doi:10.11959/j.issn.1000-436x.2017202

基于 OpenFlow 的网络层移动目标防御方案

胡毅勋[1]，郑康锋[1]，杨义先[1,2]，钮心忻[1,2]
(1. 北京邮电大学网络空间安全学院，北京 100876；2. 贵州大学公共大数据国家重点实验室，贵州 贵阳 550025)

摘 要：为在网络攻防博弈中占据主动地位，利用 OpenFlow 网络结构提供的网络灵活性，提出一个基于 OpenFlow 的网络层移动目标防御方案。在网络层，通过对防护区域内通信中的每一跳网络地址进行伪随机变换，对跨区域网络通信的出口端口进行伪随机映射，从而实现通信节点的隐藏以及网络结构的保护。实验表明，该方案有效可行。相比于现有移动目标防御方案，该方案易部署、兼容性好，并实现了节点全网的通信保护。
关键词：主动防御；OpenFlow；移动目标防御
中图分类号：TN915.08 **文献标识码**：A

Moving target defense solution on network layer based on OpenFlow

HU Yi-xun[1], ZHENG Kang-feng[1], YANG Yi-xian[1,2], NIU Xin-xin[1,2]
(1. College of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China;
2. State Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, China)

Abstract: In order to take an active part in network attack and defense, a moving target defense solution on network layer based on OpenFlow was proposed, using the flexibility of network brought by OpenFlow network architecture. On the network layer, through mapping the correspondent nodes' addresses to pseudo-random virtual addresses in the LAN and mapping correspondent nodes' ports to virtual ports, achieving the hiding of correspond nodes in the whole network and the information of network architecture. Researches verify the system's effectiveness. Comparing with existing moving target defense solutions, the proposed algorithm can be deployed easily in the traditional network, and realize comprehensive protection of the corresponding in the whole network.
Key words: active defense, OpenFlow, moving target defense

1 引言

近年来，随着多起具有广泛危害性安全事故的发生（棱镜门、SSL 心脏滴血等），网络安全再一次受到了广泛的重视。传统的网络安全技术通常采用被动式防御（如防火墙、入侵检测技术等）。这类防御技术通常将保护目标暴露在外，检测和保护的前提是攻击正在发生[1]。这种方式对于保护者而言十分不利，防御方总是处于被动防御的地位。主动防御技术是一种让保护者在攻防博弈中占有主动地位的防御手段。移动目标防御技术则是主动防御技术中一个重要的研究方向。

移动目标防御是一种新型的防御技术，这种技术的主要目的是使被保护目标对于外界而言一直处于移动和变化中，使攻击者难以捕捉目标，以此实现保护。国内外学者都对移动目标防御技术进行了相关的研究，可是现有成果大多基于第三方软件实现对上层不透明的保护，上层应用必须进行相关改动。不使用第三方软件的方案则存在对现有网络不兼容的问题。

本文结合之前的研究成果，提出了一种基于 OpenFlow 的网络层移动目标防御方案。本文方案

收稿日期：2017-03-15；**修回日期**：2017-07-08
通信作者：胡毅勋，hyx.bupt@gmail.com
基金项目：国家重点研发计划基金资助项目（No.2017YFB0802703）；国家自然科学基金资助项目（No.61602052）
Foundation Items: The National Key Research and Development Program of China (No.2017YFB0802703), The National Natural Science Foundation of China (No.61602052)

基于 OpenFlow 实现软件定义网络架构，从而为移动目标防御功能提供网络控制能力。在软件定义网络架构基础上，对域间流量使用网络地址跳变策略实现网络地址的逻辑移动，对跨域流量使用端口跳变实现跨域通信时的节点隐藏，最终实现对被保护系统的全面防御。方案具有部署简单，且与传统网络兼容的特点。被保护节点不需要进行额外的配置即可实现通信保护，对于防网络嗅探有极好的效果。

2　相关工作

移动目标防御技术是一种在当前网络攻防博弈易攻难守的情况下提出的主动防御技术[2]。移动目标防御技术有多种实现方式，包括对软件多样性的实现[3,4]、底层指令的变化[5,6]和网络层的移动目标防御[7]等。文献[8]通过对几类移动目标防御系统的研究和分析，总结出 3 种移动目标防御系统的运行模式，从而帮助研究和设计人员研究和开发新型的移动目标防御系统。文献[9]结合入侵检测技术和虚拟化技术实现攻击流量的识别以及重定向，并通过高性能计算机部署控制和计算功能，从而完成具有低处理时延的移动目标防御功能。

本文在网络层设计移动目标防御方案。文献[1]提出了一种基于端信息跳变的主动防御技术，这种技术从军事跳频中得到启发，对于通信的网络数据不断修改其通信端口，从而实现通信的跳频和信息的隐藏。这种方式的缺点是在实现方式上对操作系统网络通信内核进行修改，在使用时需要为每个通信主机都需要进行相应的部署，不利于大规模使用。软件定义网络[10]（SDN, software defined network）是一种新型的网络架构，该架构将传统数据传输的控制层和转发层分离，实现一种可编程的网络结构。其中，OpenFlow 协议[11]是 SDN 的一种主流实现。文献[12]基于 OpenFlow/SDN 技术使用 IP 地址变化的方式实现移动目标防御。在通信过程中使用虚拟 IP 作为网络地址，实现通信信息的隐藏。文献[13]同样基于 OpenFlow/SDN 技术，在文献[12]基础上将全局的 IP 变化分为低频和高频的嵌套，降低 IP 地址变化给 OpenFlow 交换机带来的负担。基于 OpenFlow/SDN 的移动目标防御方法将移动目标防御功能部署在网络层的网络设备上，对于上层应用透明，将移动目标防御功能以模块的形式添加到现有网络中。文献[14]针对现有移动目标防御系统中对于网络地址的变化范围小的问题，提出了一种基于 OpenFlow 的可以拓宽网络地址变化范围的移动目标防御系统。但是这 3 种基于 OpenFlow/SDN 的 IP 跳变技术只能适用于可以部署并且可以重构网络结构和设备的区域。对于通过 Internet 传输的网络而言（使用传统网络设备构建），重新部署公共网络是不现实的。因此，基于 OpenFlow 并且可以部署于全网环境的移动目标防御系统的设计成为了当前的一个研究重点。

