PrNet:一种应对链路洪泛攻击的机制

杨智威,林梓钘,李 睿
(东莞理工学院 网络空间安全学院,广东 东莞 523808)

摘 要:网络拓扑的特性造成了拓扑中会出现大多数流量汇聚到少部分关键节点和链路的情况,这部分节点和链路会成为链路洪泛攻击所针对的网络瓶颈。现有的防御工作主要围绕隐藏网络瓶颈展开,但对于网络瓶颈的计算度量标准较为单一,且无法应对攻击者发起的盲攻击。为了解决这些问题,提出了一种基于 SDN 的应对机制 PrNet。PrNet 首先从静态和动态的角度定义了形成网络瓶颈的度量指标,然后生成针对测绘流量的混淆拓扑,通过识别测绘流量并将其引向绕开网络瓶颈的混淆路径,使攻击者得到错误的信息,最后通过概率路径转发算法为节点之间的所有可达路径分配概率,主动分散网络拓扑中的流量,从而减少网络瓶颈的产生。仿真实验表明,PrNet 能够生成具有良好安全性的混淆拓扑,能够根据流量及时调整数据包的转发路径,在应对攻击者发起链路洪泛攻击时具有可行性,并且能够有效缓解盲攻击。

关键词:链路洪泛攻击;网络瓶颈;拓扑混淆;流量分散;软件定义网络
中图分类号:TP393 **文献标识码**:A **文章编号**:1673-629X(2023)08-0108-08
doi:10.3969/j.issn.1673-629X.2023.08.016

PrNet:A Mechanism to Counteract Link Flooding Attacks

YANG Zhi-wei,LIN Zi-xing,LI Rui
(School of Cyberspace Security,Dongguan University of Technology,Dongguan 523808,China)

Abstract:The nature of the network topology causes the situation that most traffic in the topology converges to a small number of critical nodes and links,which become network bottlenecks targeted by link flooding attacks. Existing defense works focus on hiding network bottlenecks,but their calculation metrics for network bottlenecks are relatively single and cannot cope with blind attacks launched by attackers. We propose an SDN-based mechanism called PrNet to solve the above problems. The metrics that form network bottlenecks from both static and dynamic perspectives is defined,then an obfuscation topology for mapping traffic is generated,which gives attackers misinformation by identifying mapping traffic and directing it to an obfuscation path that bypasses network bottlenecks. Finally,the probabilistic path forwarding algorithm assigns probabilities to all reachable paths between nodes and actively disperses the traffic in the network topology,thus reducing the generation of network bottlenecks. The simulations shows that PrNet can generate an obfuscated topology with good security and can adjust the forwarding path of packet in time according to the traffic,which is feasible in response to attackers launching link flooding attacks,and can effectively mitigate blind attacks.

Key words:link flooding attack;network bottleneck;topology obfuscation;traffic dispersion;software defined network

0 引 言

分布式拒绝服务(Distribution Denial of Service,DDoS)攻击是当今互联网主要的威胁之一[1],其主要通过控制大量僵尸主机向目标节点发送大量数据包,消耗目标节点的资源,以影响合法用户的正常使用。例如,在 2016 年 Mirai 恶意软件利用了受损的 IoT 设备向 DNS 提供商 Dyn 发起的 DDoS 攻击,造成了美国大规模网络瘫痪[2]。而链路洪泛攻击(Link-Flooding Attack,LFA)[3]作为新型的 DDoS 攻击,不同于以服务器为目标的传统 DDoS 攻击,LFA 通过控制大量僵尸主机向网络中的关键路由器和关键链路注入大量低速率流量,目的在于切断尽可能多的网络连接,破坏合法用户与服务器之间的正常通信。由于 LFA 发送的是具有真实 IP 地址的低速率流量,这种流量与合法流量的特征基本一致,因此诸如检测虚假 IP 地址和特定签名[4]之类的传统对策是不起作用的,与传统 DDoS 攻击相比,LFA 具有更强的隐蔽性。

根据网络拓扑的特性,节点之间的连接并不是均

收稿日期:2022-11-04 **修回日期**:2023-03-08
基金项目:国家重点研发计划(2021YFB3101300);国家自然科学基金面上项目(61972089)
作者简介:杨智威(1997-),男,硕士研究生,研究方向为软件定义网络、网络空间安全;通讯作者:李 睿(1975-),男,教授,博士,研究方向为隐私保护、网络空间安全。

