基于 DPDK 的内网动态网关关键技术设计

陈福才，何威振，程国振，霍树民，周大成
（国家数字交换系统工程技术研究中心，河南 郑州 450002）

摘 要：针对 IP 跳变技术导致数据分组处理时延高、开销大的问题，基于数据平面开发套件设计并实现了一种多层次网络部署结构的主动防御网关系统。首先，基于 DPDK 快速转发框架优化了系统的转发和处理性能；其次，针对具有 3 种不同类型 IP 地址的动态化随机映射网关，设计了高效的 IP 地址分配算法以及具有不可预测性的 IP 地址变换算法。实验结果表明，所设计的系统在有效减少嗅探攻击信息获取速率的同时，大幅提升了动态跳变导致的处理时延大的问题。

关键词：主动防御；移动目标防御；IP 地址随机化；数据平面开发套件；嗅探攻击
中图分类号：TN915.08
文献标识码：A
doi: 10.11959/j.issn.1000−436x.2020126

Design of key technologies for intranet dynamic gateway based on DPDK

CHEN Fucai, HE Weizhen, CHENG Guozhen, HUO Shumin, ZHOU Dacheng
National Digital Switching System Engineering and Technological R&D Center, Zhengzhou 450002, China

Abstract: Aiming at the problems of high packet processing delay and high overhead caused by IP hopping, active defense gateway system with multi-layer network deployment structure was designed and implemented based on the data plane development kit (DPDK). Firstly, based on the DPDK fast forwarding framework, forwarding and processing performance of the system were optimized. Secondly, for the dynamic random mapping gateway with three different types of IP addresses, an efficient IP address allocation algorithm and an unpredictable IP address conversion algorithm were designed. The experimental results show that the designed system can effectively reduce the rate of information acquisition of sniffing attack, while greatly improving the processing delay caused by dynamic hopping.

Key words: active defense, moving target defense, IP address randomization, data plane development kit, sniffer attack

1 引言

内网因其管理和维护方便，可有效提高企事业单位工作效率，逐渐在政府、企业、高校等广泛应用。内网的安全在实际网络环境中非常重要，受到很多研究机构的关注。

网络侦察[1]作为网络攻击的第一阶段，通过识别网络基础设施中的潜在目标及其漏洞，为攻击者提供攻击优势。Bou-harb 等[2]指出高达 70%的攻击在发起前是有目标的扫描活动。攻击者通过嗅探网络环境获取主机开放端口信息，映射网络拓扑，进而基于已知漏洞或零日漏洞发起进一步攻击。高级攻击者的目标侦察利用先进的攻击手段对特定目标进行长期持久性网络攻击，以获得目标网络的有用信息，传统针对内网的被动式网络安全防御技术（如防火墙、入侵检测技术等）并不能阻挡此类目标侦察，例如文献[3-5]中基于计算机蠕虫的攻击方法可以规避所有的检测技术。

收稿日期：2019−12−24；**修回日期**：2020−03−26
基金项目：国家重点研发计划基金资助项目（No.2018YFB0804004）；国家自然科学基金创新研究群体基金资助项目（No.61521003）
Foundation Items: The National Key Research and Development Program of China (No.2018YFB0804004), Foundation for Innovative Research Groups of the National Natural Science Foundation of China (No.61521003)

针对传统的被动式防御技术存在的限制，移动目标防御（MTD, moving target defense）技术[6]通过主动防御的方式来适应攻击者的攻击，其目标是不断切换网络系统中的多个配置（例如，更改开放的网络端口、网络配置、软件版本等），以增加攻击者的不确定性，削弱了攻击者固有的针对传统防御机制的侦察优势。在网络层面实施动态网络防御技术是 MTD 动态化机制的一种实现方式，其通过动态化改变主机信息，包括 IP、媒体访问控制（MAC, media access control）地址、端口、网络拓扑、操作系统等，来增加攻击者探测网络环境的难度。IP 地址是标识主机信息最关键的要素之一。研究者提出 IP 地址动态化技术，主要分为传统网络实现和软件定义网络实现。

两类实现机制存在如下 2 个共性问题。首先，聚焦动态变换最大化，忽略变换造成的时延、运维等开销。虽然有些机制能够降低时延开销，如随机主机突变（RHM, random host mutation）设计了两级跳变机制降低时延，但是仍未将时延降低到合理水平[7]。其次，引入 IP 地址动态化技术，改变了传统网络结构和连接方式，如何降低对终端用户的影响以及保持系统的稳定性尚未进行充分研究。例如，IP 地址变换的前提条件是将目标网络中的终端用户进行隔离，避免广播通信，而无论是 IP 地址隔离、虚拟局域网（VLAN, virtual local area network）隔离还是交换机端口隔离，均对现有网络结构影响较大。因此，如何在不改变现有网络获取 IP 地址方式的前提下实现隔离变得尤其重要。

针对上述问题，本文基于传统网络利用数据平面开发套件（DPDK, data plane development kit）快速转发框架设计了具有高性能数据分组处理能力的动态化安全网关系统。DPDK 为 Intel 处理器架构下用户空间高效的数据分组处理提供了库函数和驱动的支持，可以大大提高数据处理性能和吞吐量，解决了传统网络因部署软件或硬件设备带来的时延问题。系统采用 DPDK 的框架，通过为主机分配外网地址（eIP, external IP）、内网真实地址（rIP, real IP）、内网虚拟地址（vIP, virtual IP）的映射表，实现内网与外网隔离，同时在真实网络不受影响的情况下，实现主机定时动态 vIP 跳变，在隐藏真实主机信息的同时，保证系统部署之后网络的稳定性。

2 相关工作

本节分别对基于传统网络和基于软件定义网络技术实现 IP 地址的现状进行具体分析。

在基于传统网络实现 IP 地址动态化方面，早期研究提出了动态网络地址转换（DyNAT, dynamic network address translation）[8]、网络地址空间随机化（NASR, network address space randomization）、基于 IPv6 的移动目标防御（MT6D, moving target IPv6 defense）[9]、RHM 等技术。DyNAT 对数据分组中涉及主机标识的分组头部分进行随机化，使攻击者难以确定数据分组的通信双方、服务类型及目标系统的位置。NASR 在网络地址动态分配的环境中，通过调节节点 IP 地址的变化频率来扰乱攻击行为。MT6D 基于 IPv6 的超大地址空间，在会话过程中不断变换发送者与接收者的 IP 地址，从而阻止攻击者发现并锁定通信主机。RHM 将网络中各个主机的 IP 地址以一种不可预测的方式进行快速变化，在不影响正常通信的条件下，有效阻止攻击者对目标网络信息的探测。文献[10]基于传统网络提出了自适应的 IP 跳变技术，以平衡跳变机制的防御优势和跳变系统的服务质量，在保证较低跳变开销的基础上，有效防御不同类型的扫描攻击。这些方法均采用动态化网络地址来干扰攻击前期的侦察过程，但是在传统网络中部署困难，需要在终端嵌入安全应用软件或者在网络中增加硬件设备，难以大规模部署。同时，这些技术没有考虑给网络带来的时延问题，分组转发效率低。

在软件定义网络实现 IP 地址动态化方面，由于软件定义网络（SDN, software defined network）[11]通过解耦数据平面和控制平面为网络提供高度开放性和可编程性，基于 SDN 的动态防御技术得到高度的重视。Jafarian 等[12]首次提出了基于 OpenFlow 随机主机突变（OF-RHM, OpenFlow RHM）技术，通过在 NOX 控制器管理的 OpenFlow 上实现 OF-RHM，其中 OpenFlow 控制器为每个主机分配随机虚拟 IP，能够有效防御主机扫描。Jafarian 等[13]在 SDN 架构的基础上提出了基于时间和空间 2 个维度的 IP 跳变技术，使攻击者在每个位置或时间间隔内频繁地重新扫描，减慢了攻击者进行攻击的进度。Sharma 等[14]提出了灵活随机的虚拟 IP 复用（FRVM, flexible random virtual IP multiplexing）技术，在软件定义网络环境中，通过多个

