代　　号 10701　　学　　号 1101120424

分类号 TP309.2　　密　　级 公开

题（中、英文）目 基于CSP的网络隐蔽信道检测和分析技术研究

CSP-based Detection and Analysis of Network Covert Channels

作 者 姓 名 刘婷婷　　指导教师姓名、职务 朱辉 副教授

学 科 门 类 工学　　学科、专业 信息安全

提交论文日期 二〇一四年三月

万方数据

西安电子科技大学

学位论文独创性（或创新性）声明

本人声明所呈交的论文是我个人在导师的指导下进行的研究工作及所取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果；也不包含为获得西安电子科技大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志所做的任何贡献均已在论文中做了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切相关责任。

本人签名：_____ 日期：_____

西安电子科技大学

关于论文使用授权的说明

本人完全了解西安电子科技大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属西安电子科技大学。本人保证毕业离校后，发表论文或使用论文工作成果时署名单位仍然为西安电子科技大学。学校有权保留送交论文的复印件，允许查阅和借阅论文；学校可以公布论文的全部或部分内容，可以允许采用影印、缩印或其他复制手段保存论文。（保密的论文在解密后遵守此规定）

本学位论文属于保密，在___年解密后适用本授权书。

本人签名：_____ 日期：_____

导师签名：_____ 日期：_____

万方数据

摘 要

计算机网络技术是一把双刃剑，在给人们生活带来极大便利的同时，也带来了更多的网络安全问题。网络隐蔽信道是网络环境中一种泄露信息的入侵方式，它利用网络协议作为载体，通过把机密信息嵌入到网络协议首部特殊字段进行传输，对互联网造成极大的安全威胁。因此，分析网络协议和网络隐蔽信道的特点，研究网络隐蔽信道的检测技术成为当前安全研究与应用领域的热点问题。

目前，形式化方法和自动化验证技术在网络协议安全分析领域得到广泛应用。本文通过对网络协议和网络攻击者形式化描述，针对网络存储隐蔽信道的构建特性，提出一种基于 CSP 的形式化检测和分析模型。

本文首先分析网络存储隐蔽信道的特点，为网络协议首部定义属性和隐蔽漏洞的概念。并根据属性定义，把网络协议首部划分为三种类型。然后提出一种基于 CSP 形式语言的网络存储隐蔽信道检测和分析通用模型，该模型对网络协议和攻击者进行语义描述，定义网络协议交互系统和网络存储隐蔽信道中的事件与基本进程，以及所要满足的安全约束。在网络协议的语义正确性的基础上，利用 CSP 中迹模型的提炼定义以及 FDR 检测工具，通过检测进程之间的提炼关系来对网络存储隐蔽信道进行检测和分析。

最后，以 TCP 协议为例对模型进行建模验证，验证结果为检测到违反安全约束的反例序列，每条反例序列对应一个网络存储隐蔽信道攻击。该验证证明了 CSP 形式化模型的有效性和可用性。

关键词：网络存储隐蔽信道　通信顺序进程　网络协议 安全建模

万方数据

Abstract

Computer network technology is a “double-edged sword”, when it provides a great convenience for people’s life, it also has brought more network security issues. Network storage covert channel is a kind of invasion leaking information, utilizing network protocol as the carrier, embedding the secret information into the special header fields of network protocols, posing a serious security threat to the Internet. Therefore, the analysis of the characteristics of network protocols and network storage covert channel and the research of the detection technology of network storage covert channel has been the current focus in the domain of security research and application.

After the study on the problems of formal specification for network protocols and network attackers, this paper gives out a formal detection model in view of the network storage covert channels. Concrete work done as follows:

Firstly, we research on the characteristics of the network storage covert channel, defining the concept of attributes and covert vulnerabilities. And we classify the network protocol header fields into three categories according to the attribute definition. Secondly, put forwards a network storage covert channel detection and analysis method based on CSP formal language, which gives the semantic description of network protocols and the attackers, defining the events and basic process in the network protocol interaction system and network storage covert channels and the security constraints they needed to satisfy. On the basis of the semantic correctness of network protocols, we detect and analyses the network storage covert channel using the refinement definition of trace in the CSP model and FDR detection tools by testing the refinement relationship between processes.

Finally, the model has been tested in TCP as an example. The verification gets the counter-example sequences which violate security constraints, and each counter-example sequence corresponds to a network covert channel. The test proves the validity and practicability of the CSP formal model.

Keywords: Network Storage Covert Channel CSP Network Protocol Security Model

万方数据

目 录

万方数据

万方数据

第一章 绪论

1.1 研究背景及其目的和意义

随着信息化技术的发展，互联网技术在普通大众生活中越来越普及，计算机网络已经深入到人们的工作和生活中的各个方面，发挥着越来越重要的作用。据一项互联网络发展状况统计报告显示，截止到 2013 年上半年底，我国网民规模达到5.91亿，而手机网民规模也已达到 4.64 亿，然而具有根本安全意识的网民却少于 40%。2012 年国内网购市场规模达到 12，000 亿元，2013 年 11 月 11 日，光棍节一天的网络购物总额已超出 350 亿元。网络中存在着巨大的利益导致域名劫持、网页篡改、网络黑客等事件越来越多。在最近几年中，中国有几次主要的公共信息泄漏，例如：自 2011 以来，淘宝、京东等购物网站用户帐号被盗；脸谱网、人人网等社交网络出现漏洞，600 万用户账户的个人信息外泄；2013 年，中国人寿个人信息泄露，包含 80 万份保单。计算机网络在改变人们劳动与生活方式的同时，也面临十分严峻的网络安全形势考验，保护网络信息的安全已经是当务之急。

网络隐蔽信道是指把窃取的非法授权的秘密消息隐藏在正常的网络传输协议中的一种通讯机制。网络隐蔽信道能够带来系统间的信息泄露，可以被攻击者用于建立一种违反系统安全策略的通信机制实现对隐蔽信息的传递。网络隐蔽信道能够逃避防火墙和入侵检测系统，且对多级安全系统是一个很大的安全威胁。因此，越来越多的研究人员开始研究构建和检测网络隐蔽信道。网络隐蔽信道作为一种信息泄露方式，它能够逃避防火墙和入侵检测系统，通常被黑客用来窃取未被授权的数据，造成严重的信息泄露事故，对网络安全和信息安全构成了巨大的威胁。

网络隐蔽信道是一把双刃剑。一方面，在加强网络隐私[1]、水印加密[2]、跟踪VoIP电话[3]、方便获取系统记录数据[4]等几个方面具有研究意义和应用前景。另一方面，网络隐蔽信道也被许多研究学者和黑客证明具有秘密窃取机密消息的能力，为互联网中的DDos攻击[5]和网络蠕虫攻击[6]，以及物理攻击方案[7]和其他颠覆性的攻击提供了方法和机会，对计算机网络安全构成很大的威胁，造成严重的信息泄露事故，对网络安全和信息安全构成了巨大的威胁。

网络隐蔽信道作为一种秘密传输机制，能够逃避防火墙和入侵检测体系，受到黑客的青睐，被用来盗取未被授权的数据，造成严重的信息泄露事件，对网络安全和信息安全构成了巨大的威胁。因此，对于网络隐蔽信道的研究已经成为目前网络安全领域的热门课题。

万方数据

目前，国内以及国际相关安全评价准则，都明确要求高安全等级系统务必对隐蔽信道进行检测和分析。例如，国内的《信息技术安全技术信息技术安全性评估准则》（GB/T18336.1—2001）[8]和美国的《可信计算机系统评估准则》（TCSEC）[9]都对隐蔽信道有明确分析。此外，由美国、英国、德国等 9 个国家制订的《信息技术安全性评估通用准则》（简称CC标准）[10]明确规定，在对EAL-5 级和更高安全等级信息系统进行评估时，必须分析隐蔽信道，以确保系统能够正确地执行其安全策略。

本文着眼于存储式网络隐蔽信道，提出一种基于 CSP 的形式化模型，用于检测和分析存储式网络隐蔽信道，为网络隐蔽信道的更进一步研究提供参考。

1.2 国内外研究现状

网络隐蔽信道是一种恶意通信机制，它可以被攻击者通过建立一种违反系统安全策略的通信机制实现对隐蔽信息的传递。网络隐蔽信道通常很难检测，且对多级安全系统是一个很大的安全威胁。因此，越来越多的研究专家们开始研究网络隐蔽信道的构造和检测技术。

网络隐蔽信道分为网络存储隐蔽信道和网络时间隐蔽信道。网络存储隐蔽信道主要使用网络协议首部中的未使用字段、填充位、扩展位等。国外的很多研究专家提出了基于不同网络协议的隐蔽信道构建方法。例如，Kundur[11]和Handel[12]分别提出基于IP协议的网络隐蔽信道。其中，前者利用IP首部分片标志字段（即DF字段）在知道网络中的MTU(即最大传输单元)的条件下可以任意赋值的性质构建隐蔽信道；后者则使用IPv4 协议首部的服务类型字段（即TOS字段）中未利用的字节构造隐蔽信道。Fisk[13]和Hintz[14]分别提出基于TCP协议的网络隐蔽信道。Fisk把隐蔽信息嵌入在TCP重置位RST字段构造隐蔽存储信道，而Hintz则把TCP协议的紧急指针位URG作为传输载体来实现隐蔽信息的传送。网络存储隐蔽信道的这些隐蔽信息一般都被入侵检测系统或者主动网络中间件（ANIs）。例如，网络协议洗涤器)通过修改网络协议首部字段的可替换位置实现把隐蔽信息替换掉，很多学者把研究重点放在网络时间隐蔽信道上。

网络时间隐蔽信道的传输载体通常是网络数据包的传输时间间隙。Cabuk[15]等人首先提出IP时间隐蔽信道方案，该方案使用基于时间间隔的编码方案。Cabuk[16]后来提出一种基于时间重放的时间隐蔽信道，简称为TRCTC。Shah[17]等人于 2006 年设计并实现了键盘装置Jitterbug，它能够泄露击键信息。Gianvecchio[18]等人于 2008 年提出一种可以模拟正常数据流的统计特性的时间隐蔽信道。

近年来，网络隐蔽信道检测技术同样成为研究的热点。网络隐蔽信道的检测技术也分为两大类：网络时间隐蔽信道检测方法和网络存储隐蔽信道检测方法。

万方数据

（1） 网络时间隐蔽信道检测。该信道一般采用数据包的发送时刻、发送间隙等特征来表示数据。因此，通常通过分析数据包的时间数据，计算检测指标，并根据指标判断是否有人构造了隐蔽信道。例如，Cabuk 等人提出一种数据包间隔的方差指标分析 IP 数据包的时间间隔隐藏的规律性。Berk 等人提出一种数据包间隔概率分布相似度指标。Gianvecchio 提出使用熵率指标度量数据包间隔的规律性，且利用修正条件熵（corrected conditional entropy）克服统计数据不足的缺点，提高熵率计算的准确性。文献[19]提出一种时域恶意主体检测法（identify malicious subjects in the time domain，简称 IMS）。该方法构建主体和客体的修改、引用记录矩阵，存在工作量大、检测分析准确性难以保证的缺点。

（2） 网络存储隐蔽信道检测。该种信道主要利用网络协议的冗余部分，包括协议控制部分、扩展部分或者常用字段加载信息，而不是直接利用协议的常规数据部分保存隐蔽信息。隐蔽消息一般嵌入在协议数据包头的保留域、未使用字段域和填充域。例如：IP 头的 TOS 字段、DF 字段或 TCP 头的标志、RST 字段等等。一般分为两种检测方法，一是对数据包的特殊位进行监视，一种是通过检验字段值的概率分布来判断。文献[20]提出一种基于马尔科夫链的 TCP 协议隐蔽信道检测方法。文献[21]提出一种基于隐马尔可夫模型的网络隐蔽信道检测模型，该方法的流程是先对获取的数据包进行数据处理和特征提取，然后对正常网络行为建立隐马尔可夫模型，最后用模型输出的测试样本序列的概率来判断是否存在异常行为。

1.3 本文的主要研究内容

本文从研究网络隐蔽信道的构造特点和检测算法出发，分析并概括了网络存储隐蔽信道的特点，研究了现有网络隐蔽信道检测算法的原理和缺陷。在这些基础上，本文提出了一种形式化的分析检测方法——基于 CSP 的网络存储隐蔽信道检测和分析模型。具体研究内容包括如下几点：

1）分析了网络存储隐蔽信道的特点，为网络协议首部定义属性和隐蔽漏洞的概念，并根据属性定义，把网络协议首部划分为三种类型。

2）提出一种基于 CSP 形式语言的网络存储隐蔽信道检测和分析通用模型，该模型对网络协议进行语义描述，定义网络协议通信系统和网络存储隐蔽信道模型中的事件与基本进程。

3）在研究和分析网络协议语义正确性的基础上，利用 CSP 语言中迹模型的提炼定义和 FDR 检测工具，验证进程之间的提炼关系来对网络存储隐蔽信道进行检测和分析。

4）以 TCP 协议为例，研究基于 CSP 的网络存储隐蔽信道检测和分析模型在

万方数据

网络协议通信系统中的应用。该验证证明了 CSP 模型的有效性和实用性。

1.4 论文主要内容及章节安排

本文共分为五章，各章安排如下：

第 1 章：首先介绍了网络隐蔽信道的研究背景和目的、意义，以及国内外相关领域的研究现状。

第 2 章：介绍了隐蔽信道的相关知识，包括其基本概念、模型表示和分类以及网络隐蔽信道的构建，重点综述常见的网络存储隐蔽信道和网络隐蔽信道检测算法，总结各个算法的缺陷。最后阐述本文采用的 CSP 形式化语言，包括其基本概念和运算符号，为第 3 章建立 CSP 模型奠定理论基础。

第 3 章：提出一种基于 CSP 的通用网络隐蔽信道检测模型，首先详细介绍整个形式化模型检测原理框架，引入设计中的语法和原语。然后阐述网络协议形式化建模的具体细节，包括协议模型、攻击者模型的构建方法和形式化进程，为第四章对 TCP 协议检测，验证模型的有效性和可用性奠定基础。

第 4 章：在这部分，我们通过设定网络存储隐蔽信道攻击者的假设条件，针对 TCP 协议，建立 CSP 形式化模型，从而验证模型的有效性和正确性。

第 5 章：总结本文的主要工作，提出进一步改进的几点意见，为下一步工作指明方向。

万方数据

第二章 隐蔽信道的相关基础知识

本章 2.1 小节介绍隐蔽信道的基本概念，中间几个小节着重于网络隐蔽信道的相关知识，包括其基本模型、构建方法、常见网络隐蔽信道及其检测算法。最后，本章着重介绍了 CSP 通信顺序进程，包括基本概念和运算符号。

2.1 隐蔽信道的基本概念

2.1.1 隐蔽信道的定义

Lampson是提出隐蔽信道概念的第一人[22]，他指出：不是被设计或者本意不是用来传输信息的通信信道。到现在，隐蔽信道的发展已有 40 年，网络隐蔽式通道的构建和检测研究受到安全研究领域研究学者的追捧。随着安全领域研究专家对隐蔽信道研究的不断深入，隐蔽信道的定义也在不断发展和完善。如表 2.1 所示，研究人员给出了关于隐蔽信道的 5 个不同定义。该表的前面的四个定义的隐蔽通道是从一个侧面描述，即从隐蔽通道来实现的载体、方法，或双方，如定义起始位置之间的通讯的特点；而 Tsai 等人更全面的给出了定义，从强制访问控制策略的层面，对隐蔽信道的性质阐述。

