DOI：10.7652/xjtuxb201512011

采用控制流监控的 Cisco IOS 指针攻击检测方法

刘胜利，邹睿，彭飞，武东英，肖达
（中国人民解放军信息工程大学数学工程与先进计算国家重点实验室，450000，郑州）

摘要：针对当前 Cisco IOS(internetwork operating system)漏洞攻击检测方法检测效率低的问题，提出了一种采用控制流监控来判定 Cisco IOS 指针攻击的方法。该方法通过静态分析和动态跟踪相结合的方式对 Cisco IOS 中不同类别的控制流分别构造合法转移地址集合(legal transfer address collection，LTAC)，并在发生控制流转移时将转移地址在 LTAC 之外的控制流判定为攻击，同时捕获异常控制流转移的详细信息。实验结果表明：该方法可以准确地捕获针对 Cisco IOS 的指针攻击，支持对攻击过程的分析，与现有的 Cisco IOS 漏洞攻击检测方法相比，具有较高的检测效率，能够为网络安全性的提升提供帮助。

关键词：Cisco IOS；指针攻击；控制流监控；网络安全；攻击检测
中图分类号：TP309.08 **文献标志码**：A **文章编号**：0253-987X(2015)12-0065-06

A Method for Detecting Cisco IOS Pointer Attack Using Control Flow Monitoring

LIU Shengli, ZOU Rui, PENG Fei, WU Dongying, XIAO Da
(State Key Laboratory of Mathematical Engineering and Advanced Computing, The PLA Information Engineering University, Zhengzhou 450000, China)

Abstract: A method to detect Cisco IOS(internetwork operating system) pointer attacks using control flow monitoring is proposed to solve the problem that current methods for detecting exploit attacks have low detection efficiency. Legal transfer address collection (LTAC) is constructed for different categories of control flows through a combination of static analysis and dynamic tracking. When an event of control flow transfer occurs, the control flow with jump address out of LTAC will be determined to be an attack and the details of the exception control flow transfer can be captured. Experimental results show that the proposed method can accurately capture pointer attacks against Cisco IOS, and support the analysis of attack process. Comparisons with the current detection methods show that the proposed method has higher detection efficiency, and can help to enhance network security.

Keywords: Cisco IOS; pointer attack; control flow monitoring; cyber security; attack detection

路由器作为网络基础设施的重要组成部分，其安全性影响着整个互联网的安全。目前，生产路由器的公司主要有 Cisco、华为、Juniper 等。IDC 2014 年公布的报告中，Cisco 路由器占到了全球互联网路由器市场的 62.9%的份额[1]，居于首位。

Cisco 路由器中运行的是 Cisco 公司为其路由交换设备开发的操作系统 Cisco IOS，用于管理路由器的硬件资源以及实现各种网络服务。目前，在

收稿日期：2015-04-22。 作者简介：刘胜利(1973—)，男，副教授。 基金项目：国家科技支撑计划资助项目(2012BAH47B01)；国家自然科学基金资助项目(61271252)。
网络出版时间：2015-11-18 网络出版地址：http://www.cnki.net/kcms/detail/61.1069.T.20151118.1614.002.html

Cisco 路由器脆弱性利用方面，Linder 在文献[2]中完整地分析了当时在 Cisco IOS 中发现的一些安全问题，同时给出了缓冲区溢出漏洞利用的一些方法。Lynn 在文献[3]中对 Cisco IOS 的体系结构进行了分析，探讨了 Cisco IOS 任意代码执行的可行性。Muniz 在文献[4]中介绍了使用 IDA 辅助进行 IOS 逆向分析的脚本，并且给出了一种实现 IOS rootkit 注入的方法，并称该方法可以达到高度的隐蔽性。Linder 在文献[5]中对 Cisco IOS 的注入方法进行了阐述，同时分析了其安全结构。Muniz 在文献[6]中提出了一种基于虚拟化的 Fuzzing 方法，用于 Cisco IOS 漏洞的挖掘和调试分析。在安全性检测方面，Linder 在文献[7]中对 Cisco IOS 的取证工具 CIR(Cisco incidence response)[8] 作了介绍，该工具通过验证 IOS 映像的完整性和路由器运行时的状态来判断 IOS 是否受到了安全威胁，由于 Cisco IOS 映像完整性很容易被突破[9]，因此，其检测效果较差。陈立根等提出了一种基于动态污点分析的 Cisco IOS 漏洞攻击检测方法，通过将内存读写和指针污染作为攻击的判定规则来检测攻击[10]，然而正常数据包频繁的内存读写操作导致误报率较高，同时检测过程中的时间消耗较大。目前，针对 Cisco 路由器的脆弱性利用的方法不断增多，同时对于其安全性检测的方法十分有限，因此需要引入 Cisco IOS 攻击检测方法，及时发现路由器遭受的入侵。