随机的、随时间变化的虚拟 IP 复用到真实 IP 来降低攻击者获取系统信息的准确性。文献[15]在 SDN 中不仅实现了终端的动态 IP 跳变，而且将 IP 跳变引入数据平面交换机中，并通过基于哈希链同步签名的方式同步网络路径中各节点的 IP 地址，在不产生额外开销的前提下，有效防御攻击者的目标侦察。文献[16]提出了基于 OpenFlow 网络层移动目标防御方案，在软件定义网络架构的基础上，实现不同流量的分类跳变处理。文献[17]在 SDN 中实现了 IP 与 MAC 地址协同跳变，并基于博弈模型给出了最优的跳变周期。纵观以上工作，基于 SDN 的动态防御技术已有广泛的研究，但是其 SDN 控制器一旦被攻破将泄露所有信息，且基于 SDN 的动态化防御系统通常需要多个 SDN 交换机，部署代价高昂。

3　基于 DPDK 的动态网关架构设计

本文基于 DPDK 快速转发框架设计了动态跳变的安全网关，提出的动态跳变安全网关系统需要满足以下设计要求。

1) 兼容传统网络。实现的动态网关系统能够无缝接入传统网络，且部署方便。

2) 终端透明接入。系统提出的动态化方法对终端保持透明，终端不需要安装插件和改造通信协议。

3) 网络微隔离。终端可以通过隔离的方式避免广播分组的传播，阻挡攻击者通过广播分组来获取终端的网络标识信息。

4) 降低网络时延。动态网关系统因其动态化终端网络属性，必定对网络传输带来时延，影响正常网络传输。本文方案需要解决网络时延问题，优化数据转发机制。

5) 系统稳定性。终端动态跳变网络地址需要能够不影响正常的网络通信，保持网络的稳定性。

基于上述设计限制和需求，本文设计了如图 1 所示的 DPDK 动态防御系统。动态网关部署于二层交换机与三层交换机之间，当网关托管下的终端之间进行流量转发时，流量只经过网关而不通过三层交换机，只有网关托管下的终端与网关无托管下的终端进行流量转发时，流量才经过网关之后转发到三层交换机。当终端通过动态地址分配机制获取 eIP 时，多元地址分配与动态映射算法使网关为终端分配不同网段的 rIP 和 vIP，实现安全网关托管下的终端的网段隔离。同时，网关保持 rIP 不变，基于哈希的 IP 地址管理算法动态变换 vIP，隐藏主机的真实信息，保持对终端透明。针对 IP 地址动态化带来的性能问题，DPDK 多核转发框架实现转控分离以优化转发机制，使网卡达到线速状态。

动态网关系统具体的功能架构如图 2 所示。系统分为管理层、控制层和数据层，实现了数据平面和控制平面的分离，即将控制分组和数据分组送到不同的线程进行处理，有效地提高了转发效率。

在转发线程中主要包括分组分类单元、分组转发单元。对于耗时的分组，在控制线程中完成处理，主要包括动态主机配置协议（DHCP, dynamic host configuration protocol）动态分配资源单元、地址解析协议（ARP, address resolution protocol）地址解析单元、域名系统（DNS, domain name system）域名解析单元和信息存储单元。在与转发线程和控制线程交互的管理面中主要包括系统管理单元。

1) 系统管理单元。主要负责管理员与系统的交互，通过管理员可以获得系统运行的状态，也可以

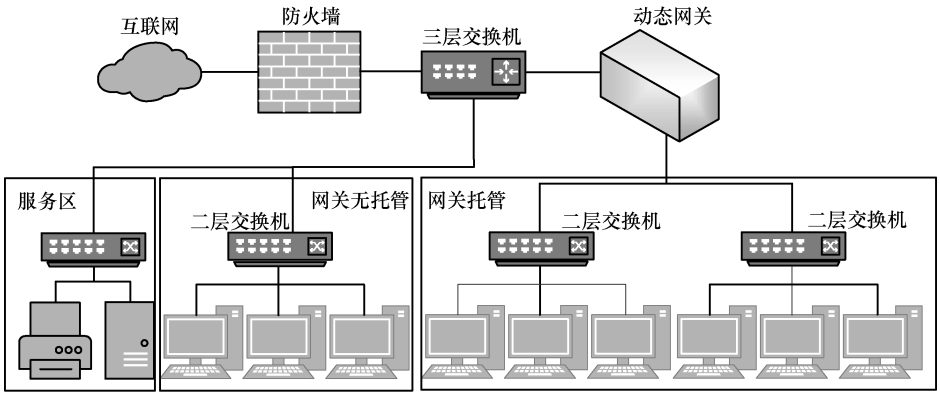


图 1　DPDK 动态防御系统部署结构

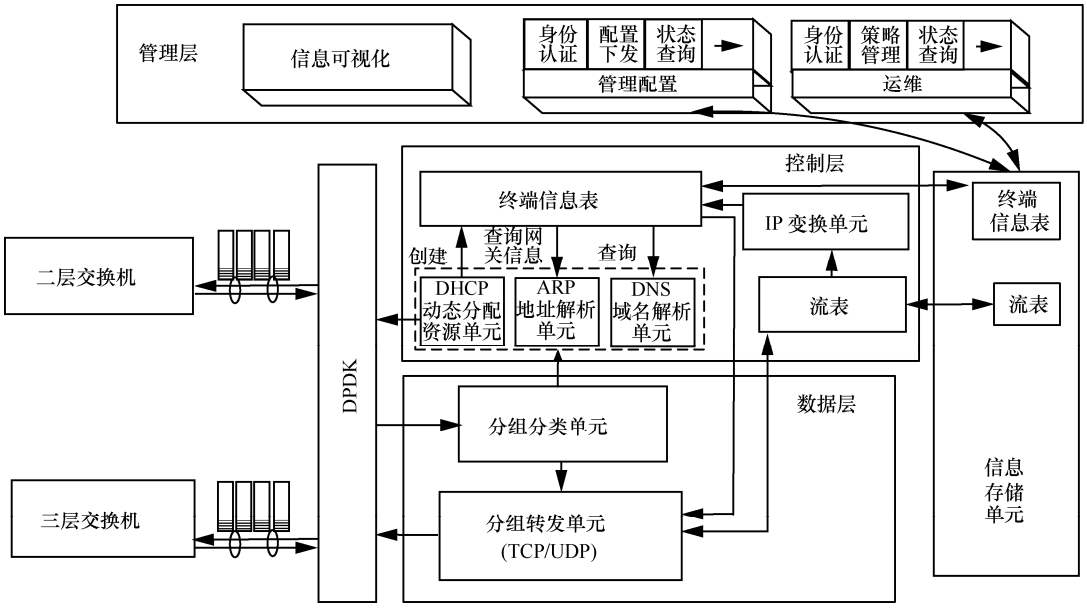


图 2 动态网关系统功能架构

下发系统配置到信息存储单元，从而实现控制面和数据面对配置消息的获取。

2) 分组分类单元。主要负责对 DPDK 绑定的网卡收到的所有流量进行分组的过滤和相关分组转发，即 DHCP、ARP、DNS、传输控制协议/用户数据报协议（TCP/UDP, transmission control protocol/ user datagram protocol）的转发。将控制分组（即 DHCP、ARP、DNS）送到控制线程处理，数据分组送到分组转发单元。

3) 分组转发单元。主要负责转发 TCP 和 UDP 的会话分组并创建流表信息以指导分组的转发规则。内网终端进行通信时需要完成 rIP 与 vIP 的替换，内网和外网终端进行通信时需要完成 rIP 与 eIP 的替换。