表 2.1 隐蔽信道的 5 个不同定义

1973年	Lampson提出隐蔽信道的定义：“隐蔽信道是一种本意既不是设计用于通信，也不是用于传递信息的通道”。
1977年	Schaefer[23]定义隐蔽信道：“如果一个信道通过改变存储单元的状态，间接改变描述该资源的状态变量，那么该信道实现从存储资源向状态变量传输秘密信息，则称该信道为隐蔽信道”。
1978年	Huskamp[24]把隐蔽信道定义为：“如果存在信道是通过资源分配策略和资源管理实现产生的，则称该信道为隐蔽信道”。
1983年	Kemmerer[25]定义隐蔽信道：“如果存在信道利用非数据客体项从一个主体向另一个主体传输信息，则称该信道为隐蔽信道”。
1990年	Tsai[26]等人提出一种隐蔽信道的新定义。即：“假设一个强制安全策略模型M和它在一个操作系统中的解释I(M)，I(M)中两个主体$I(S_h)$和$I(S_l)$之间的任何潜在通信都是隐蔽的，当且仅当模型M中的相应主体S_h和S_l之间的任何通信在M中都是非法的。”

但是，这些定义均是各个作者在研究单机系统的安全问题时提出来的，主要针对单机系统的隐蔽信道。对于基于网络环境的隐蔽信道，Cabuk 等人进行了重

万方数据

新定义:“通过网络互相连通的不同区域的计算机实现将隐蔽信息泄露出去的通讯传播机制叫做网络隐蔽信道，该机制具有很强的隐蔽性，违反系统的安全策略且很难被系统检测到。”

网络隐蔽信道的定义也随着网络技术的飞速发展而更完善，文献[27]对网络隐蔽信道的定义如下:“在互联网网络环境中，通过一种违背网络安全访问机制，逃避常规网络安全机构检测，实现隐蔽信息传送的非法信道叫网络隐蔽通道”。

在网络中，介于两个用户之间的隐蔽通信可以采用两种方式进行通信:

（1） 隐蔽数据交换，可以用输油管道问题来理解，假设存在两个输油管道 p1 和 p2，他们的半径分别为 d1 和 d2，且管道 2 位于管道 1 中所以 d2<d1。内部管道 p2 被不法分子用来输送走私的汽油(即隐蔽信息)。那么 p1 相当于合法信道，而 p2 则属于隐蔽信道。

（2） 隐蔽指示。隐蔽信道的创建者使用一种别人不知道的信息编码方案进行通信。发送者和接收者共享泄露信息的信息编码方案。例如考试问题。考生 X 通过不同的手势触发不同的事件，向考生 Y 泄露答案。

2.1.2 隐蔽信道的模型和表示

隐蔽信道的一般通信模型[28]可如图 2.1 所示。在隐蔽信道系统中，有高级用户和低级用户之分，其中高级用户为具有高级安全许可的用户，低级用户为具有低级安全代理的用户。从系统安全方面考虑，即使对系统接入自主访问控制和强制访问控制策略对用户的访问权限进行约束，未经授权的秘密信息仍然可以通过隐蔽信道由高级用户向低级用户传递。

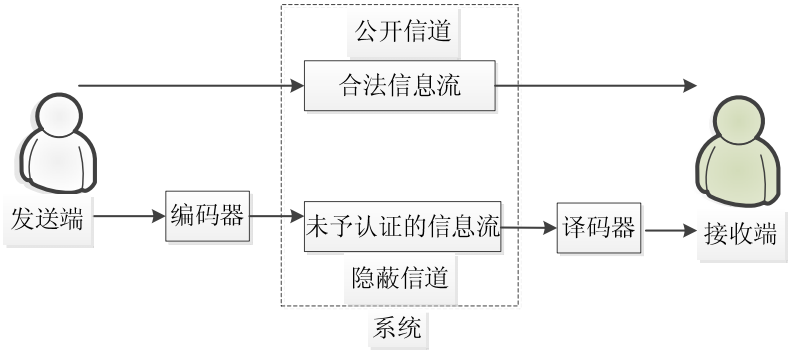

图 2.1 隐蔽信道的一般通信模型

1983 年Simmons提出“囚犯模型”[29]，该模型研究囚犯密谋偷逃出监狱的通信模型。如图 2.2 所示，两个住在只能通过书面通讯交流的不同囚室的患难囚犯爱丽丝和鲍勃，温迪负责监管爱丽丝和Bob的书信通讯内容和行为，一旦发现存在异常行为将把两人分离并加强更严格的门卫，爱丽丝和Bob将使用合同谈判的监狱条件。在这个时候，爱丽丝鲍勃的秘密信息，如果你通过传统的书面通讯，

万方数据

将会被温迪看到，逃跑是不可能成功的。所以爱丽丝通过隐蔽通道向鲍勃传达隐藏逃跑的消息。

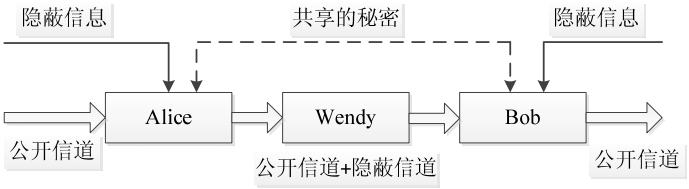

图 2.2“囚犯问题”模型

如果把囚犯问题中的Alice和Bob两名囚犯分别看作是网络中两台可以通信的计算机，那么囚犯问题的例子就可以代表网络隐蔽信道[30]。Alice和Bob之间可以创建一条隐蔽信道，传递不想被别人发现的消息。在发送端和接收端的鲍勃爱丽丝隐蔽通道共用一套机制和算法，用于建立隐蔽通道的检测，和加密和信息隐藏等认证。温迪负责网络和通信网络信息监控管理，他可以改变限制，扰乱了隐蔽通道通过他的信息。

Craver等人[31]在 1998 年，对该模型进行了不同的描述，该模型定义了三种类型的看管人分别为：

（1） 主动的看管人。看管人 Warden 不仅可以检测 Bob 和 Alice 之间进行的通信，而且还可以在保证正常通信的前提下，适当修改正在被传递的信息。

（2） 被动的看管人：看管人 Warden 可以监察 Alice 和 Bob 的通讯，然而不被赋予对通讯内容做任何修改的权利。

（3） 不怀好意的看管人：看管人 Warden 允许任意修改 Alice 和 Bob 之间传递的消息。

作为隐蔽信道的发送者和接收者，Alice 和 Bob 可以处于多个位置，如图 2.3 所示。根据 Alice 与 Bob 所处的位置，可分为相应的许多种类的隐蔽信道。

（1） Alice 与 Bob 分别位处于公开合法信道的两端，即发送端和接收端。这样隐蔽通信的发送者和接收者不仅可以实现对公开信道的控制也可以实现对隐蔽信道的控制。同时，该构造隐蔽信道的方法非常灵活，而且其带宽可控。

（2） 除了分别位于接收和发送两端外，Alice 与 Bob 可以处于接收和发送两端之间的任何两个位置之一，这些位置通常被命名为中间人位置。当 Alice 位于该位置时，每次 Alice 需要将隐蔽信息嵌入公开信道中相应的位置，然后将其发送给 Bob。

万方数据

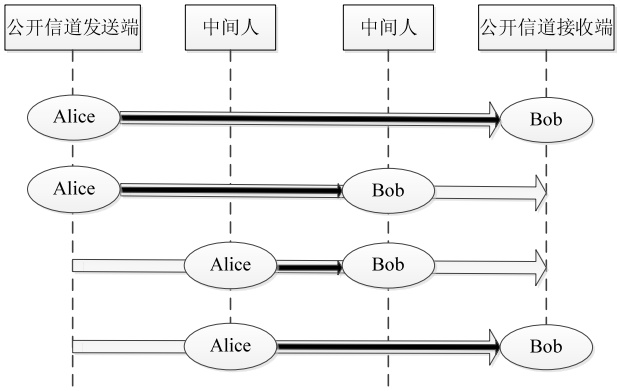

图 2.3 发送者和接收者的四种类型的隐蔽位置

2.1.3 隐蔽信道的分类

根据不同的划分标准，隐蔽信道可以被划分为不同的分类。一种普遍被认同的分类方法由Lipner[32]提出：存储隐蔽信道和时间隐蔽信道。

存储隐蔽通道是两个过程中传递的秘密信息，通过共享的内存位置，在这一过程中直接或间接地存储区写入共享的另一个过程，直接或间接地读取共享内存区域。

时间隐蔽信道则是发送进程和接收进程通过操作共享资源的时间特征，这是一个过程，采用时间调制系统资源本身，调制过程中观察到的另一个过程是真正的系统响应时间，从而实现秘密信息的传输。

两个进程之间的信息传输，调制的系统资源，实现自己的使用时间，反应时间的影响，通过在系统中运行的其他进程的一些规则，遵守活动的调制过程。

2.2 网络隐蔽信道

2.1 节已经介绍了隐蔽信道的基本概念。在本节，我们重点介绍网络环境中的隐蔽式信道的构建方式。

借鉴传统的隐蔽信道的分类方式，网络隐蔽信道同样可以分为存储式隐蔽信道和时间式隐蔽信道。

网络存储式隐蔽通道是有记忆信道，它可以利用网络协议首部的安全缺陷或特定领域的设计漏洞，使用数据包报头发送信息。这种方法不仅可以使用底层网络协议中的 IP 协议和 ICMP 协议，也可以使用高层的 TCP 协议，HTTP 协议和 DNS 协议。

网络隐蔽时间信道使用时间特性和流动特性的正常通信过程中信息传递。例如，在某一段时间，提前为流量设置一个阈值，当阈值为 10Kb/s 时，超过阈值就

万方数据

表示传输比特位“1”，反之就表示传输比特位“0”。

2.2.1 网络存储式隐蔽信道的构建

网络隐蔽式通道隐藏在正常的网络通信，或伪装成一个正常的网络流量，使各种防火墙和入侵检测系统的检测系统是很难找到他们的基本模型，如图 2.4 所示。图中主机 A 和主机 B 分别为网络隐蔽信道的发送方和接收方。主机 A 与主机 B 需要通过以下几步完成对隐蔽信道的构建：

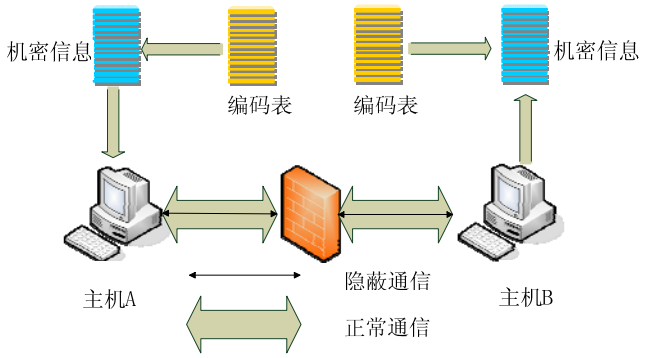

图 2.4 网络隐蔽信道的基本模型

（1） 选择合适的网络协议。构建网络存储隐蔽信道时，采用的隐蔽载体即为网络协议。如图 2.5 所示的计算机网络协议参考模型。每一层网络协议都可以被用来实现隐蔽通道的建立，但存在效率问题和不同程度的困难。

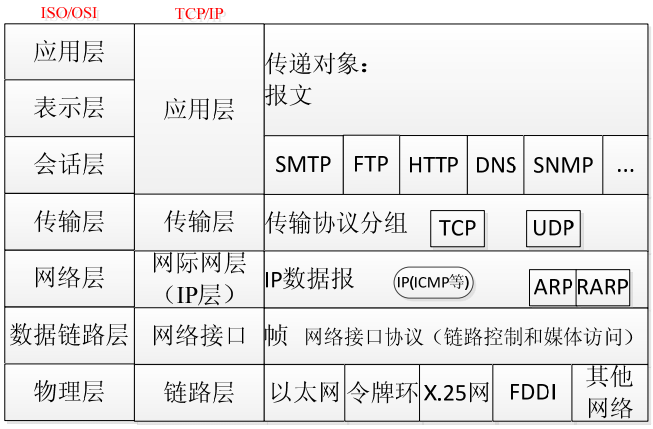

图 2.5 OSI 参考模型和 TCP/IP 参考模型

（2） 发现在网络协议设计中的漏洞。主机 A，B 需要找到一个协议的消息可以作为该领域的一个载体，我们隐藏字段中的信息。

（3） 选择编解码机制，对机密信息进行编码。或者，对发送的数据进行加密，或者先压缩编码然后加密，确保隐蔽通道的高隐蔽性，避免被防火墙和入侵检测系统检测和拦截。

网络存储隐蔽信道根据数据包头域中特定的字段来传递隐蔽信息。例如，一

万方数据

些常见的网络协议（如 IP，ICMP，TCP，HTTP）头部存在一些未被使用的领域，可以基于此建立存储式的隐蔽通道。Handel 等人把 IP 协议 TOS 字段作为载体构建隐蔽通道。Hintz 指出，TCP 的紧急指针字段能作为载体，传达隐蔽信息。Fisk 指出使用 TCP 重置位RST 字段构造隐蔽信道。

2.2.2 网络时间隐蔽通道的构建

利用网络数据包分组在网络中转和传输过程中生成的时间特征调制隐蔽消息，叫作时间式隐蔽式通道。例如和时间有关的数据包流量特征、数据包到达间隔等特征来对信息进行编码和解码。即：

（1） 基于流量特征编码构建隐蔽信道

2004 年 Cabuk 等人首次提出 IP 时间隐蔽信道：在该信道中，发送者和接收者约定一定的时间间隔，在传输过程中的每个时间段内，发送者向接收者发送一个数据包，或者保持静默，这样建立了基于流量特征编码的隐蔽信道。

（2） 基于数据包到达的时间间隔构建隐蔽信道

发送和接收者提前秘密商定相邻分组到达时间间隔的调制机制，不同的时间间隔，代表着不同的隐蔽信息。例如，通信双方约定：当传输的是“0”，两个相邻的数据包到达时间间隔超过阈值 1s；而当时间间隔小于阈值 0.5s 时，表示传输“1”。

（3） 基于数据包长度变化特征编码构建隐蔽信道

钱文玉等研究学者提出一种基于数据包的隐蔽信道建造方法，该方法通过对不同长度的数据包进行编码构建隐蔽信道。表 2.2 表示不同长度数据包的编码规则。把第一个传递秘密信息加密的密文，二进制，随后以 2 位，根据表 2.2 获得两比特的码对应的数据包的长度范围内。然后根据协议数据包的发送者在结构，然后对信息隐藏的 F 的对应范围的分组长度的调整，并将调整数据包接收器。该接收机根据特定的接收数据包，数据包长度的二进制代码对应的搜索，最后把所有的碎片粘合起来，解开隐藏的信息。

表 2.2 数据包长度编码规则

单个数据包长度	数字编码	二进制编码
[100,150)	0	0
[150,200)	1	1
[200,250)	2	10
[250,300)	3	11

（4） 基于数据包到达的顺序构建隐蔽信道

假设构造三种类类型的合法数据包 X， Y， Z，网络通信环境能够保证发送

万方数据

的顺序和接收数据包的顺序一致。正常情况下，发送方依次将3个数据包按照数据包的生成顺序发出。不妨设发送顺序为“$X \to Y \to Z$”，则接收方也按照“$X \to Y \to Z$”的顺序接收数据包。此时可根据数据包到达顺序构建时间隐蔽信道，如表2.3所示。例如，要传输隐蔽信息“01”，则发送方只需要以$X \to Z \to Y$的顺序发送数据包。