匀分布的。无论规模如何,一个片区中大多数节点的流量会汇聚到少部分关键节点和链路上,通过关键节点和链路与其他片区的节点进行通信[5]。也就是说,在真实的网络拓扑中,大多数节点和链路的重要性较低,而少数节点和链路具备较高的重要性,从对抗性的角度来看,节点和链路的重要性越大,其越可能成为导致网络瘫痪的网络瓶颈。攻击者只需要在攻击前利用网络空间资源测绘技术探测目标网络的拓扑结构,把关键节点或链路作为 LFA 的攻击目标,在破坏少数关键资源的情况下就可以对整个网络的连通性造成严重影响。隐藏拓扑的网络瓶颈是目前主流的防御机制之一,但是现有的拓扑欺骗系统普遍存在一些局限性。首先,计算网络瓶颈的度量标准较为单一,他们多数只考虑了静态部分的度量指标(如路由路径数量),但是在网络正常运行过程中的动态度量(如链路使用率)同样会导致网络瓶颈的出现。其次,现有的拓扑欺骗系统多数只考虑了应对攻击者发起的拓扑测绘行为,但是当攻击者发起不依赖测绘结果的盲攻击时,他们的防御行为就变得毫无意义。因此,目前应对攻击者发起的 LFA 主要面临以下挑战:(1)如何丰富网络瓶颈的度量指标,准确定位瓶颈节点和链路;(2)如何利用拓扑混淆技术以较低的成本有效地隐藏网络瓶颈;(3)如何应对攻击者发起的盲攻击。

为解决以上问题,该文基于软件定义网络(Software Defined Network,SDN)[6]提出了 PrNet。贡献如下:(1)该系统综合考虑了网络拓扑的不同特征,从静态和动态的角度定义网络瓶颈度量指标;(2)该系统会生成针对测绘流量的混淆拓扑,通过识别测绘流量并将其引向绕开网络瓶颈的混淆路径,使攻击者得到错误的信息;(3)提出一个概率路径转发算法,该算法会为节点之间的所有可达路径分配概率,根据概率选择数据包的转发路径,通过主动使网络流量分散,降低对度量指标值较高的节点和链路的使用,减少网络瓶颈的产生;(4)在流行的 SDN 控制器 Ryu 上部署 PrNet,并在现实网络拓扑上进行了模拟和仿真,证明了 PrNet 在应对攻击者发起链路洪泛攻击时具有可行性,并且能够有效缓解攻击者发起的盲攻击。

1　相关工作

现有的 LFA 防御方法主要分为两种:被动防御和主动防御。被动防御的重点在于捕捉 LFA,针对出现拥塞的链路执行各种反应式的操作。在 Wang 等提出的 LFADefender[7] 和 Kang 等提出的 SPIFFY[8] 中,通过在系统中部署专门检测链路拥塞的模块来检测网络中是否出现 LFA。针对出现拥塞的链路,对其中的流量执行重路由操作以将流量分散至其他链路,在逻辑上临时增加链路带宽来缓解 LFA 造成的链路堵塞。Aydeger 等[9]提出了一个基于 SDN 的模型,该模型通过收集统计测绘流量来推测可疑的目标链路,当统计的数量超过阈值时,则会启动重路由功能使流量流向替代链路。但是,被动防御主要用于发生 LFA 后,其本身并不会阻止攻击者获取目标网络的信息以发现网络瓶颈。

与被动防御方法相比,LFA 的主动防御利用拓扑混淆技术在攻击者的侦察阶段进行干预,使其形成与真实拓扑相似性较低的攻击视图,从而误导其攻击非瓶颈目标,增加攻击成本。Trassare 等[10]首先通过添加虚拟链路使关键节点的瓶颈度量最小化的方式形成虚拟拓扑,然后在网络中部署智能路由器来识别并拦截测绘流量,由智能路由器对测绘流量根据虚拟拓扑生成返回的数据包。Liu 等提出的 TopoObfu[11] 首先将部分路由器替换为 SDN 交换机,通过修改 IP 数据包的 TTL 字段来实现在拓扑中添加虚拟链路,使攻击者获得虚假的拓扑测绘结果。Kim 等提出的 SDHoneyNet[12] 首先在瓶颈节点的附近节点上部署符合幂律分布的诱饵网络,当真实网络中出现 TTL 值为 1 的 traceroute 测绘流量时,将其引入到诱饵网络中。这三种方法有一个共性在于少数的部署节点需要处理大量的测绘流量,则这些少数的部署节点就成为了瓶颈本身。Liu 等提出了一个轻量级、低消耗的防御系统 NetObfu[13],其首先针对网络中的瓶颈链路均生成对应的虚拟链路,以降低瓶颈链路的流密度,然后将尽可能多的流量引向安全性较高的链路,诱导攻击者向其发起进攻。Ding 等提出的 Linkbait[14] 首先根据各链路的流量密度来推测潜在的目标链路,然后将攻击者发送的测绘流量重路由至目标链路附近的诱饵链路,以增加诱饵链路的流量密度,达到误导攻击者的效果。这些机制在应对攻击者发起的拓扑测绘行为上均能够达到一定的混淆效果,但是当攻击者发起不依赖测绘结果的盲攻击时,针对于测绘阶段的拓扑混淆则无法发挥作用了。由 Kim 等提出的 BottleNet[15] 在缓解盲攻击问题上提供了一个思路,在瓶颈链路附近部署虚拟拓扑,当瓶颈链路出现拥塞的时候,将一部分流量重路由至虚拟拓扑,让其经过足够多的虚拟节点后再转发至目标节点。但是该方法在防御过程中欠缺主动性,而且需要额外部署虚拟节点,当瓶颈链路发生改变后,需要花费较大的部署成本。