基于上述事实，本文提出了一种基于 OpenFlow 的移动目标防御方法。该方案在网络层部署透明的移动目标防御组件，上层网络应用不需修改软件本身。同时，这种方案不仅对内部网络的域内通信有保护效果，同时对于通过 Internet 传输的跨域数据同样具有保护的作用。

3　防御方案

本节详细描述基于 OpenFlow 的网络层移动目标防御方案。该方案使用基于 OpenFlow 的 SDN 解决方案构建防御系统所需的基础网络平台，平台包括链路控制、数据转发和分组修改等功能。在该网络平台上设计网络层移动目标防御组件并实现各部分功能，最终在网络层实现移动目标防御功能。

3.1　总体结构

本文提出的移动目标防御方案具有以下 3 个设计要求。

1) 与传统网络兼容。实现一个易部署的移动目标防御系统。

2) 传输透明。节点无需安装第三方插件和对原本网络通信方式进行修改即可在本文方案部署环境下实现保护。

3) 全网的防御方案。本文方案不单保护节点在域内的通信安全，同时对于节点跨域传输的数据也进行保护，实现一个对节点通信全面的保护方案。

基于上述目标和需求，本文提出如图 1 所示部署结构的移动目标防御方案。在域内使用 IP 跳变算法进行域内通信保护，在域间则使用端口跳变进行跨域的通信保护。MTD 控制器基于 OpenFlow 控制器部署移动目标防御策略，并完成网络状态的监视和维护功能。位于不同域的 MTD 控制器通过域间同步服务器实现策略的同步。

3.2　MTD 控制器

MTD 控制器由一个 OpenFlow 控制器组件和基于其部署的移动目标防御组件构成，如图 2 所示。

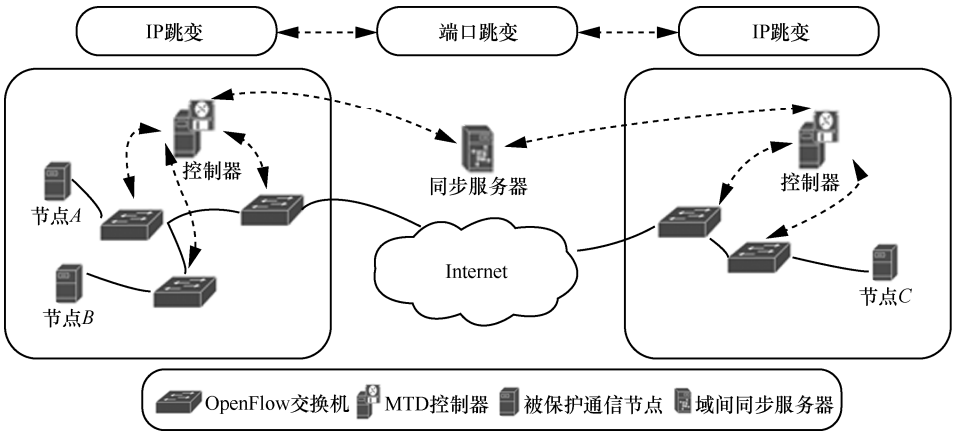


图 1　系统部署结构

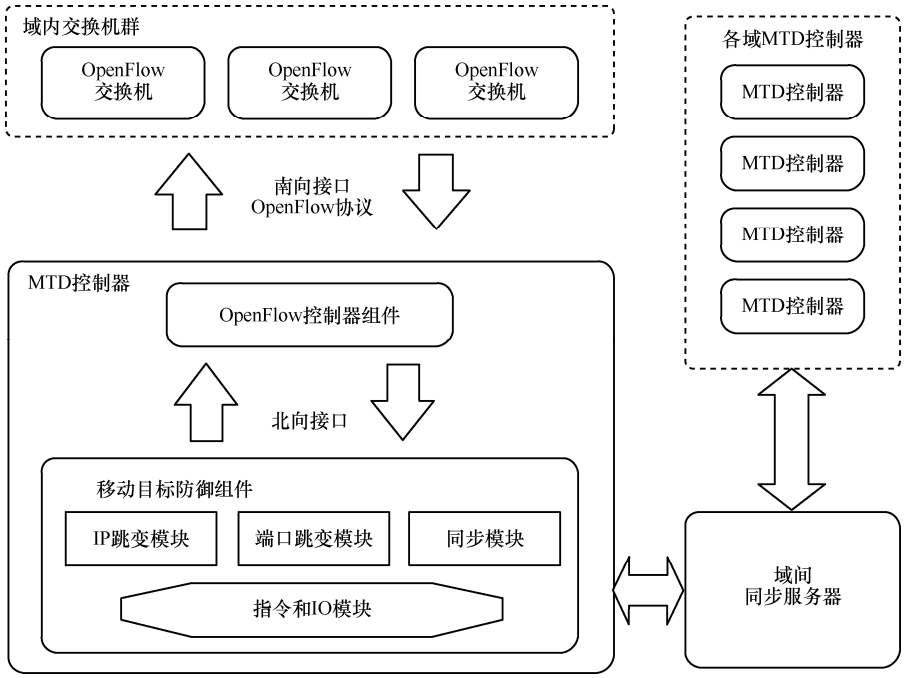


图 2　移动目标防御组件结构

移动目标防御组件通过北向接口与 OpenFlow 控制器组件交互，组件包括 IP 跳变模块、端口跳变模块、同步模块以及指令和 IO 模块。各区域内的 MTD 控制器与同步服务器相连，实现端口跳变信息的同步。

IP 跳变模块负责实现域内的 IP 地址跳变功能。IP 地址跳变指在域内进行传输的数据分组在传输路径上的每跳节点进行网络地址的变化，从而实现通信双方网络地址的隐藏。攻击者无法通过嗅探和分析在网络中间位置截取的数据分组进行网络中存活节点的识别。

端口跳变模块负责对域间的数据通信进行保护。为了实现对现有网络的兼容，本文防御方案设计部署一个功能类似于现有网络中网关功能的 OpenFlow 交换机。该交换机实现对于域间数据的端口跳变，将内部的传输数据通过端口跳变的方式进行隐藏，并实现对于从域间接收到的已跳变数据分组的还原。恶意攻击者无法通过对于域间传播的数据分组的嗅探和分析识别内部节点。