表2.3 数据包顺序隐蔽信道编码规则

连续数据包到达顺序	数字编码	二进制编码
X→Y→Z	0	0
X→Z→Y	1	1
Y→X→Z	2	10
Y→Z→X	3	11

2.3 网络存储隐蔽信道

本文的主要内容是存储类型的网络隐蔽式通道检测，本小节为所有的存储网络隐蔽通道构造算法做一个详尽的介绍。

采用公式$p = F(c,d,m)$表示网络存储隐蔽信道的嵌入机制，其中，c表示最初始数据包，d表示隐蔽信息嵌入的位置，m表示要发送的隐蔽消息，F表示置换函数，则p表示嵌入隐蔽信息的数据包。网络存储隐蔽信道可以利用TCP/IP协议头部的设计缺陷来存储隐蔽信息。下面将分别介绍计算机系统中每一层网络协议的典型网络存储隐蔽信道。

2.3.1 基于物理层和数据链路层协议的隐蔽信道

1. 基于CTS/RTS控制信号的物理层隐蔽信道

一种基于隐蔽通道构建物理层的方法是通过隐藏信息的串行数据流传输控制，如图2.6所示。CTS/RTS（清除发送/准备送）是用来控制数据流的中断信号。图2.6上面和下面的两个画面的构图，表现出对CTS / RTS控制信号的数据流时的信道，下图显示没有CTS/RTS数据流控制信号。因此，通过编码和CTS / RTS信号调制方式，基于隐蔽通道的物理层实现。

万方数据

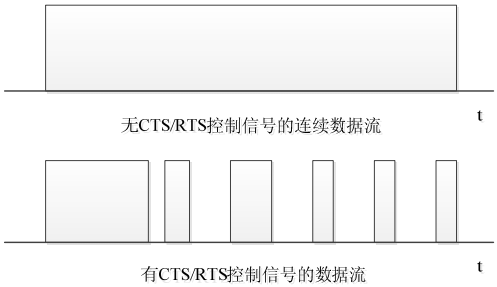

图 2.6 两种不同的串行数据流随时间变化的情况对比图

2. 基于数据帧结构的数据链路层的隐蔽信道

在数据链路层的规定，两个点通过介质直接连接之间的数据传输。计算机系统的数据帧，并发送到物理电路，数据帧头包含发送和接收的数据帧信息，尾包含错误控制信息，使用 CRC 冗余校验码校验。在汉德尔等人的文章，一个隐藏的信息添加到数据帧结构的隐蔽通道的施工方法的信息传输结束。当你需要提供一个连接与分离的隐藏信息的作用机制在数据链路层实现隐蔽通道。

2.3.2 基于网络层协议的隐蔽信道

1. 基于 IP 包的隐蔽信道

基于IP协议的隐蔽信道，即利用IP报文头部的冗余字段或者扩展数据部分构造隐蔽信道，例如：TOS、标识字段、标志位和片偏移字段、生存期字段、源IP地址字段、IP选项域字段。假设发送方和接收方采用标识域传递信息，则可以首先要传输的数据转换成二进制代码，最终形成一个 7bit二进制码乘以 256 进行扩展[33]，取代IPID字段的内容，也被称为ASCII填充方法。

2. 基于 ICMP 包的隐蔽信道

ICMP 协议的全称是网际报文控制协议。ICMP 的报文格式，如图 2.7 所示。ICMP 消息的固定格式的统一的第一个 4 字节，主要是由类型，代码和校验和三个分量，其余部分根据不同类型和不同的内容。

类型Type（8或0）	代码Code（0）	校验和Checksum
标识符Identifier	序号Sequence Number	
可选数据部分(可任意填充)Optional Data		

图 2.7 ICMP 报文格式

在 ICMP 消息，最常用的两个消息是回声（ Echo ）响应消息（0 型）和回声请求消息（8 型），这是常用的 ping 命令使用的消息类型。ICMP 数据包一般可以通过防火墙，在运行 ping 程序时，发送方主机将向目标主机发送回应请求消息，

万方数据

用于确定发方和目标主机是否是可连通。在这个过程中可以将隐藏数据嵌入到ICMP 数据包选项字段，假装记录路由器地址和途径中转站时间。因为ping命令包一般都被当作是良性的网络流量，因此网络访问控制系统和其他设备一般不会对ping数据包可选部分进行检验，所以我们可以把隐藏的信息嵌入到ping数据包的可选数据部分。

2.3.3 基于传输层协议的隐蔽信道

1. 基于 TCP 协议的隐蔽信道

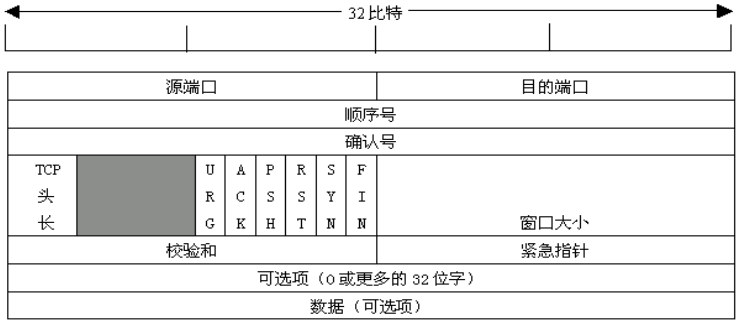

图 2.8 TCP 报文格式

TCP 报文格式如图 2.8 所示。基于不同的 TCP 首部字段，常见的基于 TCP 协议的隐蔽信道主要分为以下几类：

（1） 初始序列号（ISN）。主机会为TCP报文随机分配一个初始值作为ISN序列号，该字段可以被用作隐蔽载体传输隐蔽信息[34]。文献[35]提出采用ISN构建TCP隐蔽信道，数据在被替换到SEQ字段之前将采用特定的加密算法处理，以便使得观察到的SEQ字段和伪随机过程更相接近接近，增强隐蔽信道的隐蔽性。

（2） 确认序号（AN）。在 TCP 连接中，ACK 的值指示一个待接收数据包的序列码。发送方主机可以和一台本地安全机制授权的公共主机通讯，这时可以利用 TCP 报文的 ACK，通过第三方和接收方主机之间创建隐蔽信道，把隐蔽信息传送给接收方。

（3） 预留字段（Reserved）。预留字段占位 6bit，它是供未来修正原始协议的，在正常网络通信时一般不使用它，把它的位置赋予 0。因此，可以利用此字段填充隐蔽信息。

（4） 可选项字段（0 或多个 32 位字）。这是在 TCP 协议头部嵌入隐蔽信息的最常见的一种形式。当把隐蔽信息嵌入在该字段时，需要接收者在合法的 TCP 处理程序之前提取隐蔽信息。

2. 基于 UDP 协议的隐蔽信道

万方数据

图 2.9 UDP 报文格式

基于 UDP 协议的隐蔽信道，主要分为以下两类：

（1） 校验和（checksum）字段：该字段是可选字段，它置为 0 和 1，代表两种不同的状态：使用或者不使用状态。可以使用这两种不同的状态调制隐蔽式信息，可以隐藏信息1bit/ 包。

（2） 端口字段：根据端口数值的多样性，以及应用程序不同所使用的端口号不一样的特性，可以构造如下类型的隐蔽通道。采用8种不一样的端口号，每个端口采用二进制进行编码，代表3bit。如表 2.4 所示：基于端口字段构造隐蔽式信道。

表 2.4 利用UDP端口号构建隐蔽信道

UDP Port Number	1200	1235	1360	1480	1500	1630	1800	2000
Bit information	0	1	10	11	100	101	110	111

2.3.4 基于应用层协议的隐蔽信道

应用层位于传输层之上，应用层上有很多协议，例如：SMTP、FTP、HTTP、DNS、SNMP 等等。基于应用层协议信令语句的隐蔽通道可使用不同的信令语句表示不同的秘密信息。这样就可以把隐蔽信息嵌入到正常的应用层数据包中实现隐蔽信道的构造，因此隐蔽信道的载体是特定应用层协议中多种不同类型的通讯信令和命令语句。下面介绍基于SMTP协议的隐蔽信道构造技术。

SMTP 协议描述了电子邮件信息格式、传输和处理方法，确保正确、可靠地处理电子邮件。

SMTP 协议为邮件用户提供两个 SMTP 进程交换信息的通讯机制。SMTP 协议包含 21 种响应信息和 14 条命令，只有5条用于邮件发送分别是：HELO，MAIL，RCPT，DATA，QUIT，可以选择它们作为隐蔽通讯的信息载体。

隐蔽信道的发送者可以选用发送邮件的常用命令中的几条作为隐蔽通信的载体。例如，编码命令集合$S=\{HELO, MAIL, RCPT, DATA\}$，由其中的 4 条命令组成。如表 2.5 所示，2bit 编码位可采用四条命令。继而发送端可以据此调制协议

万方数据

命令类型，再把发出命令序列。当接收端接收到的数据包的命令不会做出响应，而是依据编码表提取秘密信息。

表 2.5 利用 SMTP 命令构造隐蔽信道

命令语句	HELO	MAIL	RCPT	DATA
2bit编码位	0	1	10	11

2.4 网络隐蔽信道检测算法

网络隐蔽通道检测和分析技术已经相当成熟，已有的隐蔽通道检测技术分为五种不同的类型。

2.4.1 基于特征的网络隐蔽式信道检测

基于特征的检测，是指针对数据包固定字段的检测，识别数据包形式上的信息即语法信息，通过搜索特征进行数据分析匹配，这种机制适用于已知的隐蔽信道，而无法对加密信息进行检测，不能总结攻击形式和判别新式隐蔽信道。

对网络上的数据流作分析，当其中每个或某几个条件满足时，系统就判断有网络隐蔽信道存在。由于计算机程序对单穿的特征匹配比较容易实现，并且具有易于构建、易于管理、扩充方便的优势。

基于特征的检测又分为基于主机的检测技术和基于网络的检测技术。基于主机的检测技术是对监控或监视主机的网络数据、主机行为特征，如主机接收到的数据包、系统日记、审计日记等，按照相应规则进行判别，从而确定当前主机是否被网络隐蔽信道入侵。基于网络的网络数据包捕获的检测方法依据相应的规则，如数据包特征、数据流的特点等，进行判别，从而验证网络中是不是含有入侵流或恶意攻击。

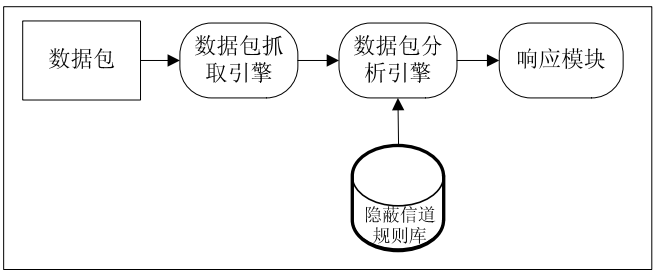

图 2.10 基于特征的检测系统

基于特征的检测系统组成如图 2.10 所示。检测方法的分析对象，即数据来源是网络流经的数据包。其中“隐蔽通道规则库”是该检测方法的知识库，定义了各种网络隐蔽类通道的特征。“响应模块”是在系统发现可疑的网络数据包时，采

万方数据

取的处理步骤。

2.4.2 基于行为的网络隐蔽信道检测

基于行为的检测，是指抽象出寻常网络模式下的某些普遍特征，判别的是协议使用的目的和效用价值等信息，例如通过对所监控的网络建立流量模型，实时统计的行为与模型的阈值之间存在很大差异，监控的网络数据流会对此差异发出告诫，然后放在日志中，并根据该差异判断是否在网络流中含有秘密信息。

文献[36]提出一种使用信息论中的校正熵的方法，分析网络的行为，来检测隐蔽信道。该方法使用将秘密信息嵌入到用户的操作序列的方法，从而肯定导致网络数据流进行检测的条件熵的变化。

2.4.3 基于统计的网络隐蔽信道检测

基于统计的网络隐蔽信道检测，是把网络数据包流量作为检测对象，通过分析收集的数据包的统计特性，实现对网络隐蔽信道的分析和预测。

对网络数据包流量的研究得出，正常数据包的流量统计服从一定的统计规律，而不是随机分布的。尤其是，在一定的时间内，对于一个稳定的局域网系统，其网络流量一般都满足如下规则：

（1） 局域网与外部网络通信流通的的数据流量符合泊松分布。

（2） 局域网内部的网络流量满足一些突发性约束条件。

在正常网络流中加入隐蔽流量，会影响数据包流量的数学分布。因此，可以通过检测和分析数据包流量的统计特性判别网络流量中是否有隐藏的隐蔽通道。

2.4.4 基于人工智能的网络隐蔽信道检测

基于人工智能的检测，例如文献[35]中采用神经式网络将隐蔽式信道的特征、规律及效用等信息通过神经元之间的连接构成模式图来学习理解，虽然这种模式图不像数学表达式那样具有推演和归纳的特点，但它模糊性和可兼容性的固有特点能较好地满足基于“全信息”的分类要求。而Sohn等提出一种采用支持向量机检测IP的ID字段和TCP协议的ISN字段中潜在的网络隐蔽类通道的方法[36]。

2.4.5 基于信息流的网络隐蔽信道检测

信息流分析是一个对程序的输出变量寻找输入变量的子集集合的过程。它需要提前制定程序和输入输出变量集合。

万方数据

信息流分析方法包含语义信息流分析方法和语法信息流分析方法。

1. Tsai 语义信息流检测方法

Tsai 等对语法信息流增加了语义分析，并提出一种用于隐蔽信道存储信道标识的新方法。语义信息流模型的方法步骤如下：

（1）分析编码语言的语义、内核代码中的数据结构，挖掘其中输入变量和输出变量的可见性以及可修改性。

（2）共享变量的分析与总结，并记录隐蔽存储信道。

表 2.6 语义信息流分析方法的优点和缺点

优点	缺点
1、适用于源代码级的形式化分析，并易于自动分析。	缺乏自动化工具。不能在TCB原语中确切取得安置隐蔽信道处置代码的位置。
2、能发现大量的伪非法流，减轻人工分析伪随机数流的负担。	
3、能找到被查看/修改的内核共享变量的位置，方便确定放置隐蔽信道处置代码的位置。	

语义信息流的优点和缺点如表 2.6 所示。

2. 语法信息流检测方法

文献[37]中 Denning 的信息流格模型是语法信息流方法中最有代表性的一个。语法信息流方法的分析对象是 TCB 原语设计或源代码语句，分析方法是对信息流模型作形式化描述，$FM = \langle N, P, SC, \oplus, \rightarrow \rangle$。其中：$N = \{a, b, ...\}$ 是逻辑存储单元的集合，即将读、写该变量的系统调用与该变量共同构成五元组。应用系统强制安全规则分析两个系统调用对该变量的读写路径，判断强制安全策略的实现是否正确。

语法信息流方法的分析步骤是：

（1）从每一条 TCB 原语或源代码语句抽象出信息流语义。

信息流分为两种类型：明流和暗流。如表 2.7 所示。

（2） 定义相关的安全信息流策略。输出信息流的方式是通过在系统的底层代码和规范上应用相关的安全信息流公式。如，赋值语句 $y := x$ 表示信息从变量 x 流向变量 y，则变量 y 的安全级别必须支配变量 x，信息流公式可以表示成 $SL(x) \geq SL(y)$，其中 $SL(x)$ 表示变量 x 的安全级别。