1 相关工作

在有源码的指针攻击检测方面，gcc 支持在编译过程中对函数指针的保护，其保护的思想是通过判定 canary word[11] 是否被修改来判定攻击。FPValidator[12] 通过在编译器中添加函数类型检查模块来判定目标程序是否受到了攻击，检测效率较高，但是其需要程序源码的支持，不支持二进制程序的检测。

目前，在二进制代码漏洞攻击检测方面主要有动态污点分析[13] 和控制流完整性检查[14] 两种方法。动态污点分析技术通过将网络数据包标记为污点源，在指令的执行过程中实现污点属性的传播，并按照一定的策略实现对攻击行为的检测，由于污点分析本身的局限性，不可避免的存在一定的误报率和漏报率；控制流完整性检查的方法通过对控制流的转移地址进行安全校验来判定攻击，一般需要对目标程序进行分析来构造合法的转移地址集合。

本文深入分析了 Cisco IOS 中的分支转移过程，提出了一种采用控制流监控的 Cisco IOS 指针攻击检测方法，该方法通过监控控制流转移地址的合法性来判定攻击。基于 dynamips 模拟器[15]，设计实现了 Cisco IOS 指针攻击检测系统(Cisco IOS detection system，CIDS)，支持对函数返回和间接跳转控制流的合法性判定，该系统具有不需要源码，检测结果准确的优点。

2 解决思路

目前，由于 Cisco IOS 内存结构的特殊性，在 Cisco IOS 的漏洞利用过程中，针对指针攻击主要是基于“代码重用技术”实现的[5]。下面是一个 Cisco IOS 中存在的漏洞利用实例，通过对其进行跟踪分析之后发现其控制流的转移过程，如图 1 所示，其中存在异常的控制流转移，包括异常的返回地址和异常的间接跳转控制流转移。

如图 1 所示，程序在地址 a 处调用了函数 func，其正常的返回地址应该是地址 b。由于在 func 执行的过程中导致了系统栈出现了溢出，使得返回地址(保存在栈中的 LR 寄存器的值)被覆盖为地址 x。在 func 执行完毕进行正常的函数返回时，指令的执行地址被修改为地址 x，此时出现了异常函数返回控制流的转移，同时，在地址 x 处的指令将系统栈中的地址 p(shellcode 地址)保存到了 ctr 寄存器中，并在地址 y 处实现了异常间接跳转控制流的转移，导致 shellcode 成功执行。

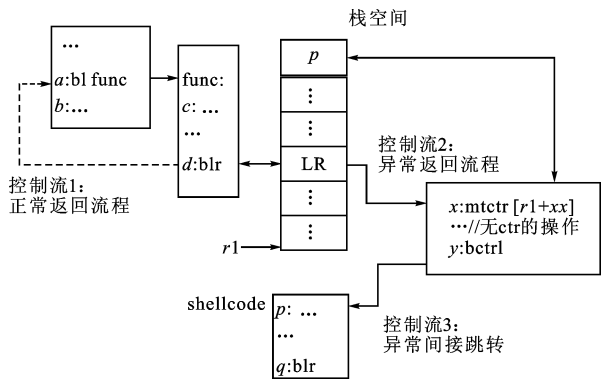

图 1 Cisco IOS 指针攻击示例

从对上述 Cisco IOS 缓冲溢出漏洞利用过程中的控制流转移的分析可知，控制流的转移是进行攻击行为检测的入手点，本文就是基于对异常控制流的监控识别来判定攻击是否发生的。异常控制流转移是漏洞攻击过程中必不可少的过程，从控制流转移的角度出发来对漏洞攻击行为进行检测可以从根本上判定目标程序是否受到攻击。