同步模块的作用在于对部署在不同区域的防御系统进行端口跳变信息的同步。同步模块与域间同步服务器通过 SSL 安全协议实现加密通信，保护同步信息。而同步服务器和控制器则通过已知安全防御方案（如防火墙、入侵检测等）实现服务器和数据安全。

指令和 IO 模块实现移动目标防御组件与

OpenFlow 控制器的数据交互和北向接口的指令生成与解析。

3.3 IP 跳变

本文设计移动目标防御方案将节点的通信过程分为 2 种：域内通信和域间通信。对于域内通信，本文设计 IP 跳变的方式进行节点保护。结合通过 OpenFlow 提供的网络控制能力和本文设计通信协议实现 IP 跳变功能。IP 跳变模块部署在基于 OpenFlow 控制器的 MTD 控制器上。

3.3.1 IP 跳变过程

假设对处于同一个域内的节点 A 和节点 B，通过 OpenFlow 交换机 $switch_1, switch_2, \cdots, switch_n$ 相连接。其中所有的 OpenFlow 交换机开启被动路由模式，即当有未知转发方式的数据分组到达时，首先询问控制器该数据分组的转发方式，再根据接收到的转发流表进行转发。当节点 A 想要对节点 B 发起连接尝试时，节点 A 根据节点 B 的网络地址信息构造数据分组。数据分组头部的五元组信息 {sIP_1,$sPort_1$, dIP_1, $dPort_1$, protocol}，表示源 IP 地址、源端口、目的 IP 地址、目的端口和协议类型。如图 3 所示，节点 A 构造的初始五元组中，源地址和目的地址分别是节点 A 和节点 B 的网络地址。在数据分组到达与节点 A 直连的 $switch_1$ 时，$switch_1$ 将该分组头部信息（五元组信息）发送给控制器询问该数据分组的转发方式。控制器收到请求信息后，根据分组头部信息，通过 3.3.2 节所述算法对传输路径上不同交换机生成不同的转发策略 {$forwarding_1$, $forwarding_2, \cdots$, $forwarding_n$}，并将策略下发到对应的交换机中。当交换机收到相应的转发策略后，对数据分组完成相应的转发行为。本文提出的 IP 跳变策略在经过的每一跳 OpenFlow 交换机上均会进行源和目的 IP 地址的改变，以此达到源地址和目的地址的隐藏和伪装，实现对保护目标的逻辑移动。假设攻击者在获取到域内关键网络位置后，在本文提出的 IP 跳变策略保护下，攻击者无法通过分析获取到的数据分组实现存活节点的嗅探。

在整个通信过程中，只有与通信节点直连的交换机端口可以收到和发出具有真实 IP 地址的数据分组，其他（包括直连交换机与其他交换机）端口收到和发出的数据分组头部均为虚拟地址，而攻击者无法与通信节点同时连接同一个交换机端口，因此，攻击者也就无法在节点通信过程中获取具有真实节点网络地址的数据分组头部。

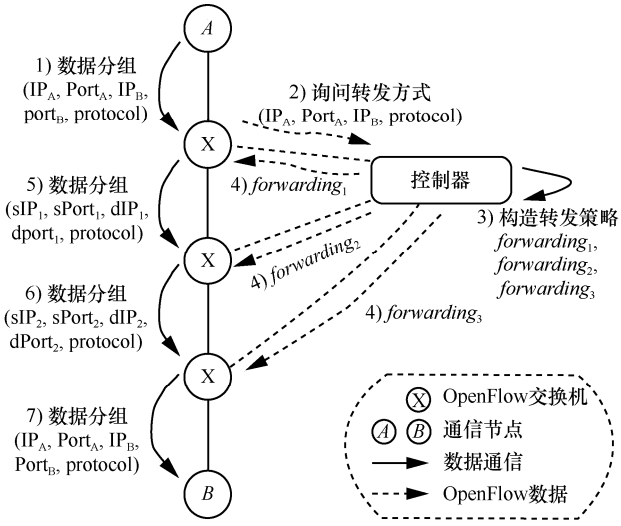


图 3　IP 跳变流程

3.3.2 IP 跳变策略循环生成算法

在部署域内 OpenFlow 交换机时，首先通过 Dijkstra 算法计算域内各交换机之间的最短路径（即路由）。在本文算法中，路径的长度为路径上经过的交换机个数。在计算最短路径时，对于域内不同交换机节点 i 和 j，首先计算最短路径 r_{ij} 及其链路长度 l_{ij}，并构建最短路径矩阵 $\boldsymbol{Route} = \{(r_{ij}, l_{ij}) \mid i, j \in LAN, i \neq j\}$。

算法 1　IP 跳变策略循环生成算法

初始化　已计算各路由器之间的最短路径矩阵 ***Route***，当前请求转发的数据分组头部信息 {sIP, sPort, dIP,dPort,protocol}。

1) 当通信双方在 δ_t 时间内不产生新数据分组时进行循环。

2) 查询最短路径矩阵，得到该数据分组的传输路径 r_{ij}。

3) 循环获取 r_{ij} 上的所有交换机，并对交换机生成相应转发策略。首先通过伪随机算法生成随机源 IP 和随机目的 IP，若新生成随机 IP 地址已被现有策略使用，则重新生成随机 IP 直到不重复为止，并将其加入当前正在使用的 IP 库中。转发策略设置如表 1 所示。

在转发策略中，数据分组的第一跳交换机使用的源 IP 和目的 IP 为数据分组的源 IP 和目的 IP，其他交换机均使用前一条策略中改变后的源和目的 IP。在最后一跳交换机上，需要将源 IP 和目的 IP 改为数据分组原始地址，以此兼容现有协议。

表 1 **转发流**

策略 1		策略 2	
匹配	行为	匹配	行为
源 IP：sIP	改变源 IP：newsIP	源 IP：newdIP	改变源 IP：dIP
目的 IP：dIP	改变目的 IP：newdIP	目的 IP：newsIP	改变目的 IP：sIP

4) 将路径上每个交换机的转发策略下发至相应交换机中。

5) 等待 δ_t。

6) 循环结束。

7) 删除路径上每个交换机关于该数据分组头部的最新策略，并将当前使用的 IP 地址从 IP 地址库中删除。

在算法中，每经过 δ_t 的时间判断该次连接是否已经结束或休眠，若已经结束或休眠则删除关于该数据分组的所有转发策略。反之，则进行 IP 跳变策略的更新，增加随机性和节点隐藏性。在 IP 跳变策略生成时，同时生成当前传输方向跳变策略和响应传输时的反向跳变策略。

