序号 *ID* 及作业分组状态标识，将去掉分组头控制信息和分组尾校验信息的数据部分遵从第二网络 *B* 的通信协议加入必要的控制信息，重组恢复出进入隔离控制装置 *M* 前的符合相关协议的原始数据，并经由第二接口 $D_2$ 通过相应的通信协议机制转发至第二网络 *B*。

下面采用 Z-符号[23]对数据进入隔离控制装置后的拆包操作；为每个定长数据分组 *r* 添加分组头操作；为每个定长数据分组 *r* 尾部添加纠错校验信息操作；对收到的作业分组 *r* 进行校验纠错操作以及删除原添加的分组头及分组尾信息操作等进行形式化描述。其中, INSULATION 是一个抽象数据类型，表示理论模型中的数据分组、作业分组、隔

离分组等组件；DATAS 是通过校验的来自源网络的数据流集合，PACKETR 为定长数据分组集合，HEADER 是数据分组分组头，PACKETH 为添加了分组头的数据分组，PACKETW 是作业分组集合。

对数据处理的相关操作进行如下形式化描述（数据处理相关函数如表 1 所示）。

*Divide*(*data:INSULATE*) ◁
*data*∉ *DATA*
if *verify*(*data*)=*true* then *DATA*'=*DATA* ∪ {*data*}
$L_{data}$= *length*(*data*)
*n*=*split*(*data*,$L_{data}$,$L_s$)
if (*length*($r_n$)≠*length*($r_1$))
then *fill*($r_n$(*xxxx*)| *length*($r_n$)=$L_s$)
if *data*.$r_1$∉ *PACKETR* then
*PACKETR*'=*PACKETR* ∪ {$r_1$}
…
if *data*.$r_n$∉ *PACKETR* then
*PACKETR*'=*PACKETR* ∪ {$r_n$}
*PACKETR*'=*PACKETR* ∪ {*data*} ▷
*Addheader*(*packetr:INSULATE*) ◁
(*packetr*∈ *PACKETR*)∧(*packetr*∉ *PACKETH*)
if (*length*($r_1$)= *length*($r_2$)=…= *length*($r_n$))
if (*creath*($r_1$,*sockID*,*state*,*length*($r_1$),*packetID*)∧(*header*.$r_1$∉ *HEADER*))
*then* *HEADER*'=*HEADER* ∪ {*header*.$r_1$}
…
if(*creath*($r_n$, *sockID*, *state*, *length*($r_n$), *packetID*)∧(*header*.$r_n$∉ *HEADER*)
then *HEADER*'=*HEADER* ∪ {*header*.$r_n$}
*HEADER*'=*HEADER* ∪ {*header*}
if (*packetr*.$r_1$ ∈ *PACKETR*∧*packetr*.$r_1$ ∉ *PACKETW*∧*header*.$r_1$∈ *HEADER*)
then $r_1$'=*addh*($r_1$,*header*.$r_1$)
…
if (*packetr*.$r_n$ ∈ *PACKETR*∧*packetr*.$r_n$ ∉ *PACKETW*∧*header*.$r_n$∈ *HEADER*)
then $r_n$'=*addh*($r_n$,*header*.$r_n$)
if *packetr*.$r_1$∉ *PACKETH*
then *PACKETH*'= *PACKETH* ∪ {*packetr*.$r_1$}
…
if *packetr*.$r_n$∉ *PACKETH*
then *PACKETH*'= *PACKETH* ∪ {*packetr*.$r_n$}
*PACKETH*'= *PACKETH* ∪ {*packetr*} ▷
*Addtail*(*packeth:INSULATE*) ◁
(*packeth*∈ *PACKETH*)∧(*packeth*∉ *PACKETW*)
*packeth*.$r$1=*packeth*.$r$1||*CRC*(*packeth*.$r_1$)||*ECC*(*packeth*.$r_1$||*CRC*(*packeth*.$r_1$))
if *packeth*.$r_1$∉ *PACKETW*
then *PACKETW*'=*PACKETW* ∪ {*packeth*.$r_1$}
…
*packeth*.$r_n$ = *packeth*.$r_n$||*CRC*(*packeth*.$r_n$)||*ECC*(*packeth*.$r_n$)
if *packeth*.$r_n$∉ *PACKETW*
then *PACKETW*'=*PACKETW* ∪ {*packeth*.$r_n$}
*PACKETW*'= *PACKETW* ∪ {*packeth*} ▷
*check*(*packetw:INSULATE*) ◁
*packetw*∈ *PACKETW*
if (*verify*(*r*) *or correct*(*r*)) *then store r into* $S_3$
else *ask resend r* ▷
*delete*(*packetw: INSULATE*) ◁
if(*sort*($r_1$,$r_2$,…,$r_n$)) then $r_1$'=*delete*($r_1$)
…
$r_n$'=*delete*($r_n$)
if(*defill*($r_1$,*length* $_{valid}$($r_1$)))
*if*(*data*.$r_1$=$r_1$'∧…∧*data*.$r_n$=$r_n$')
then $length_{total}$=*combine*($r_1$',$r_2$', …,$r_n$')
if *data*∉ *DATAOUT*
then *DATAOUT*=*DATAOUT* ∪ {*data*}