本文进行 IOS 指针攻击检测主要是基于控制流完整性(control flow integrity，CFI)检查的思想进行的。控制流完整性检查的方法通过限定控制流转移的范围，把目的地址在合法地址集合之外的控制流判定为攻击。fpcheck[16]将二进制程序的代码段作为间接跳转控制流的合法转移地址集合，并不适用于 IOS 的指针攻击检测，本文借鉴了 fpcheck 的检测思想，将数据偏移量交叉引用代码[17]的入口地址和函数入口点作为间接跳转控制流的合法转移地址集合，实现了对函数调用返回地址和间接跳转等控制流的监控。结合 Cisco IOS 本身的特点，提出的 Cisco IOS 指针攻击检测框架如图 2 所示，主要包括两个阶段：在静态分析与动态跟踪阶段，通过逆向二进制的 IOS 文件，定位出函数的入口点和数据偏移量交叉引用代码的入口地址，来构造间接跳转指令的合法转移地址集合；在动态检测阶段，实现对控制流转移合法性的判定。

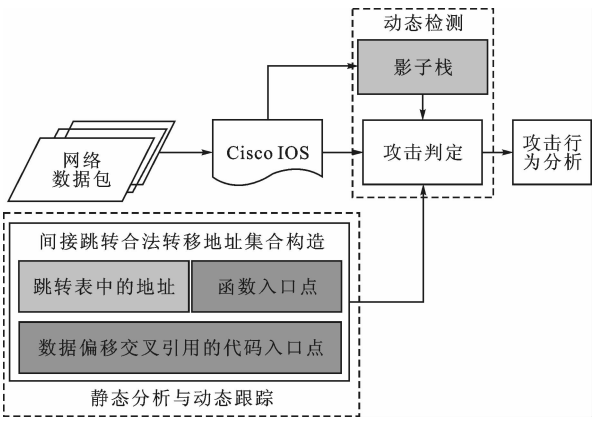

图 2　Cisco IOS 指针攻击检测框架

3　Cisco IOS 指针攻击检测方法

3.1　形式化描述

在进行 Cisco IOS 指针攻击检测框架的介绍之前，本文给出相关的形式化定义。同时，下文的指令均以 PowerPC 指令集为例进行分析。

定义 1　控制流 CF(control flow)是由间接跳转指令 indir_branch 和返回指令 return 组成的。在 Cisco IOS 中，间接跳转指令主要包括 bctr、bctrl，返回指令有 blr。

定义 2　函数入口点集合 F 指的是 Cisco IOS 中所有函数(包括库函数和普通函数)的开始地址。Cisco IOS 是在 Unix 下经过 gcc 交叉编译生成的，通过静态分析发现 Cisco IOS 中的库函数的地址和普通函数一样是固定的，因此，本文将两者当做普通的函数来使用。

定义 3　数据偏移交叉引用代码入口点集合 K 表示 Cisco IOS 中所有具有数据偏移交叉引用属性的代码段的开始地址。

定义 4　跳转表查询函数 J 是查询地址为 x 的间接跳转指令对应的跳转表的函数。其中 X 表示 Cisco IOS 中所有 bctr 指令的地址集合，$\forall x \in X$，Y 表示地址 x 对应的跳转表中地址的集合。

定义 5　间接跳转控制流转移地址集合 I 表示 Cisco IOS 中间接跳转控制流的合法转移地址集合。$I=B \cup B'$，其中 B 表示 bctr 指令的合法转移地址集合，B' 表示 bctrl 指令的合法转移地址集合。

定义 6　影子栈 S 表示在攻击检测过程中维护函数调用返回地址的影子栈，主要是用于判断函数调用返回控制流的合法性。

定义 7　攻击判定规则 D 用于判定 Cisco IOS 是否受到攻击的规则。当 $D=1$ 时，表示判定攻击存在，否则，判定攻击不存在。

3.2　合法转移地址集合构造

本文监控的对象主要分为间接跳转指令和函数调用返回指令两类。

针对函数调用返回指令，本文在对攻击行为动态检测的过程中维护了一个用于存储函数返回地址的影子栈，当发生函数调用返回时，系统会判断函数返回地址是否在影子栈中来判定攻击。