3.4 端口跳变

对于域间的数据传输而言，在与传统网络兼容需求的前提下，不能改变原有的域间传输设备和协议。而通过修改每跳路径的数据分组头部信息的 IP 跳变方式，必须将传统网络设备替换成 OpenFlow 交换机及搭载跳变组件的控制器才可以实施。因此，本文提出端口跳变的方式实现跨域通信的移动目标防御功能。

3.4.1 端口跳变过程

端口跳变的过程参考传统网络 NAT 的方式，将内部通信链路转换成域间使用的区域公网 IP 地址和相应绑定的端口。不同的是，NAT 对于一次通信会话绑定的端口无法改变，攻击者在不同时刻截取到不同的通信数据分组，可以很容易还原通信会话。因此，在传统网络中使用域间区域公网 IP 和绑定端口，可以唯一表示一次通信会话。攻击者可以针对固定的通信端口进行攻击，攻击无法避免。本文提出的端口跳变方式可以在同次通信会话过程中，不断改变通信双方网关交换机绑定的通信端口，使攻击者无法简单地从窃取到的数据分组中还原出通信内容，定位内部节点。即使攻击者根据某次截取到的 IP 和端口信息来定位攻击目标，也无法在下一时刻再次利用该 IP 和端口定位同一个域内节点。

假定节点 A 和节点 B 分别在不同的区域 LAN1 和 LAN2，LAN1 和 LAN2 通过 Internet 相连，并在 Internet 域间部署同步服务器。LAN1 和 LAN2 与 Internet 相连接的出口位置部署 OpenFlow 网关交换机 $gateway_1$ 和 $gateway_2$，网关交换机在 MTD 控制器的维护下实现端口跳变过程。如图 4 所示，当节点 A 对节点 B 发起一次连接时，节点 A 首先通过 LAN1 内的 DNS 服务器查询节点 B 的公网地址。之后节点 A 构造通信数据分组，数据分组头部信息包含源 IP 地址、源端口、目的 IP 地址、目的端口和协议类型，表示为数据分组五元组 $\{sIP_1,sPort_1,dIP_1,dPort_1,protocol\}$。

在初始数据分组中，五元组信息源地址为节点 A 的网络地址，目的地址为节点 B 的公网地址。数据分组在 LAN1 内的传输过程使用 IP 跳变的方式，每跳交换机更改源和目的 IP，具体步骤参照 3.3 节内容。当数据分组传输至网关交换机时，执行端口跳变过程。网关交换机将包含五元组信息的转发询问分组发送至 MTD 控制器，MTD 控制器通过 3.4.2 节所述算法随机生成端口跳变策略，并下发至网关交换机。同时将端口跳变策略同步至位于 Internet 中的同步服务器。交换机根据跳变策略修改源 IP 为 LAN1 的公网 IP，源端口为转发策略中的随机目的端口。每隔时间 T_{port}，控制器重新根据当前时间计算并修改该通信的端口跳变策略并下发至网关交换机，以此实现频率为 $\frac{1}{T_{\text{port}}}$ 的端口跳变功能。当数据分组通过 Internet 传输到 LAN2 的网关交换机时，LAN2 的网关交换机将收到数据分组的数据分组五元组分组头部信息发送至 LAN2 内 MTD 控制器，并询问转发策略。LAN2 内的控制器实时与同步服务器同步各域的端口跳变信息，并根据 3.4.2 节所述算法并利用当前时间计算和还原数据分组的转发规则，之后将转发规则下发至 LAN2 内相应的交换机。当数据分组在 LAN2 内传输时，同样使用 IP 跳变策略不断修改数据分组的源和目的地址。最终传输至节点 B，完成整个通信过程。