4) DHCP 动态分配资源单元。主要负责为终端分配 rIP 和 vIP 并创建终端信息表，包括 rIP、vIP 和 eIP 的对应关系、主机相关信息（包括 MAC 地址、域名、子网掩码、主机上线信息等）、外部三层交换机的信息等。当终端以广播的方式向 DHCP 服务器发送 DHCP discover 分组来获取终端 IP 地址时，DHCP 服务器会将携带 eIP 的 DHCP offer 分组广播给终端，网关收到 DHCP offer 分组后，分配 rIP 替换 eIP，同时分配 vIP 和 vdomain 用于动态变换。终端收到 DHCP offer 分组后，发送 DHCP request 分组进行确认，在网关完成相应修改之后发送到 DHCP 服务器，DHCP 服务器向终端发送 DHCP ack 完成终端 IP 地址的获取。

5) ARP 地址解析单元。主要负责两部分的功能。一是需要支持静态主机，如文件传输协议（FTP，file transfer protocol）服务器、打印机等不需要 IP 变换的设备，本文系统设计了通过 ARP 模块创建主机的终端信息表，使主机的 eIP、rIP、vIP 相同。二是由于终端采用网段隔离的方式，终端会发送 ARP request 的请求分组，ARP 地址解析单元负责回应终端的 MAC 地址请求。

6) DNS 域名解析单元。主要负责查询终端 vIP 和虚拟域名（vdomain），以及终端 rIP 和 vdomain 的对应关系。其原理类似于 DNS 服务器，用于回应终端的 DNS 查询。针对 DNS 域名解析单元，系统设计的最初过程中考虑了没有 DNS 域名解析单元的方案，在此方案下，终端通信之前通过网络管理员获取对端的 vIP 完成会话，但是这种方法与具有 DNS 服务器相比，每次通信之前终端都要通过网络管理员，为用户带来了额外的操作，所以本文只介绍具有 DNS 域名解析单元的方案。

7) 信息存储单元。主要负责存储 DHCP 动态分配资源创建的终端信息表和分组转发单元创建的流表信息，用于控制平面和数据平面对信息的共享。

4 动态网关关键技术设计

4.1 多通道分组分类与动态转发机制

具有 IP 跳变的安全网关由于功能的需求会对网络性能带来一定的损失，然而网络需要处理海量的数据以及支持高性能的应用程序，因此本文设计

了 2 种机制来提高数据转发性能。

1) 基于多核的多通道分组分类机制。DPDK 可以通过在多核设备上创建多个线程，每个线程绑定到单独的核上，减少线程调度的开销以提高性能，因此本文将控制平面和数据平面分离，即将控制分组和数据分组放到不同的核上进行处理。多核多通道的动态安全网关优化设计流程如图 3 所示。将 DHCP、ARP、DNS 等需要存储终端信息的控制分组在 $core_1$ 上进行处理，对于需要快速转发的数据分组在 $core_2$ 上进行处理，控制平面和数据平面通过 DPDK 提供的内存池管理机制实现信息的共享，规避不必要的内存复制和系统调用。为了进一步优化系统的处理性能，本系统设计根据会话的个数来动态调整数据平面绑定核的个数，系统在 fp_ether_input_one 中设计了流量统计接口，若流量负载超出阈值，可以动态调整数据平面绑定核的个数以增强数据平面处理能力。

2) 基于流表的动态转发机制。当会话首次建立时，主机根据终端映射表重写源主机和目的主机的相关地址信息，完成分组的转发并建立会话的流表信息，之后如果有相同主机信息的数据分组转发，主机根据流表信息重写主机信息完成转发处理，相反如果在一段时间内没有同样主机信息的分组进行转发，则对会话流表进行老化，即删除流表。然而删除流表会直接影响数据传输的稳定性，所以系统设计了网络管理员可以动态调整会话的保持时间，即对于短连接和中长连接的会话，系统会将流表保持固定的时间长度后再进行老化，在此时间长度内短连接和中长连接的会话能够稳定完成。对于长连接的会话，如大文件的传输，在会话开始之前，网络管理员通过增加合理的会话保持时间来保证数据的稳定传输。通过对活跃会话建立流表机制，避免了对每个分组都去查询规则较庞大的终端映射表，很大程度上又提升了分组的转发性能。同时，设计的会话保持机制在一定程度上解决了跨跳变周期数据传输稳定性问题。

4.2　多元地址分配与动态映射算法

在 DHCP 动态分配资源单元需要为终端分配 vIP 和 rIP，实现与 eIP 的映射关系，增强系统的安全性。因此，本系统设计了 IP 地址管理单元

安全网关托管下的终端采用网段隔离的方式避免因广播分组带来的安全问题，因此对于 rIP 资源的分配，需要考虑网段隔离且 rIP 分配的高效性，避免在分配的未使用的地址空间中出现 IP 资源的浪费，本文设计了基于哈希和多链的多元地址分配方法。rIP 地址分配如图 4 所示，需要维护真实 IP 池中 IP 的使用，即已使用的 IP 和从真实 IP 池取出但未使用的 IP。

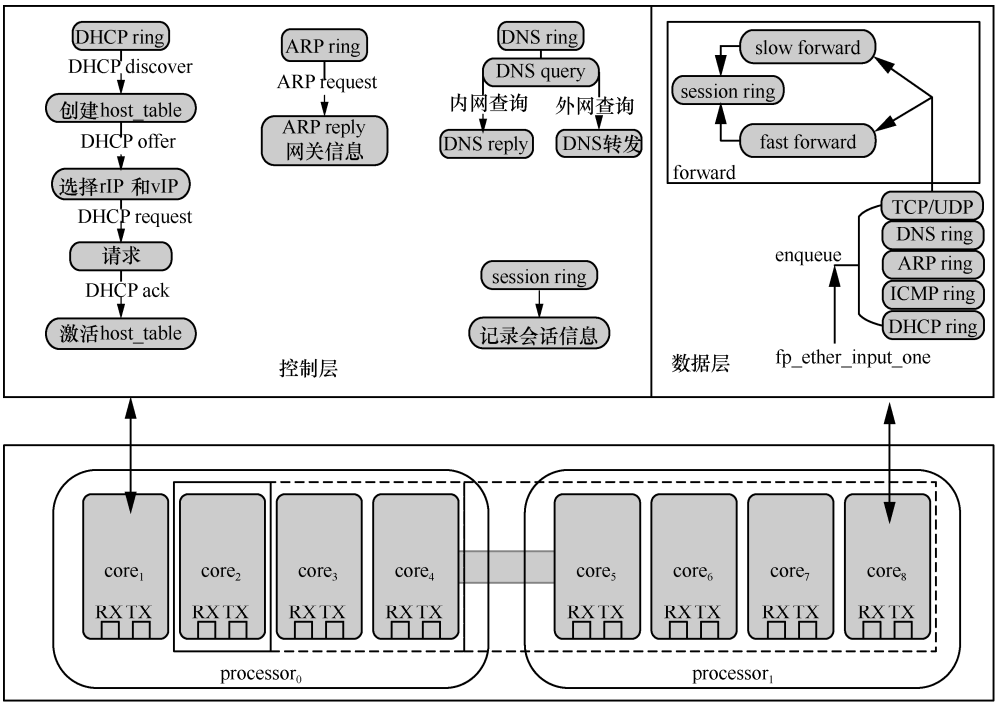


图 3　多核多通道的动态安全网关优化设计流程

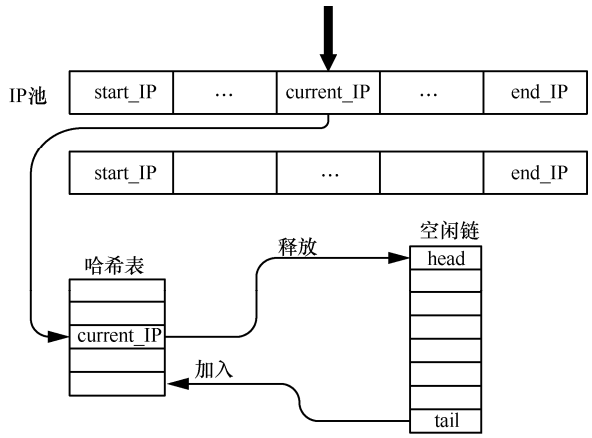


图 4 rIP 地址分配示意

如图 4 所示，当请求 rIP 资源时，会选取 IP 池中与当前 IP 地址最近但不属于同一网段的 IP 地址资源，并加入哈希表中，当哈希表中 IP 地址释放时，会将 rIP 资源放回空闲链头部，如果下次选取 IP 资源时，空闲链中的 rIP 资源大于一定的数量，会从空闲链的尾部取出 IP 资源，并加入哈希表中。具体实现如算法 1 所示。顺序选取 IP 池中的 rIP 资源和空闲链的先进先出机制保证了能够充分利用分配的 IP 池资源，且选取 rIP 资源时，不需要额外的查表操作，其时间复杂度为 $O(1)$。