2　应对链路洪泛攻击的机制 PrNet

2.1　威胁模型概述

假设攻击者控制着一组可以在网络中注入流量的主机 bots,拓扑中的节点和链路均可成为网络瓶颈,每

个节点和链路的瓶颈值由聚合流量、路由路径和链路使用率等多种因素定义。要发起 LFA,攻击者需要执行三个步骤:(1)攻击者控制 bots 多次使用测绘工具探测节点之间的路径,并形成攻击视图;(2)将流量汇聚的节点和链路确定为网络瓶颈;(3)攻击者控制 bots 向选定的节点发送大量合法的、低速率的流量来淹没网络瓶颈。如此一来,这些拥塞的网络瓶颈会严重影响整个网络的连通性。

traceroute[16] 和 iperf[17] 是攻击者常用的拓扑测绘工具。traceroute 具有定位主机之间所有路由器的功能,其通过向目标发送 TTL 值从零递增的数据包,并收集沿途节点返回的“超时”信息来刻画数据包的转发路径。攻击者通过在不同的 bot 上执行 traceroute 并分析其结果,可以确定目标网络的拓扑结构[18]。iperf 可用于测量端到端之间的带宽,这意味着攻击者可以大致发现带宽使用率大的链路。这两个工具是合法用户常用的主机在线状态和网络故障的检测工具,因此攻击者可将其测绘行为伪装成合法用户发起的故障检测行为,以绕过防御者的检测机制。此外,该文还假设攻击者会发起不依赖拓扑测绘结果的盲攻击,通过向随机目标发送大量低速流量,以消耗沿途链路的剩余带宽,进而使得网络出现拥塞甚至瘫痪的情况。

2.2 PrNet 系统设计

PrNet 的整体架构分为路径权重计算、混淆拓扑生成算法、测绘流量识别和概率路径转发算法 4 个模块,如图 1 所示。

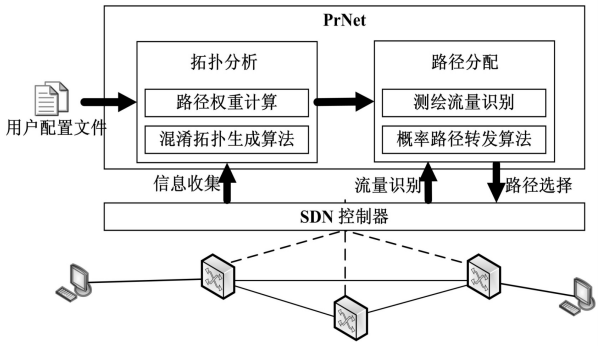

图 1 PrNet 系统架构

其中,路径权重计算和混淆拓扑生成算法主要作用是分析网络拓扑。路径权重计算模块定期收集拓扑信息,包括连接结构和流量统计信息,基于各节点和链路的瓶颈指标与配置文件信息,计算得出当前各节点之间可达路径的瓶颈权重,混淆拓扑生成算法根据计算结果生成用于欺骗攻击者的混淆拓扑。测绘流量识别和概率拓扑转发算法主要作用是拓扑运行过程中为数据包选择转发路径。测绘流量识别模块会识别网络中具有追踪功能的数据包,并为其分配对应的混淆路径。对于非测绘流量,由概率路径转发算法根据节点之间可达路径的瓶颈权重分别形成概率值,并按概率为非测绘流量选择转发路径。