## 5　安全特性比较

与 NICS 相近的网络隔离机制的相关特性对比分析如表 2 所示[17,24,25]。

完全隔离即传统的物理隔离技术[17,24]，消除了内部网络受到来自外部网络安全威胁的可能，保证了高安全特性，同时也使信息交互只能人工摆渡或借助光盘、磁盘等存储介质实现不能满足信息交互实时性的需求；没有完善的校验机制，不具备数据传输高可靠性。但同时也避免了状态信息反馈码的产生，具备抗 SCC 的功能。虽然每次数据分组长度的不同也可传递一定的附加信息，但由于其传递次数少、每次传递时的信道容量小，带来的安全威胁可忽略不计，故可视为具备抗 PLCC 的功能。

硬盘隔离技术将终端的存储硬盘划分为相互隔离的内网数据区和外网数据区，在软件层逻辑隔离硬盘数据区。2 套系统公用内存，不能保证数据

**表 1　　数据处理相关函数**

| 函数名 | 描述 |
|---|---|
| verifyp | 若收到依据源网络遵循的通讯协议的数据，则校验通过返回 True，否则返回 False<br>*verifyp*(*data*, *protocol:INSULATE*) ◁<br>*Result*=(*data* ∈*DATA*)∧(*isvalid*(*protocol*)) ▷ |
| split | 将进入隔离控制装置的的数据拆分为固定长度是 $L_s$ 的作业分组，返回拆分后的作业分组个数 *N*<br>*split*(*data*,$L_{data}$,$L_s$:*INSULATE*) ◁<br>*result*=$L_{data}/L_s$ ▷ |
| fill | 若数据分组长度小于固定长度，则填充，填充后固定长度返回 True，否则返回 False<br>*fill*($r_n$,$L_s$:*INSULATE*) ◁<br>*result*=(*length*($r_1$)=*length*($r_n$)) ▷ |
| creath | 为第 *i* 个数据分组构造分组头，若分组头属于 HEADER 则返回 True，否则返回 False<br>*creath*(*packetID*,*length*($r_i$), *state*,*sockID:INSULATE*) ◁<br>$r_i$ ∈*PACKETR*<br>*list*1=<*packetID*><br>*list*2=<*length*($r_i$)><br>*list*3=*state*<br>*list*4=*sockID*<br>$header_i$= *list*1^*list*2^*list*3^*list*4<br>*result*=($header_i$ ∈*HEADER*) ▷ |
| addh | 为第 *i* 个数据分组添加已构造好的分组头，并返回总长度 *N*<br>*addh*($r_i$,$header_i$:*INSULATE*) ◁<br>$r_i$ ∈*PACKETR*<br>*list*1=$header_i$<br>*list*2=$r_i$<br>*result*= *list*1^*list*2 ▷ |
| verifyw | 对收到的作业分组 *r* 进行校验，通过返回 True，否则返回 False<br>verifyw(r:INSULATE) ◁<br>result(r∈PACKETW∧isright(CRC(*r*))) ▷ |
| correctw | 对收到的作业分组 *r* 进行纠错，得出正确数据返回 True，否则返回 False<br>correctw (*r*:INSULATE) ◁<br>result(r∈PACKETW∧isright(ECC(*r*))) ▷ |
| sort | 将收到的所有作业分组按照分组头中的作业分组序号进行排序，排序完成返回 True，否则返回 false<br>sort($r_1$,$r_2$,⋯,$r_n$:INSULATE) ◁<br>result=(issorted($r_1$, $r_2$,⋯,$r_n$)) ▷ |
| delhdr | 将作业分组中的分组头控制信息及尾部的校验码、纠错码等信息去掉，取出其中的有效数据载荷部分，返回有效数据长度 *N*<br>delete(r:INSULATE) ◁<br>result=($length_{valid}$ (*r*)) ▷ |
| defill | 若数据分组有效载荷长度小于固定长度，则去掉填充，去填充后长度为有效数据载荷部分长度则返回 True，否则返回 False<br>defill($r_n$,$llength_{valid}$($r_n$) :INSULATE) ◁<br>result=($length_{valid}$ ($r_n$)=$length_{off}$($r_n$)) ▷ |
| combine | 将已经去掉分组头及尾部的已排好序的有效数据载荷部分组合，返回组合后的总数据长度 *N*<br>combine($r_1$,$r_2$,⋯,$r_n$:INSULATE) ◁<br>*list*1=$r_1$<br>*list*2=$r_2$<br>⋯<br>*listn*=$r_n$<br>*data*= *list*1^*list*2^⋯^*listn*<br>result=#data ▷ |