算法 1 rIP 资源的分配

输入 空闲链中可用虚拟 IP 资源的个数 len，指向空闲链尾的指针 tail，指向空闲链头的指针 head，存储已用 IP 的哈希表 hash_bucket

输出 获得的虚拟地址资源 vIP

```
1)  function VIP ALLOCATION
2)      if len > MGW_LINK_NUM
3)          vIP ← tail
4)          hash_bucket ← add(vIP)
5)          return vIP
6)      end if
7)      else
8)          vIP ← current_IP
9)          hash_bucket ← add(vIP)
10)         current_IP ← current_IP + (1 << 3)
11)         return vIP
12) end function
13) function RELEASE vIP
14)     if release vIP
15)         hash_bucket ← delete(vIP)
16)         head ← vIP
17)     end if
18) end function
```

4.3 基于哈希的 IP 地址跳变算法

为防止攻击者通过目标探测获取主机的信息，可以变换主机的 IP 地址实现目标主机的不可预测性，然而仅跳变主机真实 IP 会对网络连接带来很大的开销且对终端不透明。基于 DPDK 的 IP 跳变安全网关保持终端的 rIP 保持不变，终端之间通过 vIP 完成通信，同时定时跳变 vIP 实现目标主机的不可预测性，因此本文设计了终端 vIP 的变换方法。

对于 vIP 的变换，需要考虑合适的方式最大化主机信息的不可预测性以阻挡攻击者侦察到活跃主机。首先为 vIP 的选取分配了 n 个地址空间 VPS_i（i=1,…,n），当终端进行 vIP 的变换时，随机选取一个地址空间 VPS_j（j=1,…,n），然后根据 VPS_j 的起始 IP、终止 IP 以及子网掩码分别随机得到网段和主机，将网段和主机求和之后得到 vIP，具体算法如算法 2 所示。由于终端的 vIP 不允许相同，需要存储已使用的 vIP 以避免地址冲突，考虑到查询复杂度，算法 2 使用哈希表存储已用的 vIP，每次选取 IP 之后都需要查找哈希表，防止选取的真实 IP 资源重复，其时间复杂度为 $O(1)$。

算法 2 vIP 资源的分配

输入 rIP 地址段 rIP[j][2]，子网掩码 bitmask，子网掩码为零的个数 offset

输出 获得的真实地址资源 rIP

```
1) for j = 1 : n
2)     if rIP[j] is not NULL
3)         rIP[0] ← rIP[j][0];
   rIP[1] ← rIP[j][1];
4)     end if
5) end for
6) function RIP ALLOCATION
7)     while true do
8)         size ← ((rIP[1] & bitmask) >> offset)
– ((rIP[0] & bitmask) >> offset)
9)         host ← (rIP[1] & ~bitmask) – (rIP[0]
& bitmask)
10)        IP ← rIP[0] + (random()%size <<
offset) + host
11)        if  hash_table not exist IP
12)            hash_table ← add(size)
13)            return IP
```

14)　　end if

15)　end while

16) end function

用 t_c 表示获取 vIP 资源的时间戳，该时间戳通过调用系统时间获得，利用随机函数生成在地址空间中的随机网段和主机。与时间戳 t_c 和地址空间 VPS_j 相关的 vIP 的获取函数 $F_c(\cdot)$可表示为

$$\mathrm{vIP}_c = F_c(t_c, \mathrm{VPS}_j)$$

4.4　多层次网络交互流程

基于 IP 跳变机制的安全网关托管下的终端采用网段隔离的方式实现安全性，不同于基于 SDN 的功能流程[18]，本文系统需要利用 rIP、eIP 和 vIP 的对应关系完成网关托管下的终端之间以及与网关无托管下终端的多层次网络通信，这里将网关托管下的终端称为内网终端，网关无托管下的终端称为外网终端。当内网终端之间进行通信时，采用 vIP 进行通信，当内网与外网进行通信时，采用 eIP 进行通信。完整的通信流程如图 5 所示，内网终端之间和内外网终端之间共有 3 种通信方式。

1) 内网访问外网的通信

对于内网和外网的通信，其主要步骤如图 5 所示。

步骤 1　与内网终端通信相似，终端 1 首先需要通过 DHCP 方式获取 3 种 IP 地址且向网关发送请求 MAC 的分组，然后请求网关 DNS 服务器获取外网终端 3 的域名。