针对间接跳转指令，本文采用静态分析和动态跟踪相结合的方法来完成 I 的构造。对于 Cisco IOS 而言，I 是由间接跳转指令 bctr 和 bctrl 两者合法的目的地址集合 B 和 B' 构成。一方面，通过对 Cisco IOS 逆向分析之后发现，bctr 指令是用于处理 switch 等分支跳转的指令，对于一个确定版本的 IOS 来说，bctr 的目的地址处于跳转表限定的范围内。因此，B 可以通过逆向分析的方法来构造，对跳转表中的地址进行监控可以防止修改跳转表的攻击，因为 Cisco IOS 中的数据段是可写的；另一方面，通过对 Cisco IOS 动态执行时的 bctrl 指令进行跟踪分析发现，其目的地址不仅仅是 F、K，也可以是间接跳转指令的目的地址。因此，对 Cisco IOS 逆向分析同样可以构造 B'。

目前普遍使用的权威反汇编工具是 IDA Pro，并且支持 IOS 逆向分析的工具只有 IDA 的某些版本。如前文所述，在 Cisco IOS 中存在两种函数，即库函数和一般的功能函数，它们的地址在 IOS 中均是静态值。为了更好地识别函数调用的异常控制

流，对于获取的函数信息需满足以下要求：①获取函数信息的地址完整性，这是构造 F 必须满足的条件，只有获取完整的函数地址，才能保证函数入口点的完整性；②获取函数控制流图的完整性，函数调用 CFG 的完整性保证了函数功能的完整性；③对函数控制转移分析的准确性，函数调用控制流转移的准确性保证了对异常控制流识别的准确性。

在完成对 Cisco IOS 的逆向分析之后，可以根据汇编指令代码以及 CFG 来构造 B 和 B'，其中 B 的构造过程如下。

步骤 1 对逆向 Cisco IOS 得到的汇编代码从第一条指令开始遍历。

步骤 2 当遍历的指令是间接跳转指令 bctr 时，转至步骤 3；否则转至步骤 4。

步骤 3 根据生成的 CFG 查询跳转表中的地址，并将 bctr 指令的地址和跳转表中的地址集合作为一条记录添加到 B 中。当本条指令不是最后一条指令时转至步骤 4；否则转至步骤 5。

步骤 4 继续遍历下一条汇编指令，转至步骤 2。

步骤 5 遍历结束，输出 B 集合。

B'同样是通过遍历 IOS 的汇编指令代码进行构建的，查询的是函数的入口点地址 F 和 K，这里不再赘述。至此，间接跳转控制流的合法转移地址集合 I 构造完毕。

3.3 攻击判定

3.3.1 *函数调用返回控制流监控* Cisco IOS 在运行过程中，函数调用返回时会发生控制流的转移。正常情况下，函数返回地址是函数在被调用时压入函数栈帧中的地址，并通过 LR 寄存器传递。本文对函数返回控制流的监控是在函数调用完毕返回之前进行的，如果发现被调用函数的返回地址非法，则判定受到了指针攻击。在实现过程中本文借鉴了文献[18]的检测思想，在系统中动态维护一个用于存储函数返回地址的影子栈，并在发生函数调用时，将返回地址存储到影子栈中。当函数调用返回时，会检查目的地址是否和影子栈中保存的地址相匹配，若不匹配则判定为异常控制流转移，同时记录控制流转移的信息。考虑到异常控制流转移时，返回地址不一定在栈顶，本文只验证函数返回地址是否在影子栈中，系统在验证时会从栈顶到栈底进行遍历查找，当查找到函数的返回地址时，则将返回地址及之前遍历过的所有地址从栈中弹出。若在影子栈中没有发现被调用函数的返回地址，则判定此控制流为异常控制流。对函数调用返回控制流转移合法性的判定规则为