2.2.1 路径权重计算

PrNet 综合考虑了网络拓扑的不同特征,从静态和动态的角度定义节点和链路的瓶颈度量指标。静态度量指标的定义主要从网络拓扑的基本结构入手,不同的特征可以从不同的角度决定节点的重要性,参考 BottleNet[15] 中提出的 3 个静态指标来识别网络瓶颈,具体定义如下:

介数中心性(Betweenness Centrality,BC)表示在两个节点的可达路径中,经过某个节点的路径数量。节点的介数中心性越大,意味着它是越多路径的中间节点,该节点的故障将会造成许多路径的连接中断。对于任意节点 u 的介数中心性,其定义为:

$$\mathrm{BC}(u)=\sum_{s\neq u\neq t\in V}\frac{\mathrm{path}_{st}(u)}{\mathrm{path}_{st}} \tag{1}$$

其中,path_{st} 表示节点 s 和 t 之间的可达路径的数量,而 $\mathrm{path}_{st}(u)$ 表示这些路径中经过节点 u 的数量。

紧密中心性(Closeness Centrality,CC)可以用来表示一个节点到所有其他可达节点的平均距离,节点的紧密中心性越大,其所在拓扑中的位置越靠近中心。对于任意节点 u 的紧密中心性,其定义为:

$$\mathrm{CC}(u)=\frac{1}{\sum_{v\in V}d(u,v)} \tag{2}$$

其中,$d(u,v)$ 表示节点 u 和 v 之间的最短距离。

度中心性(Degree Centrality,DC)可以用来表示一个节点与其他节点的相关性程度。若一个节点具有较高的度,则意味着该节点与邻居节点存在着较多连接,那么该节点的故障将会对网络连通性造成很大的影响。对于任意节点 u 的度中心性,其定义为:

$$\mathrm{DC}(u)=\frac{\deg(u)}{\sum_{v\in V}\deg(v)} \tag{3}$$

其中,$\deg(u)$ 表示节点 u 的度。此外,根据网络拓扑运行过程中节点和链路的流量变化情况定义动态度量指标,通过 OpenFlow 协议[19] 收集以下 2 个动态指标来识别网络瓶颈,具体定义如下:

聚合流量比例(Aggregate Traffic Ratio,ATR)用于表示一个节点在一定时间内接收到的字节总数的比例,能够反映出拓扑中各节点的使用率。使用 REST API[20] 来实现与底层 OpenFlow 交换机的通信,定期获取各个节点的聚合统计信息,对于任意节点 u 的聚合流量比例,其计算公式为:

$$F(u)=\frac{f_u(t_2)-f_u(t_1)}{t_2-t_1} \tag{4}$$

$$\mathrm{ATR}(u)=\frac{F(u)}{\sum_{C}^{v}F(v)} \tag{5}$$

其中,$f_u(t)$ 表示节点 u 在 t 时间戳接收的字节总数,$F(u)$ 表示节点 u 在 t_2-t_1 时间间隔接收字节总数的变化率。

链路使用率(Consumed Link Ratio,CLR)表示一条链路的使用带宽占链路总带宽的比例,能够反映出该链路的使用情况。运用 OpenFlow 协议的 Port_Statistics 消息收集 SDN 交换机中所有端口的流量统计信息,对于任意链路 $e(u,v)$ 的使用率,其计算公式为:

$$\mathrm{CLR}(u,v)=\frac{\dfrac{8(\mathrm{TX}_{u,v}(t_1)-\mathrm{TX}_{u,v}(t_0))}{t_1-t_0}}{\mathrm{bw}(u,v)} \tag{6}$$

其中,$\mathrm{TX}_{u,v}(t)$ 表示在 t 时间内 $e(u,v)$ 传输的字节数量,$\mathrm{bw}(u,v)$ 表示 $e(u,v)$ 的总带宽。

PrNet 会在部署阶段使用深度优先算法计算两两节点之间所有的可达路径 $\mathrm{Path}_{u,v}$,然后在网络拓扑工作过程中定期收集各个节点和链路的度量指标信息来计算每一条可达路径的瓶颈权重。不同指标对路径权重计算的影响程度由影响因子 $M=(m_{\mathrm{BC}},m_{\mathrm{CC}},m_{\mathrm{DC}},m_{\mathrm{ATR}},m_{\mathrm{CLR}})^{\mathrm{T}}$ 决定。例如当 M 取值为[1,1,1,0,0]时,表示该路径的权重只由静态指标 BC、CC 和 DC 决定。对于每个节点的度量指标信息使用矩阵 $\boldsymbol{D}$ 来表示:

$$\boldsymbol{D}=\begin{pmatrix} d_{11} & \cdots & d_{1r} \\ \cdots & d_{ij} & \cdots \\ d_{n1} & \cdots & d_{nr} \end{pmatrix} \tag{7}$$