万方数据

（3） 证明信息流公式的正确性。此时分为两种情况，当可以证实信息流公式的正确性，则表示系统中不存在隐蔽信道；当无法判断信息流公式的正确性时，则需要进一步分析语句的对应语义，并判断该信息流是不是能构造真正的信道。

表 2.7 语法信息流示意

信息流类型	例子
明流/显式信息流	赋值语句产生明流，例如a：=b产生由b到a的信息流，用a←b表示。其中，→表示安全级别之间的信息流关系，是一个偏序关系，当且仅当主体的信息可以被容许从A流向主体B时表示为A→B。
暗流/隐式信息流	条件语句生成暗流，如，if x=a then y:=b else z:=c产生的信息流包含明流和暗流：其中明流是y←b和z←c，暗流是y←x和z←x。

语法信息流的优点和缺点如表 2.8 所示。语法信息流方法从语句语法分析的角度出发，会产生大量的伪随机流，增加人工分析的额外工作量。

表 2.8 语法信息流识别方法的优点和缺点

优点	缺点
1、可以应用于形式化顶层规范和 源代码，并易于自动分析。	1、不能应用于描述性顶层规范。
2、可以逐个对单个函数或TCB原语采用增量式分析的分析方法。	2、对完整的程序中的每条语句进行分析，会产生大量的伪非法信息流，增加人工分析的工作量。
3、搜索彻底，不会遗漏任何可能产 生隐蔽信道的非法流	3、不能在TCB原语中准确获得安置隐蔽信道处置代码的位置。

3.回溯搜索法

卿斯汉等[38]设计了一种分析代码语言的标识方法，该方法使用与Tsai 语义信息流法相同的处理规则。

上述几种语义、语法分析方法普遍存在工作量大的问题，本文采用一种形式化方法——通信顺序进程（CSP）。CSP 形式化语言是以进程代数为基础的数学方法，且采用 FDR 模型检测进行自动化验证和检测，能够降低人工工作量，提高检测和分析网络存储隐蔽信道的效率和性能。CSP 把网络协议抽象成一个交互的进程通信系统，把协议主体执行的通信动作抽象成一个个 CSP 语言描述的进程。在

万方数据

下一小节介绍 CSP 的基本概念和运算符号。

2.5 形式化语言：CSP

在网络协议安全分析和研究领域，形式化方法得到越来越多的应用和广泛的重视。网络存储隐蔽信道基于网络协议的构建，所以形式化方法和模型检测技术同样适用于网络存储隐蔽信道的检测和分析。

越来越多的研究专家实践证明，在安全协议分析、发现安全协议漏洞等方面，CSP形式化方法是一种很有效的研究方法。1996 年，Lowe[39]第一次使用CSP语言和模型检测技术分析NSPK安全协议，并成功发现了一个攻击行为。后来，随着对CSP和FDR(Failures-Divergence Refinement)联合使用的实际研究， Roscoe 认为 CSP 方法是采用形式化方法分析安全协议的一种新思路[40]。目前，在协议验证、进程模拟以及性质验证领域，CSP方法占据着一席之地。

为了描述通过消息传输进行通信的并行系统，C. A. R. Hoare教授[41]提出了本文中使用的CSP 形式化语言。CSP 方法是一种描述语言，它可以把安全目标的行为和性质描述为进程，这些进程通过交换消息而相互作用[42]。

2.5.1 CSP基本概念

进程是CSP形式化语言中的基本描述单元，是系统与其环境间交互作用的一种数学抽象。从本质上说，一个进程是事件在时间上的一个序列动作[41]。

通信这个概念代表进程与其它进程或它们的环境之间彼此交互过程的CSP的形式化描述方法。通信可以描述为两种形式，这两种形式包含可见事件和动作。

一个通信系统可以通过 CSP 建立由一系列顺序执行或是并发执行的互相通信的进程交互作用，实现对该通信系统可达状态的描述。进程模型可代表客体对象的行为，以及这些行为的原则。系统内部要完成的具体细节对描述它们的CSP进程来说是不可见的，并且系统具体细节实现的内部行为和系统所处的外部环境没有直接交互信息的过程。

2.5.2 CSP基本运算符号

常用的CSP基本运算符号如下所示[43]：

（1） 前缀（ prefixing ）结构

CSP进程初始化以及按照既定规则执行的触发因子是系统中的外部环境以及进程与外部环境相互交互的各种不同活动[43]。通过把外部触发通信行为抽象描述

万方数据

为前缀结构，实现把外部的触发事件或者动作行为引入到CSP 系统进程中[44]。假设存在一个进程 P 和某个具体事件 a，则 $a \to P$ 表示进程先执行a，然后依照P进程行动。也即表示事件a和进程P通信。

例如，一个由有限的活动序列组成的进程 $A = a \to b$ 表示通过执行外部事件a，可以引发内部活动事件b[44]。

在前缀表达式结构中，一般存在两种类型的通信机制：如果 $A \subseteq \Sigma$ 是可见动作集合的一个子集，那么进程 $?x:A \to P(x)$ 表示当选择满足条件 $x \in A$ 的任意变量 x，则遵从 $P(x)$ 执行，其中 x 是 $P(x)$ 进程中的内部变量。

（2） 选择运算符

该运算符号由“□”表示。从集合的角度出发，“□”代表 B 从集合中选择任意一个可能的元素作为自己的赋值；从进程的角度出发，“□”代表提供两个进程所有可能执行项的两个集合，然后遵从其中一个执行的动作或者事件继续执行一个进程集合中的动作[46]。如果 $A = B \cup C$，那么

$$?x:A \to P(x) = (?x:B \to P(x)) \square (?x:C \to P(x)) \tag{2-1}$$

对于选择运算符，也可以表示成如下式所示的表达式：

$$(a \to a \to A) \square (a \to b \to A) \tag{2-2}$$

式(2-2)为非确定性的表达式，因在运行动作或者事件 a 之后，需要继续执行 a 或 b。非确定性在系统执行过程中经常会出现，因此是一个很重要的概念，在CSP 引进非确定性选择运算符号“Π”。PΠQ 表示系统进程的运行具有不可预测性，会选择 P 或者 Q 中一个继续执行。

（3） 并行运算符

该运算符号由“| |”表示。当描述存在并行交互通信关系的一般网络时，CSP 提供一些进程运行过程中可以同步的事件的集合，并把彼此之间并行运行的进程通过该运算符号放置在一起。例如，接口并行符 $P|_{Y}|Q$ 表示 P 和 Q 在 Y 的所有事件中同步。当描述复杂的网络时，为每个进程分配一个字母表的概念，定义为该进程能够执行的动作的集合。例如，$P_{x}||_{y}Q$ 代表 P 与 Q 并联，且 P 和 Q 的字母表分别是 X 和 Y。

穿插并行运算符“|||”是“| |”的缩写，例如，P|||Q 表示 P 和 Q 无需在任何事件中同步，两个进程分别执行。

2.5.3 CSP的迹模型概念和提炼概念

一个进程从一个初始时刻开始运行，到一个新时刻截止，该时间段内发生的

万方数据

所有事件被一个有限集合记录下来，并把它定义为迹[44]。迹可以用$\langle A\rangle$表示，A为一个有限序列，中间有逗号隔开。进程的有限迹可以把进程随着时间的变化运行的可见事件作为有限序列记录为集合的形式。

进程的迹的集合可以采用递归的形式加以定义：

（1）进程的迹为非空集合。

（2）进程的迹为前缀闭包即如果s^t为进程的迹则s也为进程的迹。

通过以上的形式化定义可知，进程的迹是进程中可能出现的事件和活动的顺序记录迹。进程P的迹表示为$\text{traces}(P)$。通过对进程迹的分析可以直观的观察CSP模型的各种性质。集合在CSP中的表示方法为$\{a,b,c,...\}$，其中a,b,c所代表的内容为集合中的事件，序列在CSP中的表示方法为$\langle a,b,c,...\rangle$。可以通过一个叫作连接运算符的符号“^”将两个迹连接起来。

对通信实体过程中的行为模式的过程中一个CSP的描述，如果行为模式匹配，它可以与行为取代原来的行为：我们需要一个提炼关系以确保过程符合另一个进程的所有属性。因此，大型系统的描述可以用在提炼系统的特性的描述方法。

结构提炼的定义：

如果对于进程A其所有特定行为在进程B中都有包含，此时可以定义进程A提炼进程B，记作$A \subseteq B$。

迹模型的提炼：

如果对于进程A其所有能产生的迹在进程B的迹中都有包含，此时可以定义进程A迹模型提炼进程B，记作$B \subseteq_T A$。

例如，存在两个进程P和Q，判断Q是不是P的一种提炼？现在，假设Q总是遵守这些约束，仅拒绝存在于 P 的实效集合中的事件，并且只接受其迹集合中的事件。此时 Q 具有 P 的所有属性，因此称 Q 是 P 的提炼。

迹模型提炼记为$P \sqsubseteq_T Q$，即满足迹模型$\text{traces}(Q) \subseteq \text{traces}(P)$。

同样可以对进程中可能产生的非确定进行提炼：

如果对于进程A，其所有可能产生的非确定现象在进程B的迹中都有包含，此时可以定义进程A非确定现象提炼进程B记作$B \subseteq_D A$。

根据结构提炼的定义可知A的所具有的特定属性或者性质，在进程B中都有体现，因此使用进程A来代替进程 B不会带来更大规模性的错误，因为A所代表的特定性质仅仅是B该性质的一个子集。很多形式化分析工具借助结构提炼的方法可以快速高效的验证CSP表达式所建立模型的各种不同的性质。

万方数据

2.6 本章小结

本章介绍了隐蔽信道的基本概念，重点介绍了网络隐蔽信道的构建。然后又着重阐述了几种常见的网络存储隐蔽信道，以及网络隐蔽信道检测算法。本章最后介绍了形式化语言 CSP，为第三章建立模型做理论准备。

万方数据

第三章 网络存储隐蔽信道检测模型设计

针对本文 2.3 节介绍的网络分层模型中各层存储隐蔽信道的特点，本章采用 CSP 建立一种形式化的网络存储隐蔽信道检测和分析模型。CSP 具有严密定义的形式化语义，采用进程代数 CSP 来描述网络存储隐蔽信道的语义，能够将网络协议实体的通信行为与清晰的操作语义联系起来，能够对网络协议中的存储隐蔽信道进行检测和分析。

3.1 基于CSP的通用检测模型

如图 3.1 所示为基于 CSP 的网络存储隐蔽信道通用检测模型。该模型包括三个部分：理论知识学习阶段，CSP 形式化建模阶段，FDR 检测分析阶段。

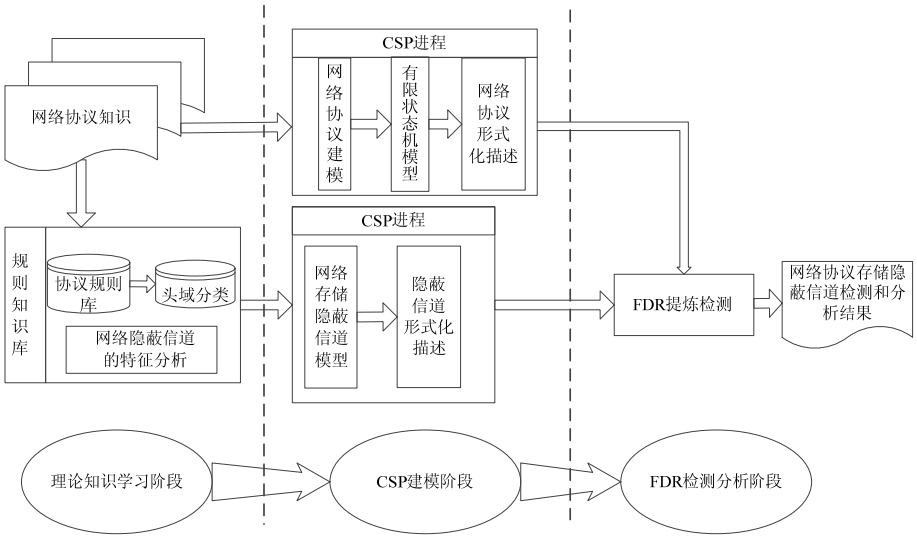

图 3.1 基于 CSP 的通用检测模型

1. 理论知识学习阶段

该阶段主要负责分析网络协议和网络存储隐蔽信道构建机制，获得对 CSP 形式化建模分析和验证有用的信息。包括：

（1） 分析网络协议相关知识。这一部分检测和分析网络存储隐蔽信道的理论基础，主要通过学习 RFC 文档中所声明的网络协议以及这些网络协议的实际应用场景，得出协议的通信规则与协议报文的组成。

（2） 分析网络存储式隐蔽通道的基本模型。对其构建规则和信道特征进行分类和总结，提取网络存储隐蔽信道相关的隐蔽漏洞资源库和规则库。

2. CSP 建模阶段

CSP 建模阶段是在对网络协议知识和网络存储隐蔽信道的特征分析基础上，对网络协议交互系统建立形式化模型。CSP 把网络协议当作交互实体，这样就可

万方数据

以很容易地描绘协议间的交互关联和仿真一个协议交互体系。

从网络协议的最初设计分析，网络协议的三个基本要素包括：

（1） 语法，这是数据和系统信息的结构或格式，包括协议的不同报文、协议的基本状态以及其他事件。

（2） 语义，是交互消息的数据结构中不同值的含义。

（3） 同步，即对事件处理的逻辑顺序。同步是协议中事件的顺序序列，协议实体的状态迁移。

在 CSP 建模过程中，就可以把网络协议的三个基本要素映射为 CSP 中的进程。语法即是数据与控制消息的结构或格式，是基本数据，包括协议的不同报文、协议的基本状态以及其他事件。语义是需要发送控制信息，完成什么动作和做出什么样的反应，同步是一个详细描述时间序列。在网络协议的模型中，语义和同步是协议中事件的顺序序列，协议实体的状态迁移。

CSP 建模阶段的具体步骤包括：

（1） 分析网络协议的基本要素，抽象出网络协议的语法和语义，包括基本数据类型、不同的报文、协议的基本状态等。

（2） 在（1）的基础上，把网络协议实体描述成一个 CSP 进程。进程是一个事件序列，网络协议交互过程中的事件包括协议实体发出何种控制信息，完成何种动作以及做出何种响应，即这些事件包括了协议的语义说明，又包括了协议的语法说明。该步骤构建了网络协议实体模型。

（3） 把网络存储隐蔽信道的构建过程抽象为一个攻击者模型，该攻击者模型是攻击者的一个进程，通过对协议的分析，得出攻击者可能具有的构建网络存储隐蔽信道的能力，例如：窃听、截获报文、篡改报文、嵌入隐蔽消息等。

（4） 在（2）和（3）的基础上，把网络协议实体模型和攻击者模型相结合，组成现实环境下网络协议交互系统的形式化模型。

从模型结构层面，形式化模型主要包括三个主要内容：环境变量、协议实体与系统进程。环境变量为网络协议应用和实施的网络环境，在模型中体现在通信信道和攻击者进程上。协议实体为该网络协议所有的参与实施者，在模型中体现运行该协议的实体。系统进程可理解为网络协议的实施行为和实现步骤，为协议的基本设计规则。通过模型检测的方法来验证协议模型是否满足这些安全约束和安全条件。

3. FDR 检测分析阶段

该阶段通过 FDR 检测工具，把 CSP 模型作为输入，检测和分析网络存储隐蔽信道的组成方式。FDR 检测加入网络存储隐蔽信道的网络协议交互系统是否还满足所要验证的性质。在检测过程中，不符合该实例的描述的，把它以反例的形式输出。然后，从得到的反例中分析出对应的网络存储式隐蔽类通道，并生成最