$$D=\begin{cases}1, & x\notin S\\ 0, & x\in S\end{cases} \tag{1}$$

式中：x 表示函数调用返回指令的目的地址。$D=0$ 表示函数调用返回控制流合法；否则，判定为非法。

3.3.2 *间接跳转控制流监控* 间接跳转控制流有两类：一类是使用 bctr 指令进行间接跳转的控制流；另一类是使用 bctrl 指令进行间接跳转的控制流。间接跳转控制流 bctrl 的合法地址应当是属于 F 和 K 的并集，而 bctr 的合法转移地址应当属于其跳转表所限定的范围。因此，本文定义间接跳转控制流合法性的判定规则如下

$$D=\begin{cases}1, & y\notin J(x)\\ 0, & y\in J(x)\end{cases} \tag{2}$$

式中：y 表示 bctr 指令的转移地址；x 表示对应 bctr 指令的地址。$D=1$ 表示间接跳转控制流非法，否则表示合法。

$$D=\begin{cases}1, & y'\notin (F\cup K)\\ 0, & y'\in (F\cup K)\end{cases} \tag{3}$$

式中：y'表示 bctrl 间接跳转指令执行时的目的地址。$D=1$ 表示间接跳转控制流非法，否则说明控制流合法。

3.4 攻击行为分析

本文在攻击检测过程中，当发现异常控制流转移之后，会对异常控制流及其之后的相关指令和函数调用信息进行记录。对于指令，记录其执行时的内存地址、操作数的值以及指令的类型；函数调用信息的记录主要有函数的调用地址、运行时的参数值以及函数的返回值等信息。

通过对上述异常控制流的转移信息进行分析，可以还原整个漏洞利用的过程。例如通过对缓冲区溢出攻击过程中的异常控制流进行分析，可以了解漏洞触发时溢出函数的地址以及修改的返回地址等信息；通过对异常控制流链的分析，可以了解攻击者对于漏洞利用的步骤，对于网络安全的提升具有重要意义。

4 实验分析

为了验证本文提出方法的有效性，基于 dynamips 设计实现了一个 Cisco IOS 指针攻击检测系统 CIDS。为了降低系统运行的时间开销，本文将间接跳转指令对应的跳转表用 hash 链表结构存储，将 $F\cup K$ 中的地址用 bitmap 结构存储。本文选取了 IOS 的公开漏洞作为样本，从功能和性能两个方面

对 CIDS 进行测试。限于篇幅，在测试过程中只记录了部分的实验结果。实验中 CIDS 系统运行于 ubuntu12.04 系统之上，并且支持多种型号的 Cisco 路由器的硬件虚拟化。

4.1　功能测试

下面以 FTP MKD 漏洞[19]样本为例，对 CIDS 的测试过程如下。

首先，启动 CIDS 加载运行 Cisco 2600 系列的主版本号为 12.3 的某款 IOS，并在上面开启 FTP 服务。其次，利用测试样本向 CIDS 发送 FTP MKD 漏洞利用数据包，其中包含了 shellcode 漏洞利用代码，在处理数据包的过程中会出现异常控制流的转移。CIDS 将不在合法转移地址集合内的控制流转移判定为攻击，并记录异常的控制流转移时的上下文信息，如图 3 所示。

```
0x80a07e44 blr:1r=0x801523f8
the function return contorl flow whose ins add is 0x80a07e44 is
dangerous,the dangerous return is 0x801523f8.
0x80152424 blr:1r==0x80f23c58
the function return contorl flow whose ins add is 0x80152424 is
dangerous,the dangerous return is 0x80f23c58.
0x80f23c74 bctr1:ctr==0x81e46e24
the indirect jump contorl flow whose ins add is 0x80f23c74 is
dangerous,the dangerous return is 0x81e46e24
```

图 3　异常控制流转移的信息

本次攻击的过程分为 3 个步骤。首先，在地址 0x80a07e44 处，blr 函数返回地址出现异常，被修改为 0x801523f8，在函数返回时，CPU 会跳转到地址 0x801523f8 处执行，满足式(1)所示的报警规则。其次，在地址 0x80152424 处，blr 函数返回地址出现异常，返回地址被修改为 0x80f23c58，之后函数返回，CPU 跳转到此处执行，满足式(1)所示的报警规则。最后，在地址 0x80f23c74 处，bctrl 间接跳转控制流出现异常，指令地址变成 0x81e46e24，通过逆向分析发现，该地址处于 Cisco IOS 的数据段，即 CPU 跳转执行了 shellcode，满足式(3)所示的报警规则。