步骤 2　终端 1 根据得到的终端 3 的域名，获得终端 3 的外网 IP，即 eIP3。

步骤 3　终端 1 将 rIP1 作为自己的源 IP，将终端 3 的 eIP3 作为目的 IP，然后根据此传递分组。

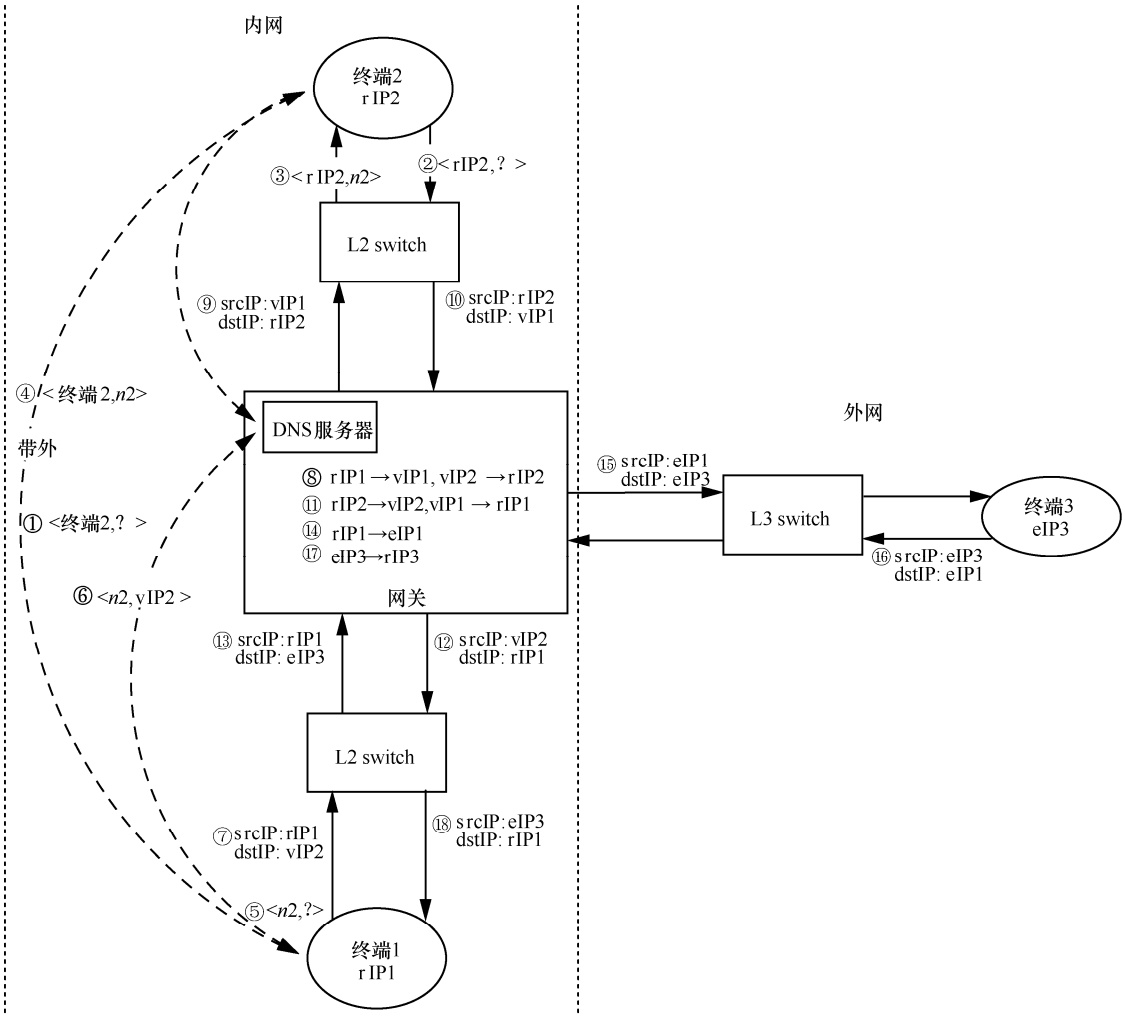


图 5　安全网关托管下的内外网终端之间的通信流程

步骤 4 当分组到达网关时，通过查询终端信息表将分组中的源 IP 修改为 eIP1 后，将分组送到终端 3。

2) 外网访问内网的通信

对于外网访问内网，由于外网终端不在安全网关的托管下，其 IP 地址仅有 eIP，通信流程与上述方式略有差别，其具体步骤如图 5 所示。

步骤 1 终端 3 首先获取外网终端 1 的域名。

步骤 2 终端 3 根据得到的终端 1 的域名，获得终端 1 的外网 IP，即 eIP1。

步骤 3 终端 3 将 eIP3 作为自己的源 IP，将终端 1 的 eIP1 作为目的 IP，然后根据此传递分组。

步骤 4 当分组到达托管终端 1 的网关时，将分组中的目的 IP 修改为 rIP1 之后，将分组送到终端 1。

3) 内网内部通信

内网终端之间通过 vIP 进行通信，在通信过程中接入端需要完成 rIP 和 vIP 的替换，其具体步骤如图 6 所示。

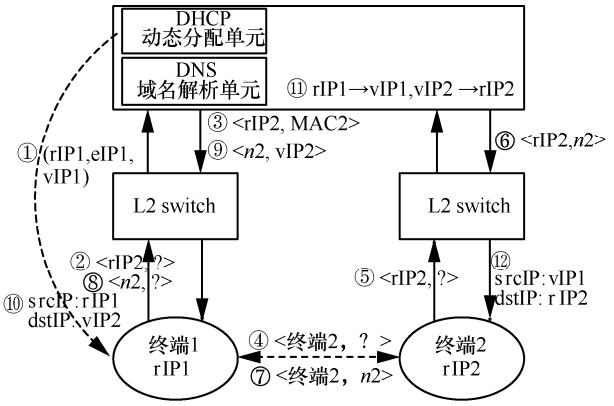


图 6 安全网关托管下的内网终端之间的通信流程

步骤 1 终端 1 通过 DHCP 的方式获取终端的 eIP1，网关接收终端与 DHCP 服务器交互的所有报文，分配与 eIP1 对应的 rIP1、vIP1 并创建终端信息表，如图 6 中的①所示。

步骤 2 终端 1 通过终端 2 的 rIP2 向网关发送 ARP 请求以获取网关的 MAC 地址，如图 6 中的②和③所示。

步骤 3 终端 1 通过带外的方式向 DNS 服务器发送查询请求获取终端 2 的虚拟域名 *n*2，如图 6 中的④~⑦所示。

步骤 4 终端 1 根据 DNS 服务器回复的虚拟域名 *n*2 获取终端 2 的虚拟 IP，即 vIP2，如图 6 中的⑧和⑨所示。

步骤 5 终端 1 把得到的终端 2 的 vIP2 作为目的 IP，把终端 1 的 rIP1 作为源 IP，然后据此建立本次通信，如图 6 中的⑩所示。

步骤 6 当通信数据分组到达网关时，网关根据通信数据分组的源 IP 查询 DHCP 模块创建的终端信息表，将源 IP 修改为与终端 1 对应的虚拟 IP，即 vIP1，将通信数据分组的目的 IP 修改为与目的 IP 对应的终端 2 的真实 IP，即 rIP2，然后将通信数据分组送到终端 2，如图 6 中的⑪和⑫所示。

步骤 7 终端 2 根据同样的步骤 5 和步骤 6 将通信分组返回终端 1。

5 安全性及系统稳定性分析

5.1 抗嗅探安全性分析

本文提出的基于 DPDK 的内网动态安全网关系统主要针对攻击者的目标侦察阶段，能够有效延缓攻击者获取内网信息的速率，阻断网络攻击链，因此本文将抗嗅探能力作为评价系统安全能力的指标。

攻击者通常使用 Nmap 等扫描工具向目标网络发送探测数据分组以检测是否存在活跃主机，本文定义攻击者的嗅探能力为每秒向网络发送探测数据分组的个数 ρ，给定时间 τ，攻击者进行探测的总数为 $T=\rho\tau$。在没有部署内网动态安全网关系统时，只要给定充足的时间，攻击者总能探测到内网所有活跃的主机。

当部署内网动态安全网关时，由于终端的 vIP 动态变换，攻击者收集的信息会在下一跳变周期变得无效，因此在一个跳变间隔 δ_t 内攻击者进行探测的总数 R 是有限的，$R=\rho\delta_t$。假设在动态网关托管下存在一台活跃主机和一台扫描主机，为终端分配 vIP 的地址空间为 Ω，当 $\Omega \leqslant R$ 时，攻击者在一个跳变间隔便能扫描整个地址空间，扫描器发送第 i 次扫描时，扫描到真实主机的概率为

$$P_{\text{detect\_succ}}=\frac{1}{\Omega-i} \tag{1}$$

给定 n 次扫描 $n<R$，未扫描到真实主机的概率为

$$P_{\text{detect\_fail}}=\prod_{i=0}^{n-1}\left(1-\frac{1}{\Omega-i}\right) \tag{2}$$

当 $\Omega>R$ 时，攻击者不能在一个跳变间隔内都整个地址空间进行扫描，共需 $\left\lfloor\frac{\Omega}{R}\right\rfloor$ 个跳变间隔和

$\Omega \bmod R$ 次扫描。对于前 $\left\lfloor \frac{\Omega}{R} \right\rfloor$ 个跳变间隔，未扫描到真实主机的概率为

$$P_{\text{detect\_fail}} = \left(\prod_{i=0}^{R-1} \left(1 - \frac{1}{\Omega - i} \right) \right)^{\left\lfloor \frac{\Omega}{R} \right\rfloor} \tag{3}$$

根据式(2)和式(3)，当 vIP 变换的地址空间为 Ω 时，攻击者成功探测活跃主机的概率可以表示为

$$P_{\text{succ}} = \begin{cases} 1 - \prod_{i=0}^{R-1} \left(1 - \frac{1}{\Omega - i} \right), & \Omega \leqslant R \\ 1 - \left(\prod_{i=0}^{R-1} \left(1 - \frac{1}{\Omega - i} \right) \right)^{\left\lfloor \frac{\Omega}{R} \right\rfloor} \prod_{i=0}^{\left\lfloor \frac{\Omega}{R} \right\rfloor - 1} \left(1 - \frac{1}{\Omega - i} \right), & \Omega > R \end{cases} \tag{4}$$

从式(4)可以看出，一个跳变间隔 δ_t 内攻击者进行探测的总数 R 越小，即跳变间隔 δ_t 越小，vIP 变换的地址空间为 Ω 越大，攻击者成功的概率越小。

5.2　系统稳定性分析

本文提出的动态安全网关在保证具有良好安全性的同时，需要保证系统能够稳定运行，不影响正常的网络通信，因此，第 3 节中的多种设计机制充分考虑了系统稳定性。其中，采用网段隔离的方式来代替端口隔离和 VLAN 隔离，在最小化改变网络结构的基础上防止因广播分组的传播而带来安全问题。同时，系统采用 3 种 IP 的随机映射，通过保持 rIP 和 eIP 不变，动态跳变 vIP 来保持连接稳定性，避免因动态跳变真实 IP 带来严重的连接中断。