万方数据

终的检测结果和分析成果报告。

3.2 网络协议的CSP描述

3.2.1 网络协议的基本要素

网络协议是计算机网络的数据交换的一组规则、标准或一个约定的集合。协议是一种定义，专用来作为描述进程之间进行信息交换数据时的规则术语。网络协议的 CSP 模型即采用 CSP 对网络协议规约进行形式化建模和描述。

CSP 可以通过进程代数语言、运算法则描述网络协议中的规则术语、通信机制。下面阐述通过 CSP 对协议建模的过程。

网络协议交互过程中的事件包括两个协议实体之间交互的控制类型的信令消息组成的事件集合，具体实现的连接动作集合以及做出回应消息的数据包集合，即这些事件包括了协议的语义说明，又包括了协议的语法说明。CSP 进程是一个事件序列，因此我们把网络协议交互过程中的事件用 CSP 进程描述。

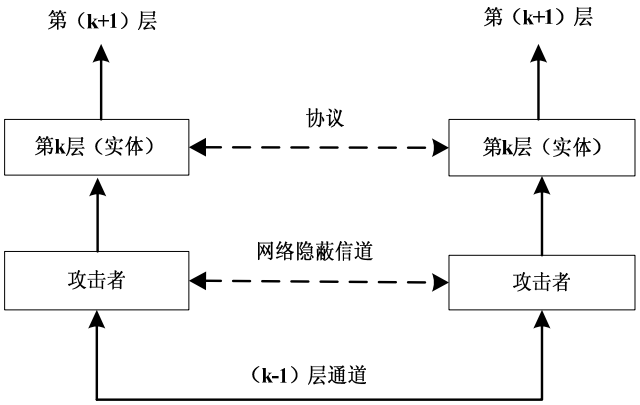

图 3.2 协议实体、服务以及协议之间的关系

从模型结构层面，计算机网络根据功能划分成多个层次，每一层都有其特定的功能。如图 3.2 所示，分层的网络协议涉及到一些名词概念：

（1） 实体，软硬件进程，可以用来表示任何可发送或接收信息。

（2） 协议，一种规则的集合，控制在一组或者更多对等实体之间完成通信。

（3） 服务，某一层向它的上一层提供的一组原语，也会涉及到两层之间的接口，其中下层是服务提供者，而上层是服务的用户。

（4） 服务原语，上层协议为使用下层协议所提供的服务而和下层协议进行数据交换的命令叫做服务原语。

（5） 攻击者，假设被网络存储隐蔽信道的构建者，具有监听、篡改报文的能力，实现隐蔽信息的传输。

万方数据

（6） 服务数据单元，相邻两层进行交换数据的单位。

CSP 形式化模型包括三个主要内容：协议实体、环境变量与系统进程。协议实体为该网络协议所有的参与实施者，在模型中体现运行该协议的实体，在模型中被表示为 CSP 中的进程。环境变量为网络协议应用和实施的网络环境，在模型中体现在通信信道和攻击者进程上。系统进程可理解为网络协议的实施行为和实现步骤，为协议提供的服务和对应的服务原语。服务数据单元被表示为事件，进而网络协议被表示为一个通信顺序进程的集合。

3.2.2 网络协议模型中的进程

如图 3.2 所示，网络协议模型包含四个角色：两个第 k 层协议实体（发送方和接收方）和两个攻击者。我们需要分别为这四个角色建立 CSP 进程模型，在每一种网络分层协议的情形下都使得进程能够传输适当的消息序列。

下面我们约定一些 CSP 建模过程中使用的语法说明。

（1） 常量，表示一个具体数值，例如协议的报文字段中的 Ack 和 Flag, 还有一些自然数常量和随机数常量。

（2） 变量，表示未知的状态变量，事件的变量或者数值变量，变量使用小写字母或者小写的字符串表示，例如 i,j,k,x,y,z 等。本文的模型也使用一些带有下标的变量，如 $x[i], z[k]$ 等。

（3） 表达式，由变量、常量和一些运算符号构成。例如，$v[i]*x+y$ 等。

（4） 进程名，表示网络协议中的具体的进程，用大写字母或者字符串表示。例如，字符串 BGP 可以表示网络协议 BGP 的进程名。

（5） 进程的变量，由大写字母 X,Y,Z 表示进程的变量。

（6） 进程的方程式，一般形式如 PROCESS□ P ,其中 PROCESS 是一个进程的名字，而等号后面的 P 是进程的表达式， P 定义了进程的具体行为。

例如，一个收集数据的简单协议 protocol，则可能由发送者进程 sender 和接收者进程 receiver 所组成[44]。如图 3.3 所示，Sender具有通信信道input和wire，它由input信道向数据源接收数据，然后通过信道wire发出数据。

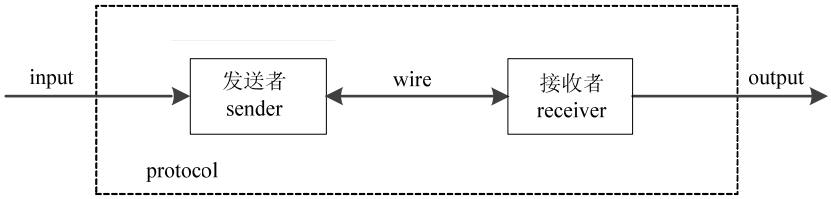

图 3.3 简单协议的收发进程

本文在对网络协议进行描述之前，还对建模的过程中使用的数据类型、信道名字等做了一些定义。

万方数据

定义 1：为方便描述网络协议交互过程，我们需要定义网络协议交互过程中用到的数据类型，这些数据类型包括主机的名字、协议状态集合。如表 3.1 所示的基本数据类型，是本文 CSP 模型中的常用数据类型，包括变量 ProtocolState、PacketType、Hosts、Port 等，变量的意义分别表示该变量在模型中的语义定义和作用。

表 3.1 基本数据类型

基本数据类型名字	变量名字	变量的意义
datatype	ProtocolState={…}	协议的状态集合
datatype	PacketType={…}	不同网络协议的报文类型
nametype	Hosts={Alice，Bob,…}	主机的名字
nametype	Port={0,1,2}	端口号
nametype	Param	参数

例如，关于 PacketType 和 ProtocolState 的例子如下所示。

（1） 定义 datatype PacketType 为不同网络协议的报文类型。例如，

datatype PacketType = ECHO_REPLY|DISTINATION_UNREACHABLE|
SOURCE_QUENCH|REDIRECT|ECHO_REQUEST|
TIMESTAMP_REQUEST|.....

表示 ICMP 协议的几种报文，其中 ECHO_REPLY 代表一个回应消息报文类型。

（2） 定义 datatype ProtocolState 为状态的集合。例如，

datatype BgpState = Idle|Connect|Active|OpenSent|OpenConfirm|Established

表示 BGP 协议的 6 种状态。

定义 2：定义基本类型 Channel 表示 CSP 信道集合，因此可以把在一个 CSP 信道 a 上通信的数据对象 d 记作复合事件 a.d。

在 CSP 模型中，信道的名字可以根据所处的局部环境定义。网络协议中的可信主体，在通信中通常存在三种通信信道：env、send 和 receive，如表 3.2 所示。其中，信道 env 表示通信的初始化，它用于协议主体和局部环境之间的通信。信道 send 和 receive 是最常用的发送消息和接收消息的信道，它用于可信主体之间或者可信主体与环境或者服务器之间通信的主要信道形式。

表 3.2 基本类型 Channel

基本数据类型名字	变量名字	变量的意义
Channel	receive	接收消息的信道
Channel	send	发送消息的信道
Channel	chU	抽象信道

万方数据

例如，信道 transport 表示传输层向下层网络层传输的通信信道，可以代表主机由传输层发起的一个事件，或者接收到一条命令。

channel transport:Transport_Addr.Transport_Data

定义 3：进程间的通信就是一个进程经某一信道发出一个数据，同时另一个进程从这一个信道上接收一个数据。因此，用c!e表示从信道 c 上输出一个数据 e。

（1） c!e，其中！表示从信道 c 上输出一个数据 e。

（2） c?x:M，其中？表示在信道 c 上接收属于数据类型 M 的基本数据 x。且信道 c 只能根据信道的传输特性，预测将接收到的数据所属于的数据类型。

（3） $(c?x:M \to P)$表示进程在信道 c 上收到数据 x，数据 x 属于类型为 M 的数据，并且进程下一步执行动作 P。

（4） $(c?x:M \to P)\square(d?y:N \to Q)$，表示如果进程在信道 c 上接收到数据类型为 M 的数据 x 后，下一步就执行 P 进程的动作；如果进程在信道 d 上接收到数据类型为 N 的数据 y 后，下一步就执行 Q 进程的动作；如果进程同时从信道 c 或信道 d 上接收到相应类型的数据时，则可任选一个信道接收数据，然后按照相应的 P 或者 Q 继续执行进程：如果进程既没有从信道 c 接收到数据，也没有从信道 d 接收到数据，那么进程不执行任何动作或者行为。

CSP 采用顺序程序设计中引入和控制非确定性而提出的卫氏命令和非确定性选择算子。

定义 4：定义 Message 表示消息。在 CSP 模型中，用事件表示消息，如果消息的定义不同，则需要专门定义不同的事件。在 CSP 模型中，用事件表示消息。如果消息的定义不同，则需要专门定义不同的事件。事件通常有着一般形式：Channel.Agent.Agent.Message。其中，Channel 表示信道，Agent 表示代理，Message 表示消息体。

在 CSP 模型中，信道的名字可以根据所处的局部环境定义。网络协议中的可信主体，在通信中通常存在三种通信信道：env、send 和 receive。其中，信道 env 表示通信的初始化，它用于协议主体和局部环境之间的通信。信道 send 和 receive 是最常用的发送消息和接收消息的信道，它用于可信主体之间或者可信主体与环境或者服务器之间通信的主要信道形式。因此，在 CSP 模型中，通常有以下三种事件形式，如表 3.3 所示。

表 3.3 事件的三种常用形式

事件形式	事件的意义
env.A.B	表示初始事件，主体 A 选择会话对象 B；
send.A.B.m	表示主体 A 给主体 B 发送消息 m;
receive.A.B.m	表示主体 B 接收主体 A 发送的消息 m。

万方数据

定义 5：CSP 进程主要通过描述协议实体的内部行为来定义协议实体的 CSP。定义通过协议名称、主机的名字和协议的状态建立 CSP 进程。

例如，BGP 协议的 CSP 描述如图 3.4 所示：

```
BGP_STA(id,init_state)=
    let otherhost=diff(Hosts,{id})    //BGP进程的初始化，包括初始化状态和为其他主机命名。

    BGP(id,state,param)=
          state==Idle & (              //BGP协议的状态为Idle时。
             (cmd?x?y?z—>
             (if(x==Start)
                then (OpenCon!StartTcpConnection.id.z—>BGP(id,Connect,z)      //发起建立TCP连接。
              else if(x==Ready)then BGP(id,Active,param)
              else BGP(id,Idle,param))))
            □
          state==Connect &(
          (OpenCon?x?y?z—>(
          if(x==TcpConnected and member(y,otherhost) and z==id)
          then
          (tcp?y.id.OPEN. correct—>tcp!id.y.OPEN.correct—>
           get!TcpConnected!id—>BGP(id,OpenConfirm,y)

          □
          state==Active & (
          (OpenCon?x?y?z—>(
          if(x==StartTcpConnection and z==id and member(y,otherhost))
          then …
```

图 3.4 BGP 协议的部分 CSP 描述

在本文中，我们给出的建模对象是各层的网络通信协议实体。不妨把协议的功能实体抽象为主机号，或者由主机号和端口号的组成。其中 id 代表主机的名字，如 Alice 或者 Bob; state 代表 id 主机的状态，例如 BGP 协议的状态集合为 $\{Idle, Connect, Active, Keepalive, ...\}$，param 表示网络协议通信对方的 id 号。

3.3 网络存储隐蔽信道攻击者建模

根据模型参考 Dolev-Yao 模型，构建网络存储隐蔽信道攻击者。在 Dolev-Yao 模型中，一个攻击者有发送任意报文、阻止报文的传递以及修改报文的能力。在 TCP 协议中，我们假设恶意攻击者可以监听网络协议建立连接的过程，可以把隐蔽消息嵌入到报文中，完成隐蔽消息的传递。我们把网络存储隐蔽信道当作攻击者建模为：隐藏在传输层的恶意程序 C 和 D，可以监听 TCP 连接，可以修改 TCP 首部中的字段，可以把隐蔽消息嵌入到字段中。

3.3.1 网络协议首部字段的分类

在构建网络存储隐蔽信道，把 TCP/IP 协议当作载体时，涉及对 TCP/IP 协议的头部字段进行修改，此时需要从以下几个方面考虑：

1）必须保证修改后的协议数据包能正常传输，因为每个网络协议头部不同区

万方数据

域的作用不同，因此不能对协议头部做任意的随意修改；

2)保证网络数据能够正常到达目的地,因为头部区域有目的端口之类的区域，如果达到了数据隐藏和存储式隐蔽类通道却没有达到目的地也是不可行的；

3）考虑到目前网络上广泛使用的包过滤防火墙，在构造隐蔽信道的时候，尽量使修改后的数据传输不会被这些安全机制影响。

综合这三个方面的考虑，为每个首部定义一个属性，命名为修改属性。头字段可以根据该属性分为三种类型。

- 安全字段：该种类型的字段属于不能做任何修改的字段，因为对他们的修改将会影响正常通信。所以这些字段被定义为安全字段。例如，TCP 首部的源端口和目的端口字段属于安全字段。一旦修改 TCP 首部的源端口号和目的端口号，就不能建立正常的 TCP 连接。
- 第 I 类隐蔽漏洞字段：该类型的字段需要特别的设置以便在某些情况下保证正常通信，所以它们的修改有一些限制。例如，TCP 协议头部的 urgent point 域在不需要传输紧急控制位的时候，是可以修改 TCP 头部的 urgent point 来传递消息的。
- 第 II 类隐蔽漏洞字段：这些字段可以任意修改，因为它们的修改通常不会影响正常通信。例如，网络协议中的预留字段是为未来改进协议的性质而预留的字段。

在此我们根据修改属性的特点，定义一个概念叫做隐蔽漏洞，即网络协议中可被利用来传输隐蔽信息的首部字段。第 I 类隐蔽漏洞字段和第 II 类隐蔽漏洞字段都属于隐蔽漏洞。