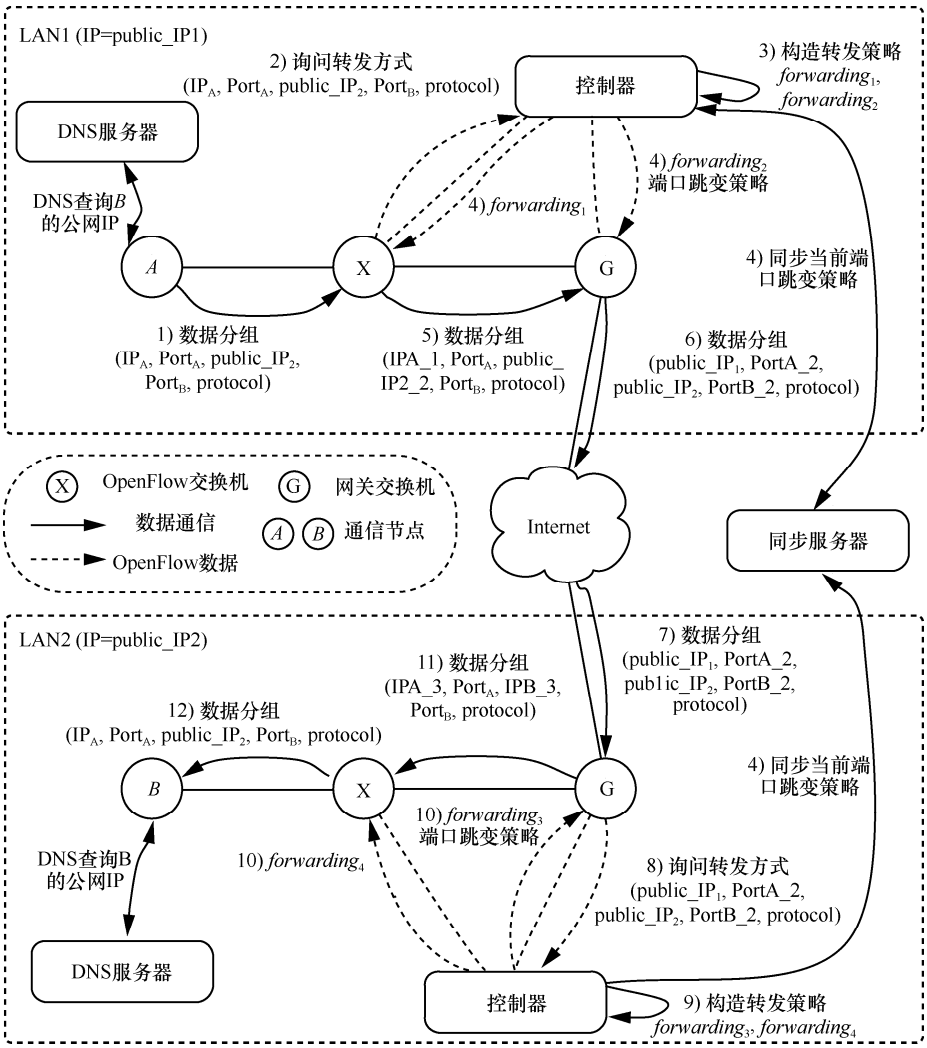


图 4　端口跳变流程

3.4.2　跨域端口跳变策略循环生成算法

当数据分组在域间传播时，本文参考当前主流的域内转域间数据分组转换方式 NAT，将域内传播的数据分组通过端口映射的方式转化为域间公网地址，并周期对映射的端口进行变化，达到隐藏通信特征的效果。

算法 2　跨域端口跳变策略循环生成算法

初始化　数据分组到达网关交换机，数据分组当前分组头部信息{sIP,sPort,dIP, dPort, protocol}。

发送域

1) 当通信双方在 δ_t 时间内不产生新数据分组时进行循环。

2) 随机生成当前未使用的新源端口 newsPort，同时查询同步服务器中获取的新目的端口 newdPort。

3) 如表 2 所示，构造映射流表项，并下发流表。

表 2　发送域端口映射流表

交换机 ID	匹配	行为
gateway1	源 IP：sIP	改变源 IP：$public\_IP_1$
	源端口：sPort	改变源端口：newsPort
	目的 IP：dIP	改变目的端口：newdPort
	目的端口：dPort	

4) 将新流表同步至同步服务器。

5) 等待 δ_t。

6) 结束循环。

7) 清除该数据分组流表项。

接收域

1) 当通信双方在 δ_t 时间内不产生新数据分组时进行循环。

2) 查询同步服务器中的发送端映射流表规则，如表 3 所示生成接收端流表，并将流表下发至网关

交换机。

3) 等待 δ_t。

4) 结束循环。

5) 清除该数据分组流表项。

表 3 接收域端口映射流表

交换机 ID	匹配	行为
gateway2	源 IP：public_IP1	改变源 IP：sIP
	源端口:：newsPort	改变源端口：sPort
	目的 IP：dIP	改变目的 IP：dIP
	目的端口：dPort	改变目的端口：dPort

3.5 安全性分析

本节将对使用基于 OpenFlow 的网络层移动目标防御方法进行保护的节点和系统从抗嗅探能力和抗 DDoS 能力进行分析。

3.5.1 抗嗅探能力分析

通常网络攻击的第一步是节点嗅探，找到存活节点后，确定节点的系统和服务指纹，根据不同的指纹进行不同的攻击。因此，抗嗅探能力可以作为衡量一个防御系统安全性的重要指标。

假定攻击者对域内节点的嗅探能力表示为每秒进行嗅探的次数 $power_{\text{sniffer}}$。在时间 δ_t 内，攻击者能够进行的嗅探总数为 $Sniffer_t = power_{\text{sniffer}} \times \delta_t$。因此，对于未部署防御方案的系统内，攻击者能够在有限的时间内对域内全体网络地址空间完成所有的嗅探工作。所需的嗅探时间与攻击者嗅探能力（单位时间嗅探次数）成反比。

使用本文所述方法搭建防护系统后，假定系统使用的 IP 跳变更新周期为 δ_{tIP}，跳变的随机地址池包含 IP 数量为 N。由于 IP 地址进行周期跳变，因此攻击者无法保证一定能够嗅探出节点。在攻击者嗅探能力不变的前提下，每个 δ_{tIP} 内攻击者嗅探失败率如下所示。

$$p_{\text{LAN\_failure}} = \frac{C_{N-1}^{power_{\text{sniffer}} \times \delta_{tIP}}}{C_N^{power_{\text{sniffer}} \times \delta_{tIP}}} = \frac{N - power_{\text{sniffer}} \times \delta_{tIP}}{N}$$

攻击者在 δ_{tIP} 内能够嗅探的网络地址数量 $count_{tIP} = power_{\text{sniffer}} \times \delta_{tIP}$。类似于未部署防御方案的系统中，随机地址池内地址数量 N 小于 $count_{tIP}$，攻击者失败率为 0，即一定能在有限的时间内嗅探成功。通过 IP 跳变策略保护的域内节点，随机地址池 N 大于 $count_{tIP}$ 时，攻击者嗅探失败率大于 0。并且随着 N 的变大，嗅探失败率变大。本文使用的随机地址池为全体 IP 空间，数量为 $2^{32}-1$，大大提高嗅探失败率。在地址空间大小 N 确定前提下，缩短更新周期同样可以提高嗅探失败率。然而更新周期缩短意味着 MTD 控制器和 OpenFlow 交换机需要更高的性能，因此，在 LAN 更新周期的选取上要考量硬件性能。

对于来自域间的攻击嗅探而言，攻击者的嗅探范围即为网络端口的空间大小，数量为 65 535。本文的端口跳变随机端口池为全端口空间，数量即为 65 535，因此，本文方法不会在端口更新周期 δ_{tPort} 内提高嗅探失败率。而是在不同的更新周期内变换端口，使攻击者无法使用固定端口进行攻击，以此提高抗攻击能力。由于端口跳变策略只作用于网关交换机，因此，更新周期的缩短不会导致严重的性能消耗增加。所以在端口跳变更新周期的选取上尽量选取小周期。

3.5.2 抗 DDoS 能力分析

参考文献[1]假定攻击者每秒发送攻击分组数量为 $count$，分组大小为 s，则每秒攻击强度为 $power_{\text{DDoS}} = count \times s$，系统使用的 IP 跳变更新周期为 δ_{tIP}，跳变地址随机地址池大小为 N。对于域内的 DDoS 攻击，每秒攻击强度分摊到地址空间，为 $power = \frac{power_{\text{DDoS}}}{N} = \frac{count \times s}{N}$。在未部署本文防御系统的域内，$N$ 即为当前网络子网掩码所代表的大小。而在部署本文防御系统后，N 即为全体 IP 空间，大大降低了平均攻击强度。