此外，本文对动态 IP 的选择算法进行仿真，验证了虚拟 IP 的动态变化不会影响系统的波动。如图 7～图 9 所示，选取在 255、6 000 和 10 000 个地址空间下，通过对地址随机选取 30 000 次得到每个地址使用的概率分布。可以看出，3 种地址空间下动态地址分布基本服从均匀分布，(即 255 个地址空间概率密度函数约为 $\frac{1}{255} \approx 0.39\%$，6 000 个地址空间概率密度函数约为 $\frac{1}{6\,000} \approx 0.016\%$，10 000 个地址空间概率密度函数约为 $\frac{1}{10\,000} = 0.01\%$)，所以无论地址空间怎么变化，对于每个 IP 地址的使用概率均相同，不会因为地址空间的波动，对系统稳定造成影响。

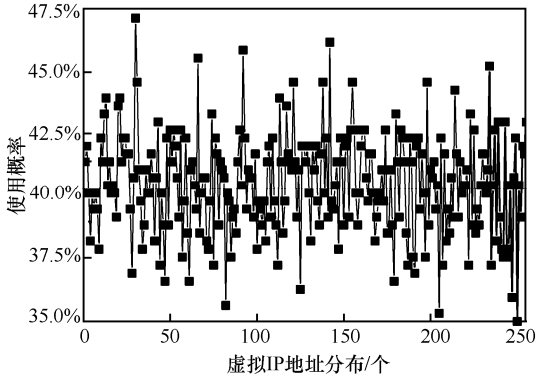


图 7　255 个地址空间状态下地址使用概率分布

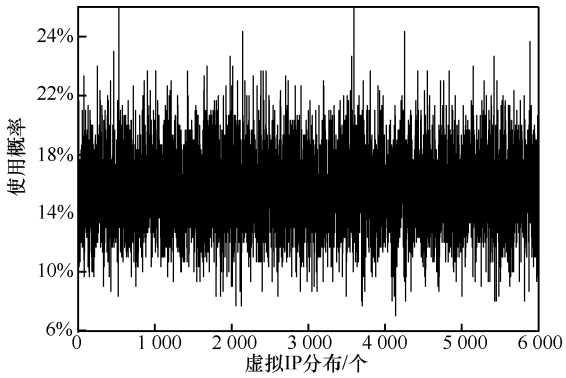


图 8　6 000 个地址空间状态下地址使用概率分布

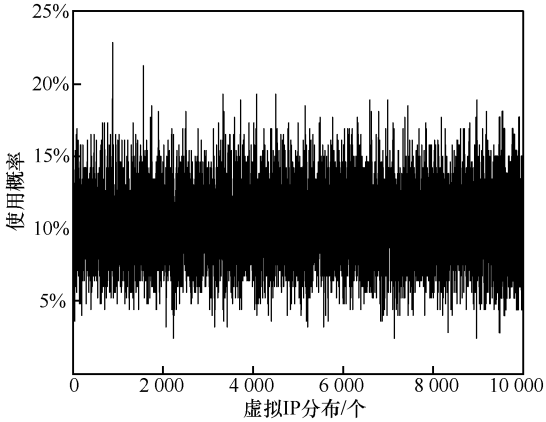


图 9　10 000 个地址空间状态下地址使用概率分布

6　实验及分析

6.1　实验场景

为了评估系统在真实网络环境中的安全性和性能，本文搭建了如图 10 所示的测试环境。测试环境中包含 45 台实体主机、4 台二层交换机、一台三层交换机，系统部署于二层交换机与三层交换机之间，所有流量均需通过系统。在主机上分别安装 Window 7、Centos 7、Ubuntu 16.04.1、Kail Linux 系统，与交换机 A 相连的主机安装 4 台 Window 7、

4 台 Centos 7、4 台 Ubuntu 16.04.1，与交换机 B 相连的主机安装 4 台 Window 7、4 台 Centos 7、3 台 Ubuntu 16.04.1，与交换机 C 相连的主机安装 4 台 Window 7、3 台 Centos 7、4 台 Ubuntu 16.04.1，与交换机 D 相连的主机中，4 台为 Window 7 操作系统、4 台为 Centos 7 操作系统、2 台为 Ubuntu 16.04.1 操作系统、一台为 Kail Linux 操作系统。

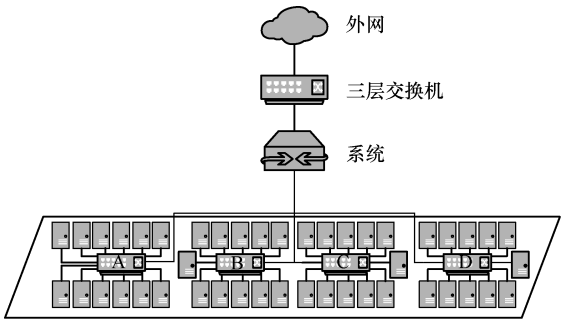


图 10 DPDK 动态防御系统测试环境

6.2 抗嗅探性能评估

对于针对性的网络攻击，如高级持续性攻击（APT, advanced persistent threat）依赖目标侦察来收集目标系统的信息用于进一步的攻击计划。本文提出的基于 DPDK 的动态安全网关通过动态跳变终端的网络地址来延缓攻击者通过网络扫描获取终端信息的速率，阻断网络攻击的连续性。因此针对系统的抗嗅探性能，本文采用扫描时间的长短进行评估。

针对系统防御网络扫描的性能，本文在 Kali Linux 主机上使用 Nmap 扫描工具对包括其他 44 台主机所在的虚拟 IP 地址段的 C 类网络地址进行扫描，通过设置不同的动态跳变频次（即每小时分别跳变 0、2、5、10、15、20、25、30、40、48、60 次），获得扫描所需要的时间，结果如图 11 所示。可以看到，随着跳变频次的增高，扫描时间越来越长。这是由于跳变频次增高以后，针对特定地址段留给攻击者的扫描处理时间越来越短，扫描成功率随之下降，系统可以有效延缓攻击者进行网络嗅探获取内网信息的效率。因此在系统托管的网络环境中，可以通过增加跳变频次提高系统的安全性能。然而跳变频次越高，因维护跳变带来的连接中断的开销越大，因此选择合适的跳变周期尤其重要。从图 11 的扫描实验可以看出，当跳变频次在每小时 15~30 次（即跳变周期在 2~4 min）时扫描 C 类网段所需的时间增加最快，而扫描频次小于每小时 15 次或大于每小时 30 次（即跳变周期小于 2 min 或大于 4 min）时扫描 C 类网段所需时间增速缓慢，考虑跳变带来的开销，因此将最优的跳变周期定为 2 min。虽然跳变周期为 2 min 时不能达到最优的安全性能，但是在系统地址动态跳变的环境下，即使攻击者能够扫描到主机的 vIP，但其在进一步发起攻击时，主机的 vIP 可能已经变化，迫使攻击者需要重新开始扫描，亦即其攻击链上各阶段的成果很难继承，攻击链很难完整地建立。