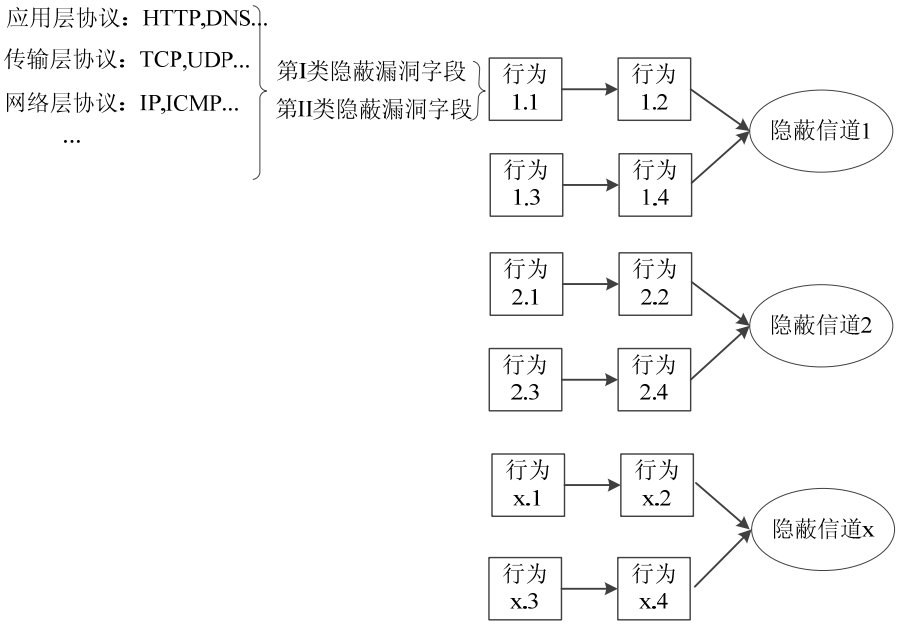

图 3.5 网络存储隐蔽信道的攻击规则图

如图 3.5 所示，描述了网络存储隐蔽信道的构建规则，通过规则将网络存储

万方数据

隐蔽信道的构建方式与网络协议资源建立一一对应关系。将攻击行为与攻击行为之间建立推理关系。

3.3.2 网络存储隐蔽信道攻击者的形式化建模

网络存储隐蔽信道攻击者针对某一特定的网络协议，根据协议实体参与者的行为和往来信道中的数据传输，分析得到攻击者可采用的网络存储隐蔽信息嵌入行为记忆能够获取的知识集合。

（1） 网络存储隐蔽信道攻击者初始知识库

CSP 形式化语言为每一个协议模型建立一个网络隐蔽信道初始知识库（Initial Knowledge）。该知识库用来记录攻击者掌握的信息，这些信息包括网络隐蔽信道传输机制、编解码机制、密钥、目标 ID、隐蔽信息等。建立攻击者的初始知识库能够更好的检测和分析网络协议中的存储隐蔽信道。

（2） 网络存储隐蔽信道攻击者建模

在定义网络存储隐蔽信道攻击者时，本文假设攻击者的构建网络存储隐蔽信道的能力足够的强大，任何经过网络传输的信息都可以监听、窃取。任何报文消息的隐蔽漏洞部分都可以被嵌入隐蔽信息。通过对网络协议交互实体模型的进程分析，就能获取对假设攻击者所应该具有的行为的有用信息。一个被动的网络攻击者可实现在线窃听敏感信息，而一个主动攻击者可截获数据包，并对其进行任意的修改，甚至可以伪装成通信主体。

网络存储隐蔽信道攻击者的行为一般表现为以下几种形式：

（1） 监听网络协议。

（2） 将网络协议报文中的首部字段篡改，嵌入隐蔽信息。

（3） 改变部分或全部消息的目的地址。

（4） 将消息发送出去。

（5） 对隐蔽信息进行编解码。

网络存储式隐蔽信道的特性是保证网上通信数据的秘密性，因此对攻击者攻击能力的假设是十分重要的。我们假设网络存储隐蔽信道，即在一群通信主体中包含一些被假设为具有一定共享构建存储式隐蔽类信道知识的主体，可用于传输和构建网络存储隐蔽信道。

3.4 模型检测

本文采用模型检测工具 FDR 分析网络存储隐蔽信道模型。模型检测是一种可以采用形式化方法的自动验证技术。可以用 FDR 得到 CSP 中所描述的并发反应

万方数据

系统的结果。

FDR(Failures-Divergence Refinement)，即故障-偏差提炼检测器，是用于 CSP 的模型检测器。这个检测器可以通过检测系统的状态，用来测试一个系统提炼后是否满足所要验证的性质。FDR 检测器首先将两个 CSP 进程作为输入进程，这两个进程包括规约（Specification）进程和实现（Implementation）进程，其分析过程就是检查实现进程是否是规约进程的提炼或者称为精化。

FDR 模型检测器对网络协议的检测是基于协议规范的正确性说明以及协议的一致性性质。

3.5 本章小结

本章在对网络协议和网络存储隐蔽信道的语义正确性的研究前提下，提出一种基于 CSP 的通用检测和分析模型。该模型从网络协议首部分类、网络存储隐蔽信道建模、隐蔽通信实体的构建等几个方面阐述了该模型的具体实现细节。

万方数据

第四章 模型的实现和验证

在本章，我们通过设定 TCP 协议中网络存储隐蔽信道攻击者的假设，针对 TCP 协议和其中的网络存储隐蔽信道攻击者建立 CSP 模型，验证基于 CSP 模型的有效性和可行性。

4.1 TCP协议的CSP模型

4.1.1 TCP协议模型简介

TCP 协议交互实体图，如图 4.1 所示。该实体图包括：两个主机，通信信道。为简化分析，只分析两个主机的应用层和传输层。应用层与传输层协议 TCP 之间通过 cmd 信道进行消息的传递和通信，TCP 协议通过信道 out 和 in 来建立 TCP 连接。建立好之后，它们通过 out 和 in 信道继续进行报文的传递。

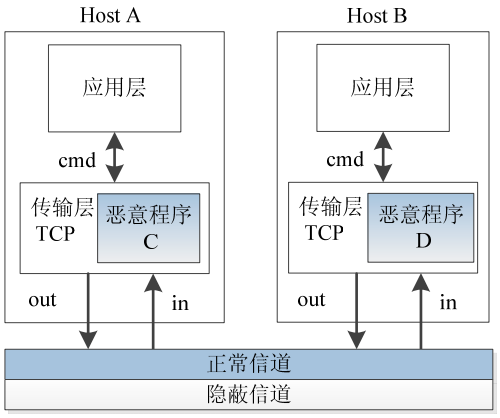

图 4.1 TCP 交互实体图

在本文，主要侧重于 TCP 协议本身，因此我们假设主机处于理想的情况下，没有对 TCP/IP 协议的其他层进行相关的细粒度建模分析。我们假设位于传输层的恶意程序 C 和恶意程序 D 为网络存储隐蔽信道的发送者和接受者，作为网络存储隐蔽信道模型。恶意程序 C 和恶意程序 D 是两个恶意攻击者，负责监听 TCP 连接的动态信息，发动网络存储隐蔽信道攻击。最终的 TCP 协议交互系统是由两个通信终端与两个恶意攻击者之间的进程并行运行。

为了方便使用 CSP 语言进行形式化建模和 FDR 检测工具进行验证，本文所有的模型代码都满足于 FDR 规范。

下面分别从 TCP 协议分析、协议头部分类、协议模型介绍三个方面进行详细

万方数据

的说明。

4.1.2 TCP协议分析

TCP 协议是在计算机通信网络中可靠的端对端的传输层协议。它提供了进程间可靠的、面向连接的通信。

在本文中，为了简化对 TCP 协议的分析和描述，我们使用简化的有限状态机模型来表示 TCP 协议的状态转移过程，如图 4.2 所示。

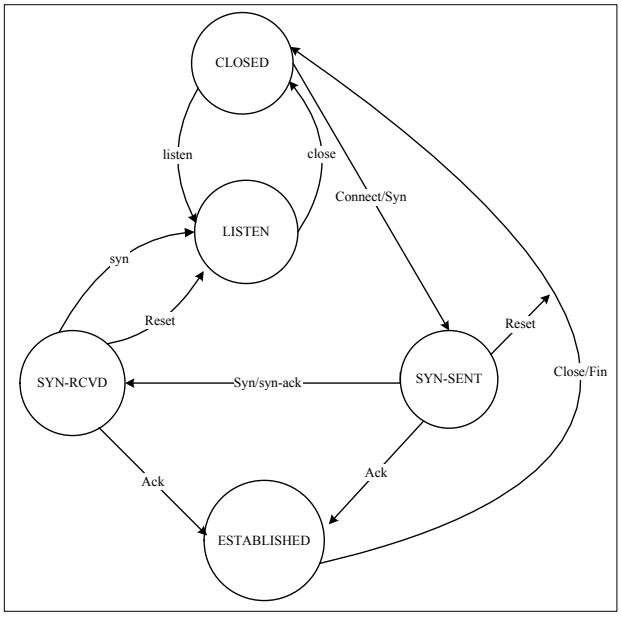

图 4.2 简化的 TCP 有限状态图

简化的 TCP 有限状态机忽略具体细节和结束连接的状态，只包含常用的 5 种状态。有限状态图的起始状态是 CLOSED 状态，当主机正在等待进来的连接请求时，即执行 listen 原语事件时，主机就进入 LISTEN 状态。当主机收到客户发送的 SYN(SEQ=x)数据包，主机就进入 SYN-RCVD 状态。如果 TCP 三次握手建立成功，有限状态图就进入 ESTABLISHED 状态，开始正常的传递数据。值得一提的是，在客户机和服务器之间传输的所有数据都可以看作一次步骤。简化的有限状态机中用到的状态的简单说明如表 4.1 所示。

表 4.1 TCP 有限状态机中用到的状态

万方数据

状态	说明
CLOSED	没有活动的连接，或者未完成的连接
LISTEN	服务器正在等待进来的连接请求
SYN-RCVD	一个连接请求已经到达，等待 ACK
SYN-SENT	应用程序已经开始打开连接
ESTABLISHED	正常的数据传输状态

客户端-主机 A 和服务器端-主机 B 发起建立一条 TCP 连接，如图 4.3 所示 TCP 三次握手的建立过程。

第一次握手：主机 A 发送序列位码为 SYN = 1，随机产生 seq number = x 的数据包到服务器，主机 B 由 SYN = 1 知道，A 要求建立联机；

第二次握手：主机 B 收到请求后要确认联机信息，向 A 发送 ack number = (主机 A 的 seq+1)，SYN = 1，ACK = 1，随机产生 seq number = y 的包。

第三次握手：主机 A 收到后检查 ack number 是否正确，即第一次发送的 seq number+1，以及位码 ACK 是不是为 1。若正确，主机 A 会再发送 ack number =(主机 B 的 seq + 1)，ACK = 1，主机 B 收到后确认 seq 值与 ACK = 1 则连接建立成功。

完成三次握手，主机 A 与主机 B 开始传送数据。

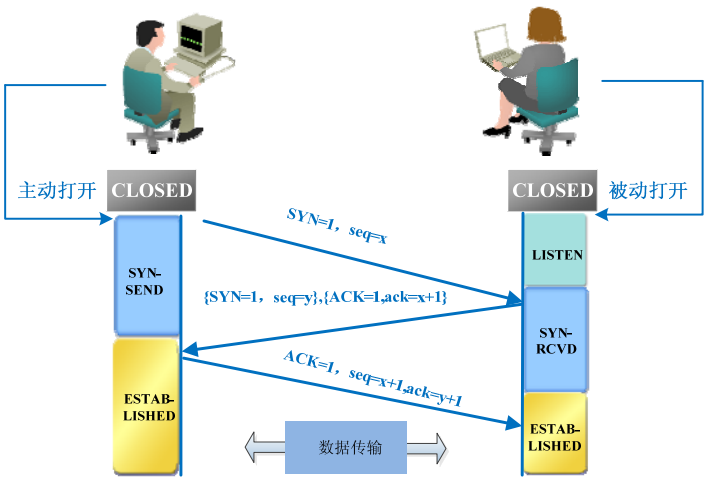

图 4.3 TCP 三次握手过程

万方数据

4.1.3 TCP首部的分类

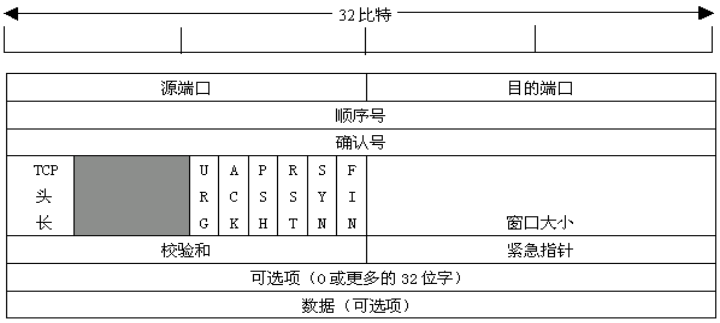

图 4.4 TCP 的首部字段

如图 4.4 所示，TCP 协议首部字段的 20 个字节包含 14 个固有字段。根据 4.2.1 对首部的分类规则，我们把 TCP 协议头部分为安全首部、第 I 类隐蔽漏洞域、第 II 类隐蔽漏洞域三种类型的首部，如表 4.2 所示。

表 4.2 TCP 首部字段的分类

类型	编号	TCP 首部字段
安全字段	1	Source port
	2	Destination port
	5	TCP header length
	7	URG
	8	ACK
	9	PSH
	10	RST
	11	SYN
	12	FIN
	13	Window size
第 I 类隐蔽漏洞域	3	SeqNo
	4	AckNo
	14	checksum
	15	Urg_p
第 II 类隐蔽漏洞域	6	Reserve field(6 bit)

上面对 TCP 协议首部字段的分类只是考虑最理想的状态，黑客设计隐蔽信道的方法巧妙，比如 TCP 的确认号域 ACK 在文献就被提出可以构造隐蔽信道。

1、填充 31 位/分组。它的长度取决于 TCP 头中可选项字段的长度，它是 31 位或 8 位/分组。这个领域需要在合法的 TCP 处理程序之前从填充字段中提取隐蔽数据。

万方数据

2、初始序列号 ISN32 位/连接。TCP 采用“三次握手”协议谈判进程。ISN 服务作为一个完美的媒介，在包损失的情况下以确保可靠的服务。在此方法中，发送者产生一个对应到实际的隐蔽数据的 ISN。隐蔽接收机提取这一领域，并没有给它一个 ACK。隐蔽发件人不断发送相同的数据包不同的隐蔽数据的 ISN 嵌入式。这是最简单的形式放置隐蔽的数据使用这一领域。

3、TCP 应答序列号字段-32bits/连接-弹跳。这种方法是基于 IP 地址欺骗。发送机的数据包被退回的远程服务器。考虑一个远程服务器 A 和通信系统之间的方案:系统 1 和系统 2。系统 1 和系统 2 通过在服务器系统 A 上的弹跳实现通信。ACK 是 TCP 头部的 4 个字节。假设整个消息被划分成 5 块。预先计算在 1 号系统确认，这些数据块被发送到远程服务器 A 把 2 号系统作为其一个源地址。远程服务器 A 的弹跳不会产生新的 ACK 号。在另一方重组，通道可以被恢复。大部分时间，此方法创建可疑的数据包，并导致被检测到。

4、标志位的操作和使用保留字段。TCP 提供端到端可靠的信息交付，因此，这种面向连接的协议，使用 6 位标志字段，也称为“代码位”。这些标志位 URG，ACK，PSH，RST，SYN 和 FIN，说明 TCP 包（TPDU 的）如何被处理。 6 位有 64 个可能的组合，其中 29 个组合都是有效的。因此，余下的组合，可用于发送秘密数据。在图 3 中的 TCP 头预留的领域也可以用来放置隐蔽的数据。

基于对 TCP 首部的分类分析和 TCP 协议规范的分析，我们编写进程 C _exploit(X) 分析 TCP 协议中存在的存储隐蔽漏洞。

4.1.4 TCP协议模型中的基本数据类型与通信信道

我们假设一个主机集合命名为 Hosts，包括所有主机，在这里包括 Alice 和 Bob，简称 A 和 B。为简化分析，端口号不妨设只有三个，命名为 0，1，2。并且描述各种报文类型的集合为 PacketTypes，各种报文的集合为 Packet，Tstate 表示状态的集合。