对于来自域间针对端口的 DDoS 攻击而言，由于周期变换通信端口，使攻击者无法长时间对目标服务和端口进行准确的攻击。即便在某一时刻嗅探出拟攻击的目标端口，在端口更新周期 δ_{tIP} 后，防御系统完成端口跳变，前一时刻的攻击目标端口失效，攻击失败。

4 实验与分析

本节将对本文所述移动目标防御方法通过 3 个场景进行实验和分析。

4.1 系统部署

基于本文所述的移动目标防御方法，本文构造了原型 MTD 控制器实现移动目标防御方案。Mininet 是一个用于构建软件定义网络环境的仿真平台，支持 OpenFlow 网络拓扑的构造和搭建。本文使用 Mininet 部署仿真实验环境，实验拓扑如图 5 所示。

实验中部署 2 个 LAN 区域（LAN1 和 LAN2），在 LAN1 内部署 7 个节点和 4 台 OpenFlow 交换机以及一个获取通信中间位置的中间人攻击节点，LAN2 内部署 4 个节点和 3 个 OpenFlow 交换机，具体的配置如表 4 所示。

在上述实验环境中，本节进行 3 个场景的实验，分别为抗嗅探实验、抗 DDoS 实验和全网通信实验。

4.2　实验结果与分析

本文使用 Nmap 软件对仿真网络进行网络嗅探，在实验中分别对未使用和使用本文移动目标防御方法进行保护的网络嗅探，比较和分析实验结果。

对于未使用本文所述主动防御方法的常规网络而言，Nmap 可以在短时间内检测出所有网络存活节点和及其信息。在部署了本文所述防御方法

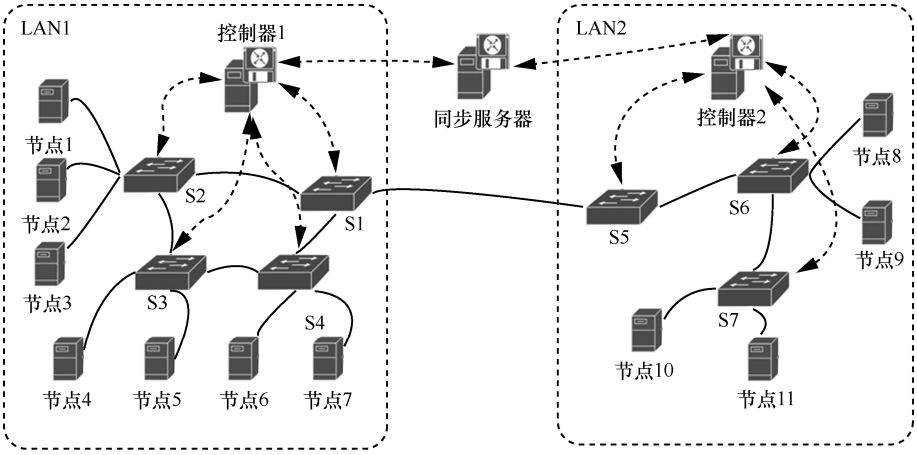


图 5　实验部署拓扑

表 4　　**实验部署参数**

设备	用途	区域	参数
同步服务器	同步控制器	域间	IP: 192.168.10.55 OS: Ubuntu 10.03
节点 1~节点 3	计算机通信节点 节点 1~节点 3 包含 3 个节点	LAN1	IP: 10.0.10.10-12 OS: Ubuntu 10.03
节点 4~节点 7	计算机通信节点 节点 4~节点 7 包含 4 个节点	LAN1	IP: 10.0.20.20-23 OS: Windows 7
S1	OpenFlow 网关交换机	LAN1	连接内网及 LAN2 内 网关交换机 S5 域间地址：172.16.0.11
S2	OpenFlow 交换机	LAN1	连接节点 1~节点 3
S3	OpenFlow 交换机	LAN1	连接节点 4~节点 5
S4	OpenFlow 交换机	LAN1	连接节点 6~节点 7
控制器 1	搭载移动目标防御组件的 OpenFlow 控制器	LAN1	IP: 10.10.20.10 控制器: Floodlight v1.2
节点 8 节点 9	计算机通信节点 节点 8、节点 9	LAN2	IP: 192.168.40.100-101 OS: Ubuntu 10.03
节点 10 节点 11	计算机通信节点 节点 10、节点 11	LAN2	IP: 192.168.50.100-101 OS: Ubuntu 10.03
S5	OpenFlow 网关交换机	LAN2	连接内网及 LAN1 内 网关交换机 S1 域间地址:172.16.0.21
S6	OpenFlow 交换机	LAN2	连接节点 8、节点 9
S7	OpenFlow 交换机	LAN2	连接节点 10、节点 11
控制器 2	搭载移动目标防御组件的 OpenFlow 控制器	LAN2	IP: 192.168.20.10 控制器: Floodlight v1.2

后，节点无法被嗅探出，达到了隐藏节点的设计需求。对于攻击者而言，由于使用了移动目标防御方法，节点在网络中使用伪造虚拟地址进行通信，因此攻击者如果扫描网段信息来识别当前活动节点，只能获取到虚拟地址。如图 6 所示，实验中对 10.0.10.0/16 网段进行扫描，该网络包括了 LAN1 中所有节点的原始网络地址。从实验结果可以看出，当部署了本文移动目标防御系统后，Nmap 无法嗅探出网络中的存活节点，扫描节点数为 0。而未部署移动目标防御系统时，扫描节点数为 7，即为部署的 7 个节点。因此，可以看出本文所述移动目标防御系统具有防攻击嗅探的功能。