**表 2　　相关隔离机制安全特性比较**

| 隔离机制 | 相关性能 | | | | |
|---|---|---|---|---|---|
| | 高安全性 | 高实时性 | 数据传输高可靠性 | 抗 PLCC | 抗 SCC |
| 完全隔离 | √ | × | × | √ | √ |
| 硬件卡隔离 | × | × | × | √ | √ |
| 数据转播隔离 | × | × | × | × | × |
| 空气开关隔离 | × | × | × | √ | × |
| 安全通道隔离 | √ | √ | √ | × | × |
| NICS | √ | √ | √ | √ | √ |

传输高可靠性。分隔的 2 个分区属于不同的系统，终端每次进入系统时需重新选择，待交换的数据在硬盘的分区间拷贝，不能保证数据传输的高实时性。但数据分组长的差异在存入存储空间时已经消除，不再隐含附加信息，故具备抗 PLCC 的功能。数据交互时不产生状态信息反馈码，故具备抗 SCC 的功能。

数据转播隔离技术利用转播系统分时复制文件的途径来实现隔离，难以控制恶意数据的传播，且切换时间久，不具备高安全、高实时和高可靠性。单一的复制途径，未对数据分组长度进行处理，难以阻止凭借数分组长度传递的附加信息，故不具备抗 PLCC 的功能。原有通信协议已有的校验等措施产生的状态信息反馈码，可被利用传递附加信息，故不具备抗 SCC 的功能。

空气开关隔离通过使用单刀双掷开关，任意时刻仅与其中一方网络连通，难以阻止恶意数据的传递，不能达到高安全性的需求。内外部网络分时访问临时缓存器来完成数据交换，不能保证高实时性。未采用校验等措施保障数据传输的高可靠性，但却避免了状态信息反馈码的产生，具备了抗 SCC 的功能。由于未对数据分组长度进行控制，不具备抗 PLCC 的功能。

隔离网闸[24,25]由内网处理单元、外网处理单元和专用隔离硬件交换单元 3 部分组成，从物理上隔离、阻断了具有潜在攻击可能的一切连接，保证了数据传输的高安全性和数据交互的高实时性。基于 MD5 算法的数据签名和基于 RSA 算法的编解码过程[24]保证了数据传输的高可靠性。将数据分组重组为静态数据[24]，切断了网络之间的通用协议连接，但并未消除重组前后数据分组长度之间的固定映射关系，不具备抗 PLCC 的功能。不能阻断接收时产生的状态信息的反馈回传，不具备抗 SCC 的功能。