为了测试 CIDS 对 Cisco IOS 指针攻击检测的适用性，实验中使用存在于相关版本 IOS 中的 TFTP 长文件名漏洞[20]、NHRP 漏洞[21]、IP option[22]等漏洞作为测试样本，对不同型号的路由器进行了测试。测试记录的结果见表 1。

表 1　CIDS 攻击测试结果统计

协议	Cisco 路由器型号	IOS 版本	IOS 数量	溢出类型	漏洞编号	异常控制流条数	是否成功检测
FTP	2621	12.0～12.3	30	栈溢出	CSCek55259/CSCse29244	3(blr/blr/bctrl)	是
TFTP	1700	11.1～11.3	35	堆溢出	CVE-2002-0813	1(blr)	是
IP	2610	11.3～12.3	50	栈溢出	CSCec71950	4(blr/blr/blr/bctrl)	是
NHRP	2621	11.3～12.3	50	堆溢出	CSCsi23231	1(blr)	是

由表 1 的测试结果可以看出，CIDS 能够准确检测出 Cisco IOS 漏洞利用的攻击行为，并可以记录异常控制流的相关信息，证明了本文检测方法的有效性。

4.2　性能测试

为了测试 CIDS 的检测效率，实验中对漏洞攻击检测的时间进行了测试，并与文献[10]提出的 CtaintDetect 系统进行了对比分析。由于 CtaintDetect 和 CIDS 均是基于 dynamips 系统开发的，因此在测试过程中针对上述 4 个漏洞，本文对比测试了原型 dynamips 系统、CtaintDetect 和 CIDS 三者在攻击过程中的时间消耗比，如图 4 所示，其中时间消耗比以 dynamips 原型系统的时间消耗为基准。实验中选取了多个不同版本的 IOS 进行测试，针对每一个漏洞，其时间消耗比是多个样本的平均值。

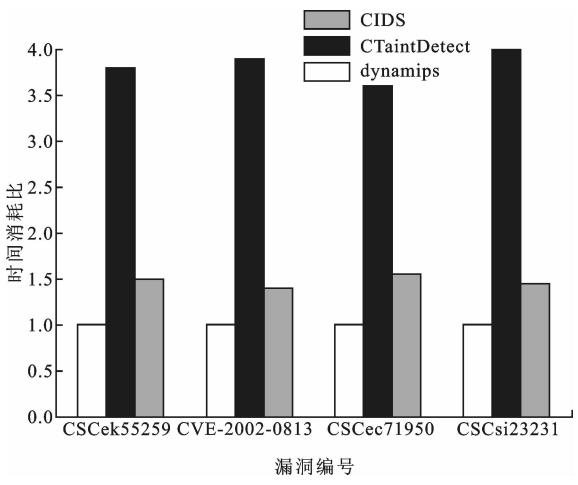

图 4　时间消耗对比测试结果

由图 4 可以看出，CtaintDetect 系统的时间消耗大概是 dynamips 原型系统的 3.8 倍，CIDS 大概是 dynamips 原型系统的 1.5 倍，可以满足性能方面的需求，因此与 CtaintDetect 系统相比，本文所提方法检测效率更高。

从上述实验结果可知，对于 Cisco IOS 的指针

攻击检测而言,CIDS 可以成功捕获攻击,并可以记录攻击过程中的异常控制流转移的信息。与利用动态污点分析技术实现的 CtaintDetect 系统相比,CIDS进行攻击检测的时间开销较低,证明了本文所提方法的有效性,并具有较高的检测效率,可以为网络安全提供支撑。

4.3 局限性分析

从实验测试的结果来看,本文提出的针对 Cisco IOS 指针攻击的检测方法具有较好的检测效果,但是也存在一定的局限性。针对指针攻击检测方法,在处理间接跳转时,是以函数入口地址以及数据偏移量交叉引用代码入口地址为合法的转移地址集合,当异常的间接跳转地址属于合法转移地址集合,并直接通过此数据偏移量交叉引用代码或者函数来实现恶意攻击时,本文方法会将此次间接跳转判定为合法的控制流转移,此时出现漏报。然而,目前主流针对 Cisco IOS 的漏洞利用攻击的目的是为了获取系统的访问权限,一般通过开启远程的 vty 来实现,通过单个函数或者数据偏移量交叉引用代码很难完成上述功能,因此,此种攻击方法很难在实际的攻击过程中应用。