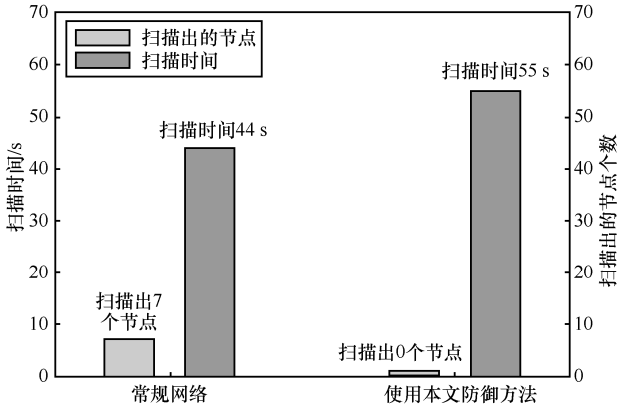


图 6 Nmap 嗅探网络结果

在实验过程中，防御系统为节点分配虚拟网络地址并节点网络地址一一对应，扫描时刻的对应关系如表 5 所示。从表中可以看出，此时节点的虚拟网络地址不在 Nmap 扫描范围内，因此，Nmap 无法扫描出节点的存在。在本文所述防御方法中，网络地址映射关系周期更新，不断产生随机地址映射对。即使攻击者通过全网段的扫描，在有限时间内检测出扫描时刻的虚拟网络地址。由于虚拟地址周期性变化，在攻击者攻击时如果使用这些当时扫描出的虚拟网络地址实施攻击，此时虚拟网络地址已经发生改变，因此攻击失效，而移动目标防御功能生效。

表 5 节点网络地址映射

节点	网络地址	虚拟地址
节点 1	10.0.10.10	13.22.16.55
节点 2	10.0.10.11	95.33.60.152
节点 3	10.0.10.12	56.28.99.98
节点 4	10.0.20.20	139.65.156.2
节点 5	10.0.20.21	66.59.239.67
节点 6	10.0.20.22	215.12.178.36
节点 7	10.0.20.23	54.33.142.233

在衡量抗攻击能力时，攻击者无法获取节点正确的网络攻击地址是一个关键。对 DDoS 攻击而言，部署本文防御系统后，由于攻击者无法获取目标节点的网络地址，因此，针对指定节点的 DDoS 攻击理论上无法实现。本文设计抗 DDoS 攻击实验假定攻击者在微小概率下成功嗅探出某时刻节点的虚拟地址，对其实施攻击。在实验中使用 Hping3 网络检测工具进行 DDoS 攻击的模拟。本文分别对节点 1 在未部署和已部署本文防御方案情况下进行 DDoS 攻击，并观察节点响应。如图 7 所示，对于常规网络中的节点 1 使用 ICMP 洪泛攻击，当每秒发送的 ICMP 数据分组少于 250 个时，响应时延几乎不变。当每秒发送大于 250 个时，响应时延快速增加。在实验中，每秒发送数据分组到达 400 时，节点 1 无法响应，拒绝服务攻击成功。而部署本文所述防御方法后，Hping3 分组发送速率到达 400 分组/秒，节点 1 仍然正常响应。并且分组发送速率达到 Hping3 发送分组上限每秒 1 000 个分组时，节点 1 上的 ICMP 服务依旧能够正常响应且保持稳定较低的时延。这是由于攻击时，节点的虚拟网络地址已经发生改变，因此，所攻击的网络地址是不存在的，攻击也就无法生效。

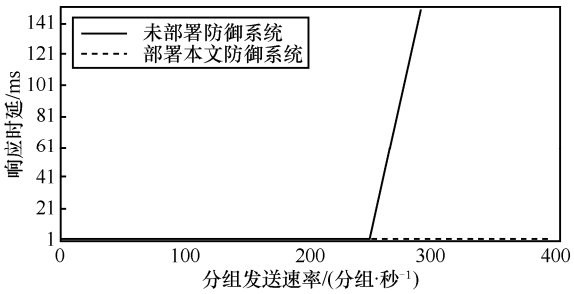


图 7 节点 DDoS 攻击后响应

全网通信作为本文设计需求之一，在实验中使用 LAN1 中的节点 1 对 LAN2 中的节点 11 发送 HTTP 请求，并在 LAN1 网关节点 3 进行抓取分组，模拟域间数据分组的截获分析。截取到的数据分组如表 6 所示。表中表示时间戳分别为 34 和 982 时，2 次节点 1 对节点 11 发送 HTTP 请求的数据分组头部信息。从分组头部信息可以看出，原本应该请求的对应 HTTP 服务的 80 端口被替换成 32 841 和 59 210 端口。从攻击者的角度，即便获取到了数据

分组是 HTTP 数据，也无法判定为同一个主机提供的服务。当攻击者用传统的网络知识判断所截获到的该数据分组时，认为在 LAN2 内部存在 2 个提供 HTTP 服务的节点。并且这 2 个节点通过 NAT 在出口网关绑定的端口分别为 32 841 和 59 210。而事实上，这 2 次数据交互均为 MTD 网关为节点 11 提供的随机通信端口。攻击者若使用这 2 个端口进行攻击，将无法实现攻击目的。

表 6　　截获数据分组头部信息

时间	源地址	源端口	目的地址	目的端口
34	172.16.0.11	12082	172.16.0.21	32841
⋮	⋮	⋮	⋮	⋮
982	172.16.0.11	8623	172.16.0.21	59210
⋮	⋮	⋮	⋮	⋮

4.3　与现有移动目标防御方案的比较

本节从移动目标防御能力、兼容性、部署方式、全网通信能力 4 个方面将本文防御方法与主流移动目标防御方案（RHM[12]、OF-RHM[13]和端信息跳变[1]）进行比较，如表 7 所示。RHM 和 OF-RHM 均基于 OpenFlow 实现 IP 跳变从而部署移动目标防御系统。不同点在于 OF-RHM 区分 IP 跳变的方式，将其分为高频与低频变换，降低 OpenFlow 控制器计算消耗。端信息跳变方式则设计基于操作系统内核的网络通信组件，从而实现移动目标防御功能。

移动目标防御能力指的是方案对于攻击的防御能力。本文方案在抗网络嗅探和抗网络攻击方面在 4.2 节进行了实验和分析，具有良好的防御能力。RHM 和 OF-RHM 均使用 OpenFlow 搭建移动目标防御体系，在移动目标防御的实现上，均使用 IP 变换。因此在部署范围内攻击者难以从窃取的数据分组分析出通信双方信息。端信息跳变则在通信时不断变换短信息实现移动目标防御功能，同样在部署范围内具有良好的防御能力。

兼容性指代对现有网络的兼容和配合能力。OF-RHM 与 RHM 使用 OpenFlow 构建移动目标防御方案，构建特有的传输协议。这种传输协议与现有网络协议难以兼容，无法部署在现有传统网络中。端信息跳变方案在主机节点部署端信息跳变的第三方库实现底层的跳变功能，因此在网络设备上，无需改变网络结构即可实现功能。而本文提出的移动目标防御方案虽然基于 OpenFlow 方案，但充分考虑与现有网络协议的兼容问题，能够与现有网络兼容。