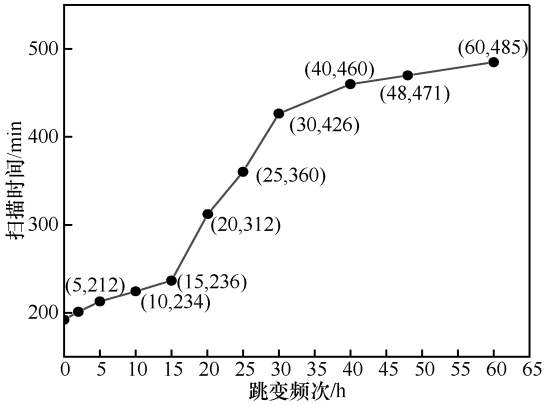


图 11 跳变次数与扫描时间关系

为了进一步验证系统能够有效延缓攻击者获取信息的速率，实验中采用 3 种扫描策略对网络环境进行扫描，即均匀扫描策略、顺序扫描策略、协同扫描策略。

1) 均匀扫描策略。攻击者在给定地址空间 Ω 中随机发送探测消息，在网络环境中，攻击者探测漏洞主机的概率是相同的。

2) 顺序扫描策略。在给定起始地址和终止地址之后，攻击者以递增的方式探测地址空间。

3) 协同扫描策略。多个攻击者以并行的方式对地址空间进行扫描并共享扫描结果。

本文为了实现上述 3 种扫描策略，使用 libnamp 来定制 3 种扫描策略，libnamp 是一个 Python 库，能够自动安排 Nmap 扫描，操纵 Nmap 扫描结果进行报告。实验选取根据图 11 分析得到的最优跳变周期，即跳变周期为 2 min，对比 DPDK 安全网关、无安全网关和 SDN 安全网关[18] 扫描 C 类网段，3 种扫描策略扫描到 45 台主机的 vIP 的检测概率与扫描时间的关系，得到的结果如图 12～图 14 所示。

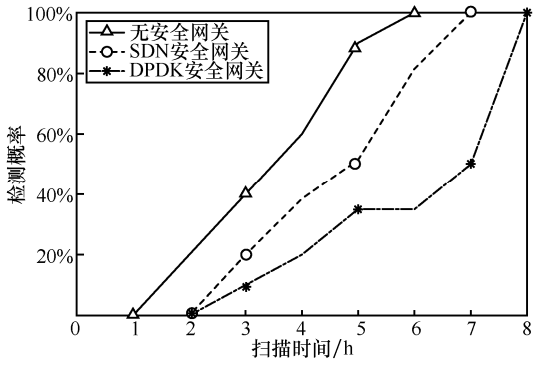


图 12　顺序扫描策略时扫描时间与检测概率的关系

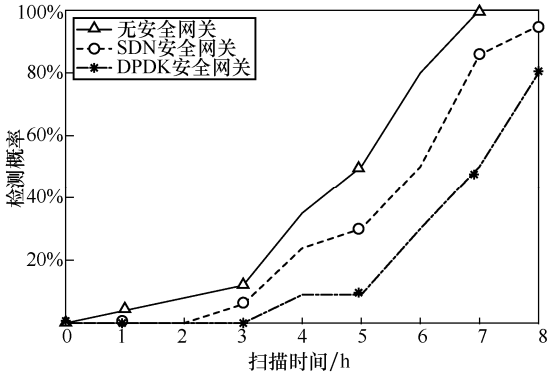


图 13　均匀扫描策略时扫描时间与检测概率的关系

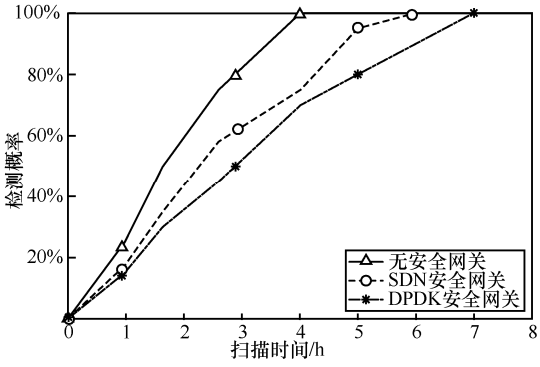


图 14　协同扫描策略时扫描时间与检测概率的关系

可以看出，无论攻击者以何种扫描策略进行扫描动态安全网关托管下的网络环境，系统均能延缓攻击者获取信息的速率，且基于 DPDK 的动态安全网关系统相当于 SDN 网关在阻挡攻击者获取信息的能力上具有一定的优势。

6.3　时延性能评估

本文系统旨在利用 DPDK 高性能的数据处理能力抵消系统中动态化带来的性能损失。针对系统的时延开销，本文采用 Sprient Test Center 测试仪对系统 1 GB 网卡的吞吐量进行了测试。通过在测试仪上添加 2 台主机，分别构造了 64～128 B 的数据分组进行打流测试，实验结果与基于 SDN 的动态化防御系统和 H3C 交换机进行对比，网络时延测试结果如图 15 所示。可以看出，由于 DPDK 高效的数据分组收发方式以及基于 DPDK 的多核平台转控分离的设计，系统的时延明显低于基于 SDN 的动态化防御系统，时延开销与普通交换机的时延差别不大。实验结果显示，对于大于 128 B 的分组，网关数据分组转发速率为线速；对于 64 B 的分组，网关数据分组转发速率为线速的 80%，数据分组平均时延低于 40 μs。

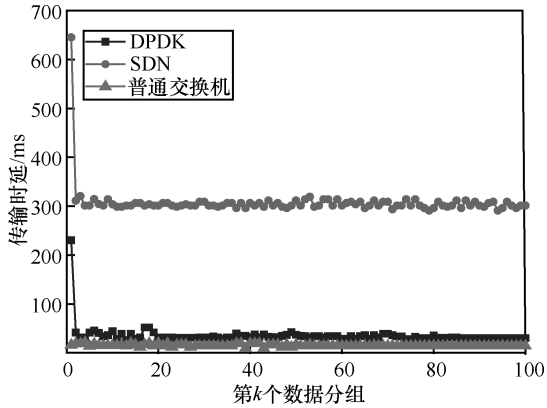


图 15　网络时延测试

图 15 中在第一个数据分组通过系统时，出现一个尖峰，这是因为第一个数据分组通过系统时，需要进行建流表操作，所以第一个数据分组通过系统时会出现一个很高的时延，之后系统的时延会趋于稳定。

6.4　系统稳定性评估

在实验过程中，本文将时延作为评价系统稳定性的指标。系统稳定性验证的实验环境如图 16 所示，在系统保持动态跳变的基础上，验证 2 个终端之间进行文件传输时所需要的时间以及完整性。

如图 16 所示，使用 FTP 文件传输工具 FileZilla 在终端上搭建客户端和服务端，其中服务端不在网关的托管下，保持网络地址不变，而客户端在网关的托管下，动态跳变网络地址。通过客户端分别向 FTP 服务器传输不同大小的文件（即文件大小分别为 10 MB、30 MB、100 MB、300 MB、600 MB、1 024 MB），对比无安全网关、DPDK 安全网关和 SDN 安全网关传输文件所需要的时间，结果如图 17 所示。从实验结果可以看出，DPDK 安全网关在传输文件时与无安全网关所用时间相差不大，且在传输大文件时，DPDK 安全网关所用时间明显低于 SDN 安全网关。