其中,r 表示节点度量指标的类型,n 表示节点的数量,d_{ij} 表示第 i 个节点的第 j 类指标的度量值。

对于每条链路的使用率,使用矩阵 $\boldsymbol{C}$ 来表示:

$$\boldsymbol{C}=(c_1\cdots c_e\cdots c_n)^{\mathrm{T}} \tag{8}$$

其中,n 表示链路的数量,c_e 表示链路 e 的使用率。

因此,对于每条可达路径 path 的瓶颈权重 path_w 可表示为:

$$\mathrm{path\_w}=\sum_{i\in\mathrm{Path}}\sum_{j\in r}m_j\times d_{ij}+\sum_{e\in\mathrm{Path}}m_{\mathrm{CLR}}\times c_e \tag{9}$$

其中,m_j 表示第 j 类指标的影响因子。

2.2.2　混淆拓扑生成算法

拓扑混淆使攻击者形成与实际流量拓扑不一致的视图,从而达到隐藏网络瓶颈的目的。该实际流量拓扑意为当前时间数据包的转发路径形成的拓扑视图,攻击者只有掌握流量的走向和分布,才能实施有效的攻击,因此隐藏数据包的实际转发路径是拓扑混淆的关键所在。攻击者可能会在不同的时间段发起多次测绘行为,测绘结果的不一致会引起攻击者的怀疑,进而发起盲攻击,因此 PrNet 根据静态度量指标来生成混淆拓扑,并且混淆拓扑均由真实的网络节点和链路形成,避免额外的部署开销。针对攻击者的测绘流量,PrNet 会分配一条包含最少瓶颈节点的最远的转发路径,因此测绘流量会经过更多的非瓶颈节点,使得非瓶颈节点在攻击者的攻击视图中呈现更高的瓶颈度量。用 B_u 表示每个节点的瓶颈权重,I_b 表示一条可达路径中包含的瓶颈节点,W_b 表示一条可达路径中瓶颈节点的权重和。

$$W_b=\sum_{i\in I_b}B_u(i) \tag{10}$$

混淆拓扑生成算法 Obfu-Topology 的工作流程如算法 1 所示,输入为网络拓扑 G、节点度量指标 D、瓶颈节点个数 K、影响因子 $M=[1,1,1,0,0]$,输出为混淆拓扑 G'。混淆拓扑生成算法首先取 B_u 中最大的 K 个节点形成瓶颈节点集合 V_b,优先选择不包含瓶颈节点的路径作为混淆路径,若一对节点中所有的可达路径都需要经过瓶颈节点,则选择 W_b 最小的路径作为混淆路径。并且 PrNet 会对瓶颈节点的选择进行负载均衡处理,适当调整被选中的瓶颈节点的权重值 B_u。

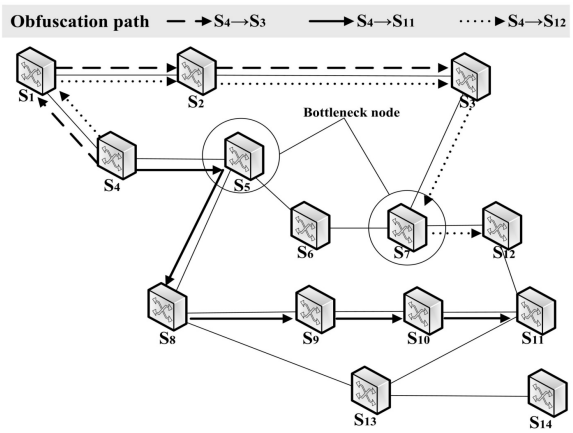

图 2　拓扑 Eunetworks

图 2 为网络拓扑 Eunetworks,假设 K 取值为网络拓扑节点总数的 15%,根据计算可得出(S_5, S_7)为瓶颈节点。当 S_4 中的主机向 S_3 发起 traceroute 测绘时,会分配不经过瓶颈节点的路径 path=(S_4, S_1, S_2, S_3)作为混淆路径;对于 S_4 到 S_{11} 的混淆路径,会分配为经过最少的瓶颈节点且最远的路径,即 path=(S_4, S_5, S_8, S_9, S_{10}, S_{11});而对于 S_4 到 S_{12} 的混淆路径,系统会进行负载均衡处理,将其分配为 path=(S_4, S_1, S_2, S_3, S_7, S_{12})。

Algorithm 1:Obfu-Topology

Input: Physical topology G; Node Metrics D; Number of bottleneck nodes K; Impact Factor $M=[1,1,1,0,0]$