部署方式主要比较部署移动目标防御的难易度。OF-RHM、RHM 与本文提出移动目标防御方案，直接在底层网络设备上实现移动功能，对于用户上层透明，即上层节点和用户无需知道并且改变就可以实现防御方案。而端信息跳变方案则需要在节点部署第三方库才能实现功能，对于用户和网络通信程序而言需要做出相应的修改才能实现防御功能，部署想对复杂和困难。

防御范围评价移动目标防御方案对于节点通信防御的范围，该范围包括域内和域间的通信。本文提出的移动目标防御方案不单可以部署在小范围的域内，对于跨域的通信，同样具有保护能力。而其他 3 个移动目标防御方案只能保护所部署的小范围，无法保护与外部网络的通信。

5　结束语

为了改变网络攻防博弈中攻易守难的困境，本文提出了一种基于 OpenFlow 的网络层移动目标防御方法。这种方法在攻防中实现主动防御，在网络

表 7　　与现有移动目标防御方案比较

功能	RHM	OF-RHM	端信息跳变	基于 OpenFlow 的网络层移动目标防御方案
移动目标防御方案	好 具有良好的防嗅探能力	好 具有良好的防嗅探能力	好 具有良好的防嗅探能力	好 具有良好的防嗅探能力
兼容性	差 部署时需要改变现有的网络结构	差 部署时需要改变现有的网络结构	与现有网络兼容 无需改变网络结构	与现有网络兼容 无需改变网络结构且可以保护跨域通信
部署方式	简单 上层透明，用户无需改变通信实现	简单 上层透明，用户无需改变通信实现	复杂 需要每个保护节点安装第三方通信库	简单 上层透明，用户无需改变通信实现
防御范围	小 仅限于部署的域内区域	小 仅限于部署的域内区域	小 仅限于部署的域内区域	全网保护 对于域内和跨域的通信均实现防御功能

结构和通信方式上，兼容传统网络的同时，不断修改网络通信地址，让攻击者在攻击第一步的嗅探上就增加了极大的难度，无法轻易识别攻击目标的地址和信息。不同于现有的网络层移动目标防御方案，本文在保证移动目标防御能力的同时，适配现有网络。不需改变现有网络结构，也不需在保护节点安装第三方工具和程序库，即可实现移动目标防御方案。并且全网通信保护的能力，也增强了被保护系统的安全性。实验表明，本文提出的移动目标防御方案是有效可行的。在以后的工作中，本文将会把基于移动目标防御方法的攻击分析和捕获放在研究的重点，结合蜜网相关的技术，进一步实现一个主动的网络安全防御和诱捕体系。

参考文献：

[1] 石乐义, 贾春福, 吕述望. 基于端信息跳变的主动网络防护研究[J]. 通信学报, 2008, 29(2):106-10.
SHI L Y, JIA C F, LYU S W. Research on end hopping for active network confrontation[J]. Journal on Communications, 2008, 29(2): 106-10.

[2] 蔡桂林, 王宝生, 王天佐, 等. 移动目标防御技术研究进展[J]. 计算机研究与发展, 53(5):968-987.
CAI G L, WANG B S, WANG T Z, et al. Research and development of moving target defense technology[J]. Journal of Computer Research and Development, 53(5):968-987.

[3] JACKSON T, SALAMAT B, HOMESCU A, et al. Compiler-generated software diversity[J]. Moving Target Defense, 2011: 77-98.

[4] VIKRAM S, YANG C, GU G. Nomad: towards non-intrusive moving-target defense against Web bots[C]//Communications and Network Security (CNS). 2013: 55-63.

[5] PORTOKALIDIS G, KEROMYTIS A D. Global ISR: toward a comprehensive defense against unauthorized code execution[J]. Moving Target Defense, 2011: 49-76.

[6] LUCAS B, FULP E W, JOHN D J, et al. An initial framework for evolving computer configurations as a moving target defense[C]//The 9th Annual Cyber and Information Security Research Conference. 2014: 69-72.

[7] APPLEGATE S D. The principle of maneuver in cyber operations[C]//2012 4th International Conference on Cyber Conflict (CYCON 2012). 2012: 1-13.

[8] CAI G L, WANG B S, LUO Y B, et al. Characterizing the running patterns of moving target defense mechanisms[C]// 2016 18th International Conference on Advanced Communication Technology (ICACT). 2016: 191-196.

[9] TOMMY C, XIONG K Q. Dynamic generation containment systems (DGCS): a moving target defense approach[C]//3rd International Workshop on Emerging Ideas and Trends in Engineering of Cyber-Physical Systems (EITEC). 2016: 11-16.

[10] KIRKPATRICK K. Software-defined networking[J]. Communications of the ACM, 2013.

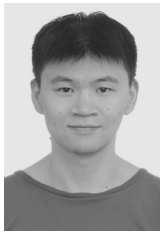
[11] MCKEOWN N, ANDERSON T, BALAKRISHNAN H, et al. OpenFlow: enabling innovation in campus networks[C]//ACM SIGCOMM Computer Communication Review. 2008: 69-74.

[12] JAFARIAN JH, AL-SHAER E, DUAN Q. OpenFlow random host mutation: transparent moving target defense using software defined networking[C]//The first Workshop on Hot Topics in Software Defined Networks. 2012: 127-132.

[13] AL-SHAER E, DUAN Q, JAFARIAN J H. Random host mutation for moving target defense[C]//SecureComm. 2012: 310-327.

[14] WANG S L, ZHANG L, TANG C J. A new dynamic address solution for moving target defense[C]//Information Technology, Networking, Electronic and Automation Control Conference. 2016: 1149-1152.

作者简介：



胡毅勋（1988-），男，江苏苏州人，北京邮电大学博士生，主要研究方向为网络安全、蜜网、SDN。

郑康锋（1975-），男，山东烟台人，北京邮电大学副教授，主要研究方向为网络与信息安全。

杨义先（1961-），男，四川盐亭人，北京邮电大学教授、博士生导师，主要研究方向为信息安全与密码学。

钮心忻（1963-），女，浙江湖州人，北京邮电大学教授、博士生导师，主要研究方向为信息安全、数字内容及安全。