NICS 可从物理上隔离、阻断所有潜在攻击可能的连接，具有高安全性；2 条物理上完全独立的单向传输路线，保证了数据传输的高实时性；为进入隔离控制单元的数据重新分组装填充为固定长度的数据分组，且为每一个分组添加校验纠错码，每一阶段的校验措施保证了数据传输高可靠性；固定的数据分组长度，打破了与原有数据分组长度之间的固定映射关系，具备抗 PLCC 的功能；每次都校验数据正确无误后才接收存储，请求重发时无需追溯至信源处获取正确作业分组，从而避免了状态信息反馈码的传递，具备抗 SCC 的功能。

## 6　结束语

在任意的通信系统中，本质上都存在隐蔽通道，传统的物理隔离人工摆渡的方式并没有消除隐蔽通道，只是由于交互次数少且每次信道容量小，故通过隐蔽通道传递的信息量可忽略不计，进而不会形成具有潜在安全威胁的攻击行为。

本文针对网络隔离通信方案中存在的 PLCC 和 SCC 问题提出抗隐蔽通道特性的新型的网络隔离通信方案 NICS，实现了不同网络之间的实时信息安全交互。在理论模型上证明了 NICS 具有抗上述 2 类隐蔽通道的特性，安全性分析得出在交互相同信息量的前提下，可实现与物理隔离等价的抗隐蔽通道的安全效果。本方案已在“某边界安全网关”项目中实现并验证可行。

## 参考文献：

[1] LIU J Y, FANG Y J,ZHANG D H. PROFIBUS-DP and HART protocol conversion and the gateway development[A]. Proceeding of 2nd IEEE Conference on Industrial Electronics and Applications(ICIEA)[C]. Harbin, China. 2007, 15-20.

[2] DONG G S, LIU ZH J, ZHAO D. A security domain isolation and data exchange system based on VMM[A]. Proceeding of 3rd International Conference on Signal Processing and Communication Systems

(ICSPCS)[C].Omaha, NE,USA. 2009.1-5.

[3] DU J, LIU P P. Design and implementation of efficient one-way isolation system based on PF_RING[A].Proceeding of 2012 Fourth International Conference on Multimedia Information Networking and Security(MINES)[C]. Nanjing, China, 2012.105-108.

[4] LAMPSON B W. A note on the confinement problem[J]. Communications of the ACM. 1973, 16(10):613-615.

[5] National Computer Security Center, DoD, Trusted Computer System Evaluation Criteria[R]. National Computer Security Center, Washington, DC, USA, 1985.

[6] ZHAI G S, ZHANG Y F, LIU C Y, *et al*. Automatic identification of covert channels inside linux kernel based on source codes[A]. Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human (ICIS'09)[C]. Seoul, Korea, 2009.440-445.

[7] MOSKOWITZ S I, NEWMAN R E, CREPEAU P D, *et al*. Covert channels and anonym zing networks[A]. Proceedings of the 2003 ACM Workshop on Privacy in the Electronic Society[C]. New York, NY, USA. 2013.79-88.

[8] WANG Y, FERRAIUOLO A, SUH G E. Timing channel protection for a shared memory controller[A]. Proceeding of 2014 IEEE 20th International Symposium on High Performance Computer Architecture (HPCA)[C].Orlando, FL,USA,2014.225-236.

[9] JI L P, JIANG W H, DAI B Y. A novel covert channel based on length of messages[A]. Proceedings of International Symposium on Information Engineering and Electronic Commerce (IEEC '09)[C]. Ternopil, Ukraine, 2009.551-554.

[10] LI S, EPHREMIDES A. A covert channel in mac protocols based on splitting algorithms[A]. Proceeding of Wireless Communications and Networking Conference[C]. New Orleans, LA, USA, 2005.1168-1173.

[11] QU H P, SU P R, FENG D G. A typical noisy covert channel in the IP protocol[A]. Proceeding of 38th Annual 2004 International Carnahan Conference on Security Technology[C].Albuquerque, NM, USA, 2004. 189-192.