Output:Obfuscated Topology G'

1. $B_u \leftarrow D * M$
2. $V_b \leftarrow$ The K largest value in B_u
3. for $u,v \in G$
4. for path in $\mathrm{Path}_{u,v}$
5. $I_b \leftarrow \mathrm{path} \cap V_b$

6. if $I_b = \varnothing$
7. $O_{u,v} \leftarrow$path
8. else
9. $W_b \leftarrow \sum_{i \in I_b} B_u(i)$
10. $O_{u,v} \leftarrow$The path with the smallest W_b
11. $B_u(I_b) \leftarrow B_u(I_b) + 1$
12. end if
13. end for
14. $G' \leftarrow$ The largest path_w in $O_{u,v}$
15. end for
16. return G'

2.2.3 测绘流量识别

测绘流量识别模块主要用于识别网络中的测绘数据包,当新的数据包进入 OpenFlow 交换机时,因为缺少针对该数据包的转发流表,所以交换机会触发 table_miss 向 SDN 控制器请求下发相应的流表,此时测绘流量识别模块会对该数据包的内容进行特征分析,若为测绘流量,则分配混淆路径。攻击者能够使用 traceroute 工具进行拓扑发现,其通过向目标主机发送目的端口为 33 434-33 534 并且 TTL=1 的 UDP 数据包,再根据沿途路由器返回的“超时”信息刻画数据包的转发路径。一个简单的解决方案是部署蜜罐检测所有 TTL=1 的 UDP 数据包,但是如果攻击者精心设计数据包使其在蜜罐节点上的 TTL 值大于 1,或者使用其他类型的数据包,则可绕过蜜罐的检测。为了使检测规则更加完善,PrNet 让所有 OpenFlow 交换机都成为检测节点,捕捉网络中目的端口大于 33 434 并且 TTL 值小于 6 的所有类型的数据包,再由 SDN 控制器为相应的节点下发高优先级的特定流表用于转发该测绘数据包。

2.2.4 概率路径转发算法

经过测绘流量识别模块的过滤后,需要为合法的流量分配转发路径,该部分工作由概率路径转发算法完成。在传统的网络架构中,路由转发路径是固定的,当网络中出现链路拥塞或者攻击者的恶意盲攻击时,路由器没法灵活调整流量的走向,从而影响了数据的传输效率。为了提高网络的灵活性和可用性,PrNet 会在源节点和目标节点之间选择 L 条最优的可达路径作为备选路径 $A_{s,d}$,定期变更数据包的转发路径,使网络中的流量尽可能实现负载均衡,减少网络瓶颈的产生。

可达路径的瓶颈权重 path_w 越大,则代表经过的节点和链路越多,或者经过的节点和链路正处于高负荷状态,选择 path_w 较大的路径将使得数据包的转发延迟增大。考虑到转发效率的问题,概率路径转发算法需要先对数据包的源地址 s 和目的地址 d 之间所有的可达路径 $\text{Path}_{s,d}$ 根据用户自定义的备选路径数量 L 进行筛选,排除掉 path_w 较大的路径,再对剩下的路径进行概率值分配。备选路径的概率值 $\text{Prob}_{\text{path}}$ 计算公式为:

$$\text{Prob}_{\text{path}} = \frac{\frac{1}{\text{path\_w}_{\text{path}}}}{\sum_{i \in A_{s,d}} \frac{1}{\text{path\_w}_i}} \tag{11}$$

最后从备选路径中选择一条路径作为数据包在 T 时间的转发路径,确保转发路径的分配会根据网络拓扑流量的实际情况进行调整。概率路径转发算法 Prob-Path 的工作流程如算法 2 所示,输入为数据包的源地址 s、目的地址 d 和备选路径的数量 L,输出为数据包的转发路径 $\text{path}_{\text{forward}}$。