主机名字为 $\text{datatype Host} = \text{Alice} \mid \text{Bob}$

$\text{nametype Port} = \{0..2\}$

$\text{datatype Connection\_Data} = \text{bit}.\{0,1\}$

则两台主机可以建立起来的连接为：

$$\text{Connections} = \{(Alice.0, Bob.1), (Bob.0, Alice.1), (Alice.2, Bob.2)\}$$

传输层地址数据类型如下式所示：

万方数据

nametype Transport_Addr = Host.Direction

传输层传输的数据类型为：

nametype Transport_Data = Connection_Addr.Network_Data

信道：

channel transport:Transport_Addr.Transport_Data

data type PacketTypes = SYN | SYN_ACK | ACK | RST | FIN | RSH.Command

分别代表 TCP 连接中的六个常见报文类型。每个数据包包含源地址、目的地址和数据。报文以以下的方式表示：

$$Packet = \{s.d.m \mid s \in Hosts, d \in Hosts, m \in PacketTypes\}$$

TCP 层通过 cmd 信道和上层应用层通信。信道 cmd 代表来自应用层的命令，这些命令包括连接其他主机、监听、在建立连接之后获得服务、返回结果。

datatype TcpCmd = listen|connect.h|close.h|connErr.h|connected.h|rsh.c.h|
exec_rsh.c $h \in Hosts, c \in Command$

channel cmd:TcpCmd

其中 TcpCmd 表示在主机中应用层协议与传输层 TCP 协议交互的一些命令的集合，rsh.c.h 是一个事件请求，该事件请求执行参数为 h 的 c 命令。exec_rsh.c 是一个事件请求在服务器中执行命令 c。

综上所述，TCP 的 CSP 进程如图 4.5 所示：

```
TCP(id, init_state)=
    let  otherhost =diff(Hosts,{id})    //TCP协议进程的初始化，包括初始化状态和为其他主机命名。

 TCP_SM(id,state,param)=
      state==CLOSED &(                  //TCP协议状态为closed时。
       (cmd.connect?x:otherhost→
       if(x==connect)
       then (out!id.x.SYN→TCP_SM(id,SYN_SENT,x))
    □(cmd.listen→TCP_SM(id,LISTEN,param))
 □
    state==LISTEN&(
       （in?x.y.z→
        if((y==id)&(z==SYN))
        then (state=SYN) →TCP_SM(id,SYN_SENT, param)
        else (out!error) →TCP_SM(id,LISTEN,param)
 □
     state==SYN_SENT&(
       （in?x.y.z→
        if((y==id))&(z==SYN_ACK))
        then (out!x.id.ACK→TCP_SM(id,ESTABLISHED,param))
        else(out!error) →TCP_SM(id,SYN_SENT,param)
 □
     state==ESTABLISHED& …
```

图 4.5 TCP 协议的部分 CSP 模型

在图 4.5 中，id 表示主机的代号。例如，A 或者 B，initi_state 表示 id 主机的

万方数据

初始状态。Param 表示通信对方的 id。

假设网络中只有两个 TCP 协议实体，一个为 TCP 连接发起者：客户端主机 A，一个为 TCP 连接响应者：服务器端主机 B。不妨假设 TCP 协议中存在隐蔽通信的可能是：恶意程序 C 和恶意程序 D 分别寄存在宿主主机 A 和 B 上，C 和 D 事先约定好，通过主机 A 和主机 B 的 TCP 连接来传递隐蔽信息。

主机模型可以表示如图 4.6 所示：

```
CLIENT(i)=
            cmd!connect.B→cmd?result?x
    →if(result==connected)
    then cmd.rsh.AnyCmd.A
    →cmd!close.B→CLIENT(i)
    else CLIENT(i)
SERVER(i)=(cmd.connected?x→cmd.exec_rsh?x
            →cmd!listen→SERVER(i))
            □
            (cmd.connErr?x→cmd!listen→SERVER(i))
HOST(A)=
    (TCP(A,CLOSED))|[{|cmd|}]|CLIENT(A))
HOST(B)=
    (TCP(B,LISTEN))|[{|cmd|}]|SERVER(B))
```

图 4.6 主机的 CSP 模型

为采用 CSP 语言描述 TCP 有限状态机的运作机制，建立三次握手的 TCP 连接都是从一个“被动打开”操作开始，此时的服务器端 TCP 有限状态机处于“听”（LISTEN）的状态，直到有一个客户机发起“主动打开”与它联络，它会发送一个SYN 报文段给服务其，并进入“SYN 已发出”（SYN-SENT）状态。有限状态机从最初的CLOSED 状态迁移，在每一种操作的触发下改变状态，每个状态对应着一些特定的原语操作，可以映射为 CSP 语言中的一些可见事件。定义一个 Tstate 表示状态的集合，即：

$$Tstate = \{CLOSE,LISTEN,SYN\text{-}SENT,SYN\text{-}REC,ESTABLISHED\}$$

我们假设一个主机 A 和主机 B 之间通信的数据包类型集合命名为 PacketTypes。

$$datatype\ \ PacketTypes = SYN \mid SYN\_ACK \mid ACK \mid RST \mid FIN \mid RSH.Command$$

分别代表 TCP 连接中的五个常见报文类型。每个数据包包含源地址、目的地址和数据。报文以以下的方式表示：

万方数据

datatype Packet = {s.d.m|s ∈ Hosts,d ∈ Hosts,m ∈ PacketTypes}

4.1.5 隐蔽存储信道-攻击者模型

在 Dolev-Yao 模型中，一个攻击者有发送任意报文、阻止报文的传递以及修改报文的能力。在 TCP 协议中，我们假设恶意攻击者 C 和 D 可以监听 TCP 连接的建立，可以把隐蔽消息嵌入到报文中,完成隐蔽消息的传递。我们把网络存储隐蔽信道当作攻击者建模为：隐藏在传输层的恶意程序 C 和 D，可以监听 TCP 连接，可以修改 TCP 首部中的字段，可以把隐蔽消息嵌入到字段中。

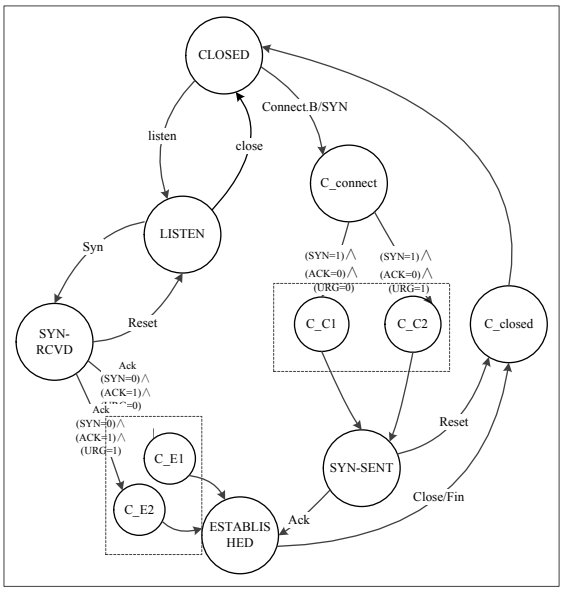

图 4.7 TCP 协议交互系统状态图

对隐蔽信道攻击者攻击能力的假设要考虑到它们的攻击能力。保证传输信息的隐蔽性是网络存储隐蔽信道的最大特征。根据前面对隐蔽信道特点的分析，可知网络存储隐蔽信道攻击者的行为一般表现为以下几种形式：

（1） 监听网络协议。

（2） 将网络协议报文中的首部字段篡改，嵌入隐蔽信息。

（3） 改变部分或全部消息的目的地址。

（4） 将消息发送出去。

（5） 对隐蔽信息进行编解码。

基于网络存储隐蔽信道攻击者的行为，本文对网络存储隐蔽信道攻击者进程的状态进行抽象，定义为状态集合 Cstate：

Cstate = {C_connect,C_closed,C_E1,C_E2,C_C1,C_C2}

其中 C_connect 代表等待建立网络存储隐蔽信道，C_closed 表示网络存储隐

万方数据

蔽信道关闭，C_E1 代表创建类型 1 的网络存储隐蔽信道等。

网络存储隐蔽信道攻击者进程如图 4.8 所示。

```
ATTACKER(id, mes)=|~|x:Hosts,y:Hosts,x!=y, m:Packet,d—>
                        mes,z:PacketState
                        @
                          (out?x?y?m?z→ATTACKER(union(mes,{m})))
                            □
                          (in!x.y.d.incorrect→ATTACKER(id,mes))
```

图 4.8 攻击者的形式化进程

网络存储隐蔽信道攻击者的 CSP 描述如图 4.8 所示。网络存储隐蔽信道嵌入隐蔽信息的协议载体资源表如表 4.3 所示。

表 4.3 协议载体资源表

协议字段	数据类型	说明
SeqNo	Int(32)	第 I 类隐蔽漏洞字段
AckNo	Int(32)	第 I 类隐蔽漏洞字段
Urg_p	Int(1)	第 I 类隐蔽漏洞字段
Resvd	Int(6)	第 II 类隐蔽漏洞字段

在模型中我们假设主机之间发送的报文都是正确的，而经过网络存储隐蔽信道攻击者发送的报文都是经过修改的，有误的。|~|表示内部选择。

最终的模型为两个主体与两个攻击者之间的并行，如图 4.9 所示。

```
System= (HOST(A)|||ATTACKER(C,mes)|||
                ATTACKER(D, mes)|||HOST(B))|[in，out]|
```

图 4.9 协议交互系统的 CSP 描述

该模型隐藏了通信的内部通道。主机 A 和主机 B 之间通过信道 in 和 out 信道来建立连接，攻击者通过隐藏在传输层，监听主机 A 和主机 B 通信的动态。

安全属性是系统所要满足的约束。本文中，我们验证模型的一个性质。

$$Spe = starting \rightarrow connected \rightarrow Sp$$

$$Sp = processing \rightarrow Sp$$

$$Covert = CHAOS(diff(Events, \{out.covert\_msg.B, in.seqNo = covert\_msg.A, ...\}))$$

CHAOS 是一个非确定性的，不发散的进程，可以表示所有事件的任意序列。Covert 表示 TCP 交互系统的事件序列应该不包含 covert_msg 相关事件。

万方数据

4.2 FDR检测结果分析

在本小节，我们分析和验证网络存储隐蔽信道的存在通过使用 FDR 检测工具，且使用 FDR 检测工具中的迹模型提炼关系。

FDR(Failures-Divergence Refinement，故障偏差精炼检查器)是基于 CSP 的一个模型检测工具，通过检测系统的状态，用来测试一个系统的 CSP 模型是否满足所要验证的安全性质。FDR 支持 CSP 的几种模型：traces model、stable failures model、failures/divergences model、refusal testing model、revivals model 等。在检测过程中，对不满足描述的例子，将会给出一个反例。在本文中我们使用 CSP 的 traces model。

TCP 协议的 CSP 模型加载到 FDR2 后的界面如图 4.10 所示：

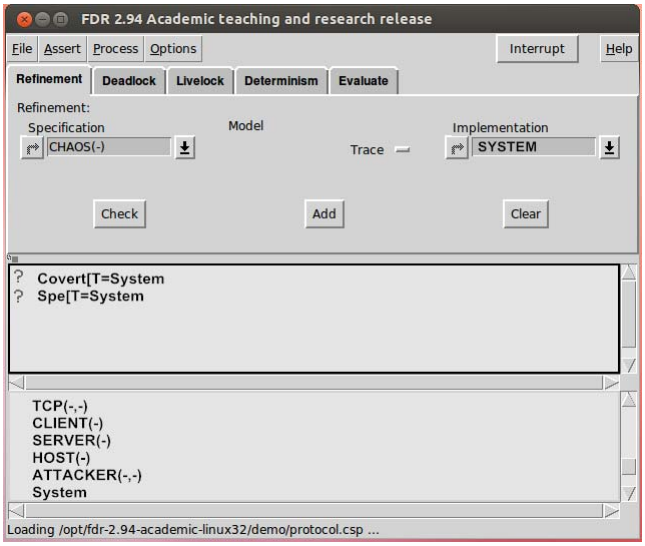

图 4.10 FDR2 加载 CSP 程序图

如图 4.10 所示，在 FDR 模型检测器中，对几个进程主体 TCP(-,-), CLIENT(-), SERVER(-), HOST(-)等，对 Covert 和 Spe 两个说明进程进行检测的过程。

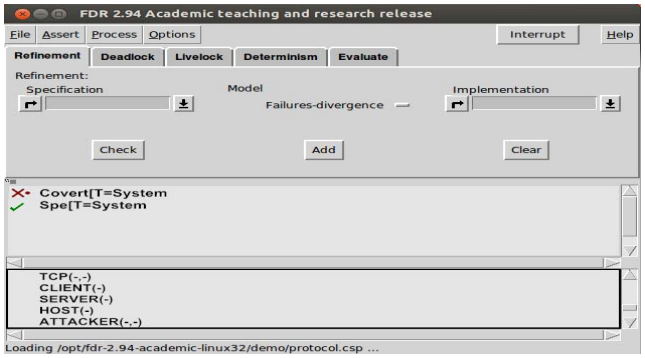

图 4.11 FDR2 提炼验证结果

FDR2 提炼得到验证结果如，如图 4.11 所示。从验证结果图中，可以看出，对于第一个提炼模型，红叉号代表我们得到了反例。对于第二个提炼模型，是对

万方数据

号，表示满足安全性质，没有得到反例。第一个提炼模型得到的反例结果如下图 4.12 所示。

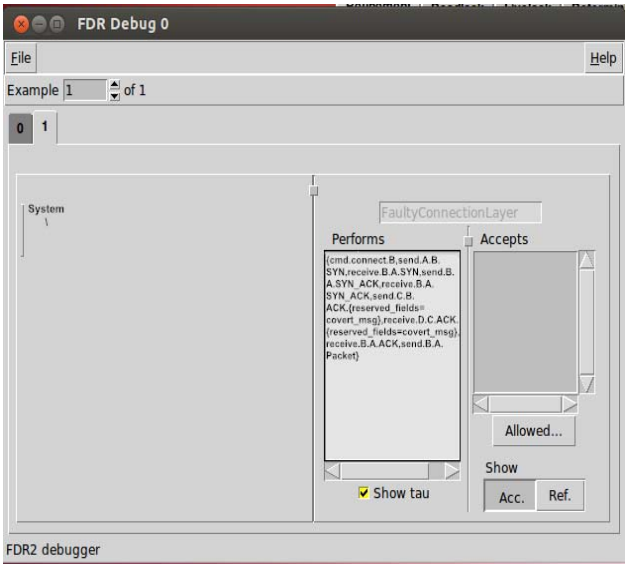

图 4.12 FDR 提炼的反例

图 4.12 中包含 TCP 协议交互系统不满足第一个提炼模型性质时的反例示例。图中所示的一条反例序列为：

$<$ cmd.connect.B,send.A.B.SYN,
receive.B.A.SYN_ACK,send.A.B.ACK,
receive.B.A.RSH,send.A.B.RSH,
receive.A.C.RSH.$(\text{urgent\_pointer} = \text{covert\_msg})$,
send.C.B.RSH.$(\text{urgent\_pointer} = \text{covert\_msg})$,
receive.C.B.RSH.$(\text{urgent\_pointer} = \text{covert\_msg})$,
receive.A.B.RSH,... $>$