5 结 论

本文研究了 Cisco IOS 的指针攻击检测技术,提出了采用控制流监控的 Cisco IOS 指针攻击检测方法。首先对比分析了当前二进制软件的漏洞攻击检测方法,对 Cisco IOS 的分支转移过程进行了深入分析,介绍了本文解决思路和检测框架。其次,实现了 Cisco IOS 的指针攻击检测技术,包括不同控制流合法转移地址集合的构造,不同攻击行为的判定规则的设定等。最后,实现了一个 Cisco IOS 指针攻击检测系统 CIDS,通过实验对本文所提检测方法的有效性进行了验证,并与 CtaintDetect 系统进行了对比测试。实验结果证明了本文所提方法的有效性。CIDS 支持对攻击流程的分析,并具有较高的检测效率,有助于提升网络安全性。

参考文献:

[1] IDC. IDC's worldwide quarterly Ethernet switch and router tracker shows record ethernet switch market size, weaker router market [EB/OL]. (2014-12-03) [2015-03-15]. http://www.idc.com/getdoc.jsp?containerId=prUS25266314.

[2] LINDER F. Design and software vulnerability in embedded system [EB/OL]. (2003-04-25)[2014-08-19]. https://www.blackhat.com/presentation/bh-usa-03/bh-us-03-fx.pdf.

[3] LYNN M. The holy grail: Cisco IOS shellcode and exploitation techniques [EB/OL]. (2005-11-12)[2014-10-14]. http://cryptome.org/lynn-cisco.pdf.

[4] MUNIZ S. Killing the myth of Cisco IOS rootkits: DIK (Da IOS rootKit) [EB/OL]. (2008-03-26)[2014-06-19]. http://www.coresecurity.com/content/killing-the-myth-cisco-ios.pdf.

[5] LINDER F. Cisco IOS router exploitation [EB/OL]. (2009-06-22)[2014-09-02]. http://www.blackhat.com/presentations/bh-usa-09/Linder/BH_US_09_Linder_RouterExploit_PAPER.pdf.

[6] MUNIZ S, ORTEGA A. Fuzzing and debugging Cisco IOS [EB/OL]. (2011-12-21)[2014-07-18]. http://www.pdfpedia.com/download/13758/fuzzing-and-debugging-cisoc-ios-blackhat-europe-2011-pdf.html.

[7] LINDER F. Developments in Cisco IOS forensics [EB/OL]. (2009-08-14)[2013-03-10]. http://www.blackhat.com/presentions/bn-usa-08/Linder/BH_US_08_Linder_Developments_in_IOS_Froensics.pdf.

[8] Recurity Labs. CIR [EB/OL]. (2008-02-16)[2014-01-12]. http://cir.recurity.com.

[9] SU Xiaoyan, WU Dongying, XIAO Da, et al. Research on Cisco IOS security mechanisms [C]// Proceedings of the International Conference on Computer Science and Information Technology. Piscataway, NJ, USA: IEEE, 2012, 51: 653.

[10] 陈立根, 刘胜利, 高翔, 等. 一种基于动态污点分析的 Cisco IOS 漏洞攻击检测方法 [J]. 小型微型计算机系统, 2014, 35(8): 1798-1802.
CHEN Ligen, LIU Shengli, GAO Xiang, et al. A vulnerability attack detection method based on dynamic taint analysis for Cisco IOS [J]. Journal of Chinese Computer Systems, 2014, 35(8): 1798-1802.

[11] COWAN C, PU C, MAIER D, et al. StackGuard: automatic adaptive detection and prevention of buffer-overflow attacks [C]// Proceedings of the 7th Conference on USENIX Security Symposium. Berkeley, CA, USA: USENIX, 1998: 63-78.

[12] WANG Hua, GUO Yao, CHEN Xianqqun. FPValidator: validating type equivalence of function pointers on the fly [C]// Proceedings of the 2009 Annual Computer Security Applications Conference. Piscataway, NJ, USA: IEEE, 2009: 51-59.

(下转第 111 页)