例如,在网络拓扑 Eunetworks 中,与 S_7 直连的 H_7 向与 S_8 直连的 H_8 发送 UDP 数据包,节点 S_7 到 S_8 一共有 4 条可达路径,分别是 path=(S_7,S_6,S_5,S_8)、path=(S_7,S_{12},S_{11},S_{13},S_8)、path=(S_7,S_{12},S_{11},S_{10},S_9,S_8)、path=(S_7,S_3,S_2,S_1,S_4,S_5,S_8),其 path_w 分别为 1.3、1.6、3.2、4.5,假设此时备选路径数量为 3,则 path=(S_7,S_3,S_2,S_1,S_4,S_5,S_8)在该时间段会被剔除,而剩下的 3 条路径作为备选路径,其被选中的概率分别为 0.45、0.37、0.18,根据概率值组成三个区间(1,45)(45,82)(82,100),假设由随机算法产生的随机值为 56,位于第二个区间,则将 path=(S_7,S_{12},S_{11},S_{13},S_8)分配为数据包(H_7,H_8,UDP)的转发路径。SDN 控制器会向路径沿途的交换机 S_7、S_{12}、S_{11}、S_{13} 和 S_8 下发相应的流表,在 T 时间内该类型的数据包可在网络拓扑中直接转发而无需经过控制器,在 T 时间后,先前分配的流表超时,控制器需要重新计算出新的转发路径,并将流表下发至对应的交换机中。PrNet 以数据包的源地址、目的地址和协议区分不同的数据包,若 H_7 向 H_8 同时发送 UDP 和 TCP 数据包,控制器会为两种数据包类型进行独立计算,并下发不同的流表。

Algorithm2:Prob-Path

Input: Source Address s; Destination Address d; Number of alternative paths L

Output: Forwarding path $\text{path}_{\text{forward}}$

1. $A_{s,d} \leftarrow$The L smallest values of path_w in $\text{Path}_{s,d}$
2. for path in $A_{s,d}$
3. $\text{Prob}_{\text{path}} \leftarrow \frac{\frac{1}{\text{path\_w}_{\text{path}}}}{\sum_{i \in A_{s,d}} \frac{1}{\text{path\_w}_i}}$
4. Set[i]$\leftarrow$Set[$i-1$]+$\text{Prob}_{\text{path}}$
5. end for
6. $r \leftarrow$random(1,100)
7. for i in Set
8. if Set[$i-1$]$< r <=$Set[i]

9. $path_{forward}$←The i-th path in $A_{s,d}$
10. end if
11. end for
12. return $path_{forward}$

3　实验评估

该文使用基于 OpenFlow1.3 协议的 Ryu 作为 SDN 控制器,使用 Mininet 对数据集 Topology Zoo[21] 中 3 种不同规模的网络拓扑进行模拟仿真,其网络节点与链路的数量如表 1 所示。主要从以下几个方面来评估 PrNet:(1)验证混淆拓扑的安全性;(2)验证概率路径转发算法的有效性;(3)PrNet 性能基准测试。对下面给出的每项评估均重复实验 30 次,并展示出最终的平均值。

表 1　网络拓扑信息

Types of topologies	Eunetworks	Agis	Iris
The number of nodes	14	25	51
The number of links	16	30	64

3.1　验证混淆拓扑的安全性

混淆拓扑的安全性可以通过比较混淆拓扑与实际流量拓扑的相似性来衡量。该文引入编辑距离(Levenshtein distance)表示混淆路径与实际流量路径之间的差异,路径的差异性可以表示为编辑距离和实际流量路径的长度比,实际流量路径为备选路径的概率的加权和,拓扑整体的相似性表示为:

$$simi = 1 - \frac{1}{|F|}\sum_{f\in F}\frac{\sum_{n}^{i}\text{Levenshtein\_distance}(l_{f\_i}, l_f') \times \text{Prob}_{l_{f\_i}}}{\sum_{n}^{i}(l_{f\_i} \times \text{Prob}_{l_{f\_i}})} \tag{12}$$

其中,F 表示拓扑中所有节点连接关系的集合,即 $F = [(S_1,S_2),(S_1,S_3),\cdots]$,$l_{f\_i}$ 表示其中一条备选路径,l_f' 表示混淆路径。在 Eunetworks 中,设定瓶颈节点数量为总节点的 20%,混淆拓扑与实际流量拓扑的相似性随备选路径数量的增大而减小,如图 3 所示。当备选路径数量为 2 时,在拓扑 Eunetworks 中相似性为 62%,在 Agis 中为 73%,在 Iris 中为 59%,当备选路径数量为 3 时,在拓扑 Eunetworks 中的相似性下降至 55%,在 Agis 中下降至 64%,在 Iris 中下降至 55%。随着备选路径数量的增大,数据包的转发路径选择增多,混淆路径与备选路径之间的编辑距离也会随之增大,导致混淆拓扑与实际流量拓扑的相似性减小,因此 PrNet 能够生成与实际流量拓扑明显不一致的混淆拓扑,具有良好的安全性。