该序列表示 TCP 建立连接和传输信息的过程中，TCP 协议头部字段被攻击者利用，构建网络存储隐蔽信道。我们在建立模型时，假设环境是不安全的，所以攻击者可以任意发送所学到的报文，以及可以修改报文，所以最终 TCP 协议被用来传递存储隐蔽信息。

FDR 得到的反例序列对应的实际运行过程如图 4.13 所示：

万方数据

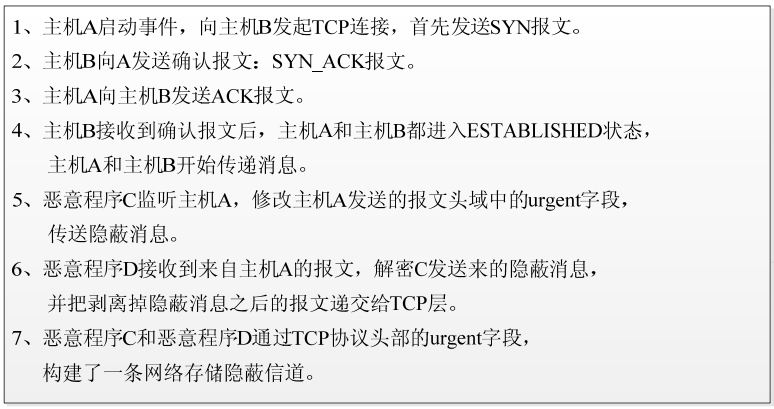

图 4.13 FDR 提炼的反例

由图 4.12 得到的另一条反例序列为：

<cmd.connect.B,send.A.B.SYN,CovertCprocessing,
send.A.B.SYN.(reserved_fields=covert_msg),
receive.B.A.SYN_ACK.(reserved_fields=covert_msg),
CovertCprocessing,
receive.B.A.SYN_ACK,send.A.B.ACK,CovertCprocessing,
send.A.B.ACK.(reserved_fields=covert_msg),...>

该条反例序列对应的实际运行过程是恶意程序 C 和 D 利用 TCP 协议头部的保留字段 reserved fields 构建了一条隐蔽信道，传递信息。利用 reserved fields 构建的网络存储隐蔽信道最早由 Handel 提出。

基于 CSP 的网络存储隐蔽信道检测和分析框架在 TCP 协议上的验证，检测和分析出很多前人已经发现的存储隐蔽信道，并且分析出 TCP 协议头部的隐蔽漏洞。本章对 TCP 协议中的网络存储隐蔽信道的分析和验证，证明该基于 CSP 的形式化分析方法是有效的。

4.3 本章小结

在本章，我们通过设定 TCP 协议中网络存储隐蔽信道攻击者的假设，针对 TCP 协议和其中的网络存储隐蔽信道攻击者建立 CSP 形式化模型，最终发现了 TCP 协议中的反例，每条反例对应一条基于 TCP 协议的网络存储隐蔽信道。该模型在 TCP 协议的验证，证明了 CSP 形式化模型的有效性和可行性。

万方数据

第五章 总结

5.1 本文工作总结

随着信息安全和隐私保护越来越受国家和个人的重视，网络存储隐蔽信道作为信息泄露的一种方式，受到越来越多研究学者的重视。本文的研究即是网络存储隐蔽信道的形式化检测和分析方法，加强对网络存储隐蔽信道的形式化检测和分析研究。

具体而言，本文的研究成果主要有：

1)分析了网络存储隐蔽信道的特点，为网络协议首部定义属性和隐蔽漏洞的概念，并根据属性定义，把网络协议首部划分为三种类型。

2)提出一种基于 CSP 的网络存储隐蔽信道检测和分析通用模型，该模型对网络协议进行语义描述，定义网络协议通信系统和网络存储隐蔽信道模型中的事件与基本进程。

3)在研究和分析网络协议通信系统的语义正确性的基础上，利用 CSP 进程中迹模型的提炼定义以及 FDR 检测工具，提出通过检测进程之间的提炼关系来对网络存储隐蔽信道进行检测和分析的方法。

4）以 TCP 协议为例，研究基于 CSP 的网络存储隐蔽信道检测和分析模型在网络协议通信系统中的应用。该验证证明了 CSP 描述检测和分析方法的有效性和实用性。

5.2 进一步工作展望

本文提出的基于 CSP 的网络存储隐蔽信道检测和分析模型，在加强网络协议分析的语义描述和分析方面，以及自动化工具分析方面具有一定的优越性，但对模型的语义描述仍然有很多要完善和改进的地方，需在以下方面进一步工作：

1) 协议本身的模型不够完善，缺少对时间约束的描述忽略了网络协议中计时器的描述：因为基本的进程代数 CSP 不支持时间，将来会用基于 Timed-CSP 对协议进行更细粒度、更加完善的建模。

2) 对网络存储隐蔽信道的攻击者的能力假设过于简单，希望在未来能够把攻击者描述地更恰当、更准确的表现实际环境。

3) 模型的验证结果表现仍然不够直观，如何以更直观的方式表示模型的验证结果和反例结果也是一个值得研究的问题。

万方数据

由于作者时间和水平有限，论文中还存在诸多不足之处，敬请各位评审专家和读者批评指正。

万方数据

致谢

岁月如歌，光阴似箭，近两年多的硕士研究生学习生涯即将结束，经历了研究生阶段的学习、为人处世，以及找工作的坎坷与曲折，我深深体会到了研究生生活带给我的身心上的快速成长。回首两年半的求学历程，对那些引导我、帮助我、激励我的人们，我心中充满了感激之情。

本论文是在我的导师朱辉副教授的悉心指导之下完成的，在此论文完成之际向我的恩师朱老师表示最真挚的感谢，感谢朱辉老师近三年来对我的谆谆教导、培养与关怀。很荣幸能够成为朱辉老师的学生，朱老师学识渊博，治学严谨，为人谦逊平和，严于律己宽以待人，朴实无华、平易近人的人格魅力对我影响深远。近三年的相处中老师孜孜不倦的进行科学研究，教书育人，不仅使我学会了细心认真的做事做学问态度，更重要的是很多为人处世的道理。在此，谨向朱老师表示我最诚挚的敬意和感谢！

本论文的完成也离不开其他各位老师、同学和朋友的关心与帮助。感谢刘北水师兄在论文写作期间提出的宝贵意见，感谢同门的说师弟师妹们，在科研工作中给我很多帮助和鼓励。

还要感谢实验室的潘文海、何红等师兄师姐在学习和找工作过程中一直以来的指点迷津，从他们那里我学会了严谨认真的学习态度和集中精力做一件事情的好习惯，广阔的视野以及努力奋斗的重要性。

感谢张美珍老师一直以来给予的帮助和鼓励。感谢同年级的魏光辉、陈亮、刘爱花、陈屯等同学一直以来对我的帮助，愿友谊长存。

感谢舍友史高娃、魏文佳、周茹，和他们共同生活在一个屋檐下两年半，他们在我的生活和个人成长中给予热情的帮助和勉励，改变我的人生观念，提升我的幸福值，在此向他们表示深深的谢意。

感谢我的家人，感谢他们养育我长大成人，感谢他们在物质和精神上带给我的支持和鼓励。在面临人生选择的迷茫之际，为我排忧解难，他们对我无私的关爱与照顾是我不断前进的动力，他们给我的人生教诲让我受益匪浅。仅借此机会向所有给予我关心、爱护、支持和帮助的人们表示衷心的感谢。

万方数据

万方数据

参考文献

[1] M. Bauer. New covert channels in HTTP: adding unwitting Web browsers to anonymity sets[C]. Proceedings of the 2003 ACM workshop on Privacy in the electronic society. ACM, 2003:72-78.

[2] X. Wang, D.S. Reeves. Robust correlation of encrypted attack traffic through stepping stones by manipulation of interpacket delays[C]. Proceedings of the 10th ACM conference on Computer and communications security. ACM, 2003:20-29.

[3] X. Wang, S. Chen, S. Jajodia. Tracking anonymous peer-to-peer VoIP calls on the internet[C]. Proceedings of the 12th ACM conference on Computer and communications security. ACM, 2005: 81-91.

[4] D. V. Forte, C. Maruti, M. R. Vetturi, et al. SecSyslog: An approach to secure logging based on covert channels[C].Systematic Approaches to Digital Forensic Engineering, 2005. First International Workshop on. IEEE, 2005: 248-263.

[5] A. Singh, O. Nordström, C. Lu, et al. Malicious ICMP tunneling: Defense against the vulnerability[C]. Information Security and Privacy. Springer Berlin Heidelberg, 2003: 226-236.

[6] S. E. Schechter, M. D. Smith. Access for sale: A new class of worm[C]. Proceedings of the 2003 ACM workshop on Rapid malcode. ACM, 2003: 19-23.

[7] R. Rogers, M. Devost. Hacking a Terror Network: The Silence Threat of Covert Channels. Syngress, 2005.

[8] 2001 G B T. 信息技术安全技术信息技术安全性评估准则 [S][D]. 2001.

[9] U. S. Department of Defense. Trusted computer system evaluation criteria. DoD 5200.28-STD, 1985.

[10] ISO/IEC 15408. Information technology-security techniques-evaluation criteria for IT security. 1999.

[11] K. Ahsan, D. Kundur. Practical data hiding in TCP/IP[C]. Proc. ACM Workshop on Multimedia Security. 2002:1−8.

[12] T. G. Handel, M. T. Sandford. Hiding data in the OSI network model. Information Hiding[C]. 1996: 23−38.

[13] G. Fisk, M. Fisk, C. Papadopoulos. Eliminating steganography in Internet traffic with active wardens[C]. Information Hiding. Springer Berlin Heidelberg, 2003:18-35. http://portal.acm.org/citation.cfm?id=732023.

[14] A. Hintz. Covert channels in TCP and IP headers [J]. Presentation at DEFCON, 2002, 10.

[15] S. Cabuk, C. Brodley, C. Shields. IP covert timing channels: Design and detection [C]. Proceedings of the 11th ACM conference on Computer and communications security.USA: ACM, 2004: 178-187.

[16] S. Cabuk, C. E. Adviser-Brodley, E. H. Adviser-Spafford. Network Covert Channels: Design, Analysis, Detection, and Elimination [J]. 2006.

[17] G. Shah, A. Molina, M. Blaze. Keyboards and covert channels[C].Proceedings of the 15th conference on USENIX Security Symposium. 2006, 15: 5.

万方数据

[18] S. Gianvecchio, H. Wang, D. Wijesekera, et al. Model-based covert timing channels: Automated modeling and evasion[C]. Recent Advances in Intrusion Detection. Springer Berlin Heidelberg, 2008: 211-230.
[19] C. Wang, S. Ju. Searching covert channels by identifying malicious subjects in the time domain [C].Information Assurance Workshop, 2004. Proceedings from the Fifth Annual IEEE SMC. IEEE, 2004: 68-73.
[20] J. Zhai, G. Liu, Y. Dai. A covert channel detection algorithm based on TCP Markov model [C]. In: Proceedings of Second International Conference on Multimedia Information Networking and Security, 2010: 893-897.
[21] 刘红英.基于 HMM 的网络隐蔽信道检测模型的研究[D].南京理工大学.2011.
[22] B. W. Lampson. A note on the confinement problem [J]. Communications of the ACM, 1973, 16(10):613-115.
[23] M. Schaefer, B. Gold, R. Linde, et al. Program confinement in KVM/370[C]. Proceedings of the 1977 annual conference. ACM, 1977: 404~410.
[24] J. C. Huskamp. Covert communication channels in timesharing systems [J].Berkley USA: University of California, Berkeley. 1978.
[25] R. A. Kemmerer. Shared resource matrix methodology: An approach to identifying storage and timing channels [J]. ACM Transactions on Computer Systems (TOCS), 1983, 1(3): 256-277.
[26] C. R. Tsai, V. D. Gligor, C. S. Chandersekaran. A formal method for the identification of covert storage channels in source code[C]. Los Alamitos, California, USA: Symposium on Security and Privacy, 1990: 569~580.
[27] 谷传征.DNS 协议隐蔽信道的构建和检测技术研究[D]. 上海交通大学. 2012.
[28] 王育民,张彤,黄继武.信息隐藏-理论与技术[J].北京：清华大学出版社. 2005.
[29] G. J. Simmons. The Prisoners' problem and the subliminal channel [C].Advances in Cryptology. Springer US, 1984: 51-67.
[30] M. Wolf. Covert channels in LAN protocols [M]. Local Area Network Security. Springer Berlin Heidelberg, 1989:89-101.
[31] S. Craver. On public-key steganography in the presence of an active warden [C]. Information Hiding. Springer Berlin Heidelberg, 1998:355-368.
[32] S. B. Lipner. A comment on the confinement problem [J]. ACM SIGOPS Operating Systems Review, 1975, 9(5): 192-196.
[33] K. Ahsan. Covert channel analysis and data hiding in TCP/IP [D].Toronto: University of Toronto, 2002.
[34] C. H. Rowland. Covert channels in the TCP/IP protocol suite [J]. First Monday, 1997, 2(5).
[35] E. Tumoian, M. Anikeev. Network based detection of passive covert channels in TCP/IP[C]. Local Computer Networks, 2005. 30th Anniversary. The IEEE Conference on. IEEE, 2005: 802-809.
[36] T. Sohn, J. T. Seo. J. Moon. A study on the covert channel detection of TCP/IP

万方数据

header using support vector machine [M]. Information and Communications Security. Springer Berlin Heidelberg, 2003: 313-324.

[37] D. E. Denning. A lattice model of secure information flow [J]. Communications of the ACM, 1976, 19(5): 236-243.

[38] 卿斯汉, 朱继锋. 安胜安全操作系统的隐蔽信道分析[J].软件学报, 2004,15(9): 1385-1392.

[39] G. Lowe. Breaking and fixing the Needham-Schroeder public-key protocol using FDR [M]. Proceedings of TACAS, number 1055 in LNCS. Springer, 1996: 147-166.

[40] W. Roscoe., C. A. R Hoare. The Theory and Practice of Concurrency [M]. Prentice-Hall. 1998.

[41] C. A. R. Hoare. Communicating sequential processes [M]. Programming Languages. Springer Berlin Heidelberg, 1983.

[42] Formal Systems (Europe) Ltd, Failures-Divergences Refinement. FDR2 User Manual, 1998.

[43] 张玉清, 莫燕, 吴建耀. 安全协议的建模与分析: CSP 方式 .机械工业出版社.2005.

[44] 周巢尘. 通信的顺序进程及其研究 [J]. 计算机学报, 1983, 9(1): 1-9.

万方数据

万方数据

硕士研究生期间取得的研究成果

一、参加科研情况

1. 信息安全试验平台的关键技术研究 2012.7-2013.8

2. 陕西省自然科学基础研究计划资助项目 2012.5-2013.4

二、发表论文情况

[1] H. Zhu, T. Liu, G. Wei, et al. PPAS: privacy protection authentication scheme for VANET [J]. Cluster Computing. Dec 2013, 16(4), 873-886 SCI 检索: 000327854400021.

[2] H. Zhu, T. Liu, G. Wei, et al. CSP-Based General Detection Model of Network Covert Storage Channels [M]. International Conference, ICT-EurAsia 2013, Yogyakarta, Indonesia, March 25-29, 2013. Information and Communication Technology. Springer Berlin Heidelberg. 2013. 459-468. EI 检索: 20131316145990.

三、专利申请情况

[1] 朱辉, 李晖, 王勇, 裴庆祺, 魏光辉, 曾栋, 康毓涛, 任海, 刘婷婷. 文件安全保护系统及其方法. 中国：CN102970299 A. 2013-3-13

万方数据

万方数据