[12] WANG Z H, LEE R B. Covert and side channels due to processor architecture[A]. Proceedings of 22nd Annual Computer Security Applications Conference (ACSAC '06)[C].Miami Beach, FL,USA. 2006.293- 302.

[13] M. A. PADLIPSKY M A, SNOW D W, KARGER P A. Limitations of End-to-End Encryption in Secure Computer Networks[R]. ESD-TR-78-158, Mitre Corporation, August 1978.

[14] GENNUSO K. Disconnect from the Internet - Whale's e-Gap In-Depth. GSEC Practical Assignment Version 1.2f[S].2001.

[15] LINDSKOG S, GRINNEMO K J, BRUNSTROM A. Data protection based on physical separation: concepts and application scenarios[A]. Proceedings of the 2005 International Conference on Computational Science and Its Applications(ICCSA'05)[C]. Singapore. 2005.1331-1340.

[16] WU H Y, TAN C X, WANG H H. Building a high-performance communication framework for network isolation system[A]. Proceedings of IEEE International Conference on Networking, Sensing and Control(ICNSC)[C]. Sanya, China, 2008.1086-1091.

[17] 方勇, 刘嘉勇. 信息系统安全导论[M]. 北京: 电子工业出版社, 2003.
FNAG Y, LIU J Y. Introduction to Information System Security[M]. Beijing: Publishing House of Electronics Industry, 2003.

[18] YU S S, PENG Y, ZHAN Y J, *et al*. Session mechanism research based on agent in NetGAP[A]. Proceedings of International Symposium on Intelligent Information Technology Application Workshops(IITAW'08)[C]. Shanghai, China, 2008.395-398

[19] 章思宇, 邹福泰, 王鲁华等. 基于 DNS 的隐蔽通道流量检测[J]. 通信学报, 2013, 34(5):143-151.
ZHANG S Y, ZOU F T, WANG N H, *et al*. Detecting DNS-based covert channel on live traffic[J]. Journal on Communications,2013, 34(5): 143-151.

[20] GIRLING C. Covert channels in LAN's[J]. IEEE Transactions on Software Engineering, 1987, 13(2):292-296.

[21] 钱玉文, 李勇, 王执铨. 网络包长度隐蔽信道的建模与仿真[J].系统仿真学报, 2010, 22(7):1773-1781.
QIAN Y W, LI Y, WANG Z Q. Modeling and simulation of covert channel based on network packet length[J]. Journal of System Simulation, 2010, 22(7):1773-1781.

[22] OMAR S N, AHMEDY I, NGADI M A. Indirect DNS covert channel based on name reference for minima length distribution[A]. Proceedings of International Conference on Information Technology and Multimedia (ICIM)[C]. Kuala Lumpur, Malaysia. 2011.1-6.

[23] SPIVEY J M. The Z Notation: A Reference Manual. International Series in Computer Science[M]. Prentice-Hall, New York, NY, USA. 1992.

[24] WANG P, LIU S S, YAN L X, *et al*. Net gap among different security level networks[A]. Proceedings of 2012 Fourth International Conference on Multimedia Information Networking and Security (MINES)[C]. Nanjing, China, 2012. 4-7.

[25] YU S S, ZHAN Y J, CAI Q L, *et al*. A Reflective NetGAP logic framework design[A]. Proceeding of Workshop on Power Electronics and Intelligent Transportation System(PEITS '08)[C]. 2008.565-568.

**作者简介：**

**李凤华**（1966-），男，湖北浠水人，中国科学院信息工程研究所副总工、研究员、博士生导师，主要研究方向为网络与系统安全、可信计算。

**谈苗苗**（1990-），女，甘肃陇南人，西安电子科技大学硕士生，主要研究方向为网络安全。

**樊凯**（1978-），男，陕西西安人，西安电子科技大学副教授、硕士生导师，主要研究方向为无线网络安全、移动社交网络及移动医疗网络中的隐私保护等。

**耿魁**（1989-），男，湖北红安人，西安电子科技大学博士生，主要研究方向为网络安全。

**赵甫**（1978-），男，河北元氏人，北京航天数控系统有限公司副总经理，主要研究方向为信息安全技术、伺服控制技术等。