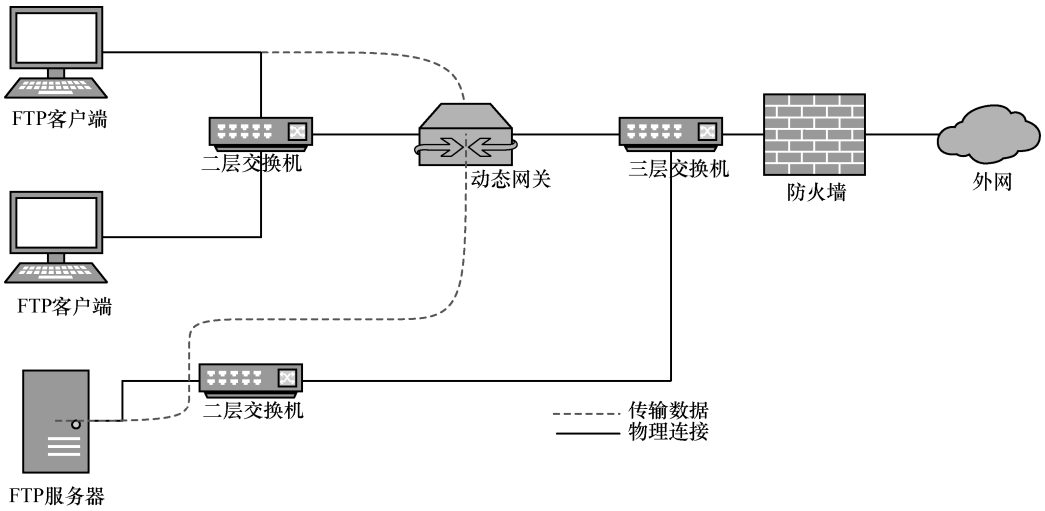


图 16 系统稳定性验证的实验环境

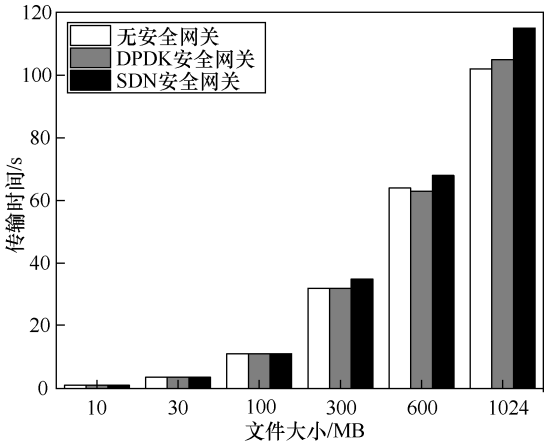


图 17 文件传输时间与文件大小的关系

为了进一步验证 DPDK 安全网关存在的情况下文件传输的完整性，本节对比了客户端传输的文件与服务端接收到的文件，文件大小的一致性有效地说明了系统的稳定性。

7 结束语

本文针对内网防护存在的安全问题，设计并实现了基于 DPDK 的内网动态网关系统来防御攻击者的目标侦察，该系统通过为终端分配 rIP、vIP、eIP 实现终端隔离，通过随机且不可预测的跳变主机 vIP 地址来降低扫描的准确性。首先基于高性能处理能力的 DPDK 框架介绍了系统的架构，接着根据系统架构，设计并实现了 rIP、vIP 的分配算法，给出了终端之间的通信流程。实验结果表明，本文系统能够有效延缓攻击者进行网络扫描，相比于基于 SDN 框架的动态防御系统，本文系统的数据传输时延具有明显优势。今后的工作将围绕 IP 跳变周期，给出适应性跳变策略；进一步完善系统功能，增加虚拟节点功能，实现对攻击者的欺骗感知。

参考文献：

[1] YADAV T, RAO A M. Technical aspects of cyber kill chain[C]// Proceedings of the International Symposium on Security in Computing and Communication. Berlin: Springer, 2015: 438-452.

[2] BOU-HARB E, DEBBABI M, ASSI C. Cyber scanning: a comprehensive survey [J]. IEEE Communications Surveys & Tutorials, 2013, 16(3): 1496-1519.

[3] WEAVER N, PAXSON V, STANIFORD S, et al. A taxonomy of computer worms [C]// Proceedings of the 2003 ACM Workshop on Rapid malcode. New York: ACM Press, 2003: 8-11.

[4] ANTONATOS S, AKRITIDIS P, MARKATOS E P, et al. Defending against hitlist worms using network address space randomization [J]. Computer Networks, 2007, 51(12): 3471-3490.

[5] AI J, GUO Z, CHEN H. Thwarting worm spread in heterogeneous networks with diverse variant placement [J]. IEEE Communications Letters, 2018, 22(7): 1346-1349.

[6] CHO J-H, SHARMA D P, ALAVIZADEH H, et al. Toward proactive, adaptive defense: a survey on moving target defense [J]. IEEE Communications Surveys & Tutorials, 2020, 22(10): 709-745.

[7] JAFARIAN J H, AL-SHAER E, DUAN Q. An effective address mutation approach for disrupting reconnaissance attacks [J]. IEEE Transactions on Information Forensics and Security, 2015, 10(12): 2562-2577.

[8] KEWLEY D, FINK R, LOWRY J, et al. Dynamic approaches to thwart adversary intelligence gathering[C]// Proceedings of the DARPA Information Survivability Conference and Exposition II DISCEX'01. Piscataway: IEEE Press, 2001: 176-185.

[9] DUNLOP M, GROAT S, URBANSKI W, et al. Mt6d: a moving target ipv6 defense; proceedings of the 2011-MILCOM[C]// 2011 Military

Communications Conference. Piscataway: IEEE Press, 2011: 1321-1326.

[10] LEI C, ZHANG H Q, MA D H, et al. Network moving target defense technique based on self-adaptive end-point hopping[J]. Arabian Journal for Science and Engineering, 2017, 42(8): 3249-3262.

[11] REHMANI M H, DAVY A, JENNINGS B, et al. Software defined networks-based smart grid communication: a comprehensive survey[J]. IEEE Communications Surveys & Tutorials, 2019, 21(3): 2637-2670.

[12] JAFARIAN J H, AL-SHAER E, DUAN Q. Openflow random host mutation: transparent moving target defense using software defined networking[C]// Proceedings of the First Workshop on Hot Topics in Software Defined Networks. New York: ACM Press, 2012: 127-132.

[13] JAFARIAN J H H, AL-SHAER E, DUAN Q. Spatio-temporal address mutation for proactive cyber agility against sophisticated attackers[C]// Proceedings of the First ACM Workshop on Moving Target Defense - MTD '14. New York: ACM Press, 2014: 69-78.

[14] SHARMA D P, KIM D S, YOON S, et al. FRVM: flexible random virtual ip multiplexing in software-defined networks[C]// 2018 17th IEEE International Conference On Trust, Security and Privacy in Computing and Communications/ 12th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE). Piscataway: IEEE Press, 2018: 579-587.

[15] CHANG S Y, PARK Y, BABU B B A. Fast IP hopping randomization to secure hop-by-hop access in sdn [J]. IEEE Transactions on Network and Service Management, 2018, 16(1): 308-320.

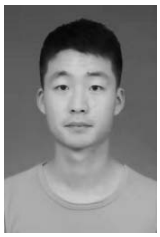
[16] 胡毅勋, 郑康锋, 杨义先, 等. 基于OpenFlow的网络层移动目标防御方案[J]. 通信学报, 2017, 38(10):103-112.
HU Y X, ZHENG K F, YANG Y X, et al. Moving target defense solution on network layer based on OpenFlow[J]. Journal on Communication, 2017, 38(10): 103-112.

[17] 王鹏超, 陈福才, 程国振, 等. 软件定义的 L2/L3 地址协同拟态伪装策略研究[J]. 电子学报, 2019, 47(10): 2032-2039.
WANG P C, CHEN F C, CHENG G Z, et al. L2/L3 address cooperative mimicry strategy research based on SDN[J]. Acta Electronica Sinica, 2019, 47(10): 2032-2039.

[18] 陈扬, 扈红超, 程国振. 软件定义的内网动态防御系统设计与实现[J]. 电子学报, 2018, 46(11): 2604-2611.
CHEN Y, HU H C, CHENG G Z. The design and implementation of a software-defined intranet dynamic defense system[J]. Acta Electronica Sinica, 2018, 46(11): 2604-2611.

[作者简介]

陈福才（1974– ），男，江西高安人，国家数字交换系统工程技术研究中心研究员，主要研究方向为网络通信、网络安全。



何威振（1996– ），男，安徽亳州人，国家数字交换系统工程技术研究中心硕士生，主要研究方向为网络安全。

程国振（1986– ），男，山东菏泽人，博士，国家数字交换系统工程技术研究中心副教授，主要研究方向为云数据中心、SDN、网络安全。

霍树民（1985– ），男，山西长治人，博士，国家数字交换系统工程技术研究中心副研究员，主要研究方向为网络安全。

周大成（1996– ），男，河南信阳人，国家数字交换系统工程技术研究中心硕士生，主要研究方向为网络安全、SDN。