3.2　验证概率路径转发算法的有效性

在网络拓扑运行过程中验证概率路径转发算法的有效性,首先测试备选路径的概率值是否根据网络实时流量的变化而变化,其次测试概率路径转发算法对盲攻击的缓解效果。

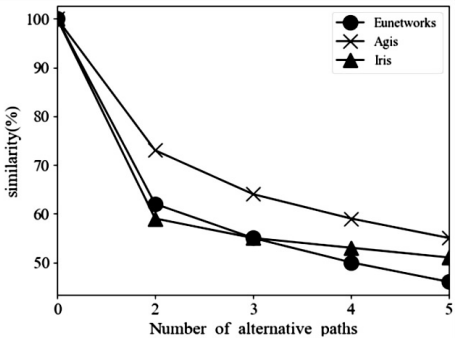

图 3　相似性

3.2.1　备选路径的概率变化

在 Eunetworks 中,设定瓶颈节点数量为总节点的 20%,备选路径数量为 4,为了能够凸显拓扑运行过程中链路使用率 CLR 对备选路径概率值的影响,影响因子 M 取值为[1,1,1,1,3]。H_7 和 H_8 是分别连接在节点 S_7 和 S_8 上的两台主机,$path_1 = (S_7,S_6,S_5,S_8)$、$path_2 = (S_7,S_{12},S_{11},S_{13},S_8)$、$path_3 = (S_7,S_{12},S_{11},S_{10},S_9,S_8)$、$path_4 = (S_7,S_3,S_2,S_1,S_4,S_5,S_8)$ 为数据包(H_7, H_8, UDP)的所有可达路径,由于备选路径数量为 4,故该 4 条可达路径均为备选路径。

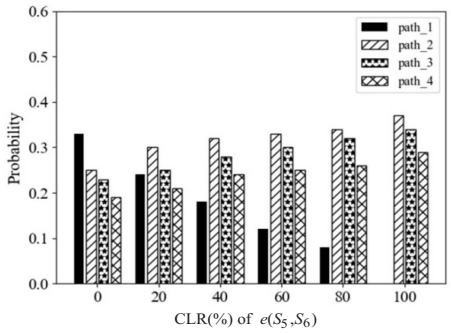

图 4　备选路径的概率变化

图 4 中给出了数据包(H_7, H_8, UDP)的备选路径概率值变化情况。随着 $e(S_5,S_6)$ 的 CLR 上升,包含有 $e(S_5,S_6)$ 的备选路径 $path_1$ 的瓶颈权重会上升,使得分配给该备选路径的概率值会随之下降,当 $e(S_5,S_6)$ 的 CLR 为 0.2 时,$path_1$ 的概率值为 0.24,而当 $e(S_5,S_6)$ 的 CLR 为 1.0 时,代表 $e(S_5,S_6)$ 处于拥塞状态,此时 SDN 控制器将不再把流量引向拥塞的链路,因此 $path_1$ 的概率值会调整为 0。实验结果与预想相符合,说明 PrNet 在拓扑运行过程中能够定期收集各个节点和链路的度量指标信息用于更新可达路径的瓶颈权重,并调整可达路径分配得到的概率值,使得网络具备更好的处理能力和灵活性。

3.2.2　缓解盲攻击

在 Eunetworks 中验证 PrNet 如何缓解网络中出现

的盲攻击。假设攻击者在节点 S_1、S_6、S_9 和 S_{13} 上部署有 bots,其中 S_6、S_9、S_{13} 中的 bots 发送大量低速率流量至 S_1 的诱饵服务器上,此时 $e(S_4,S_5)$ 成为了最有可能被攻击者洪泛的链路,如图 5 所示。将所有链路的带宽设置为 10 Gbps,每个节点之间产生的背景流量消耗每条链路大约 40% 的链路使用率。考虑到 LFA 具有低速率流量的特性,规定每个 bot 最多发送 8 Mbps 的攻击流量。

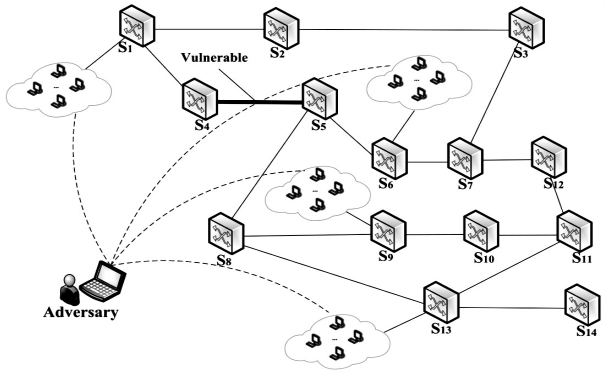

图 5 用于测试盲攻击的拓扑 Eunetworks

图 6 给出了分别运行最短路径策略(Shortest Path First,SPF)和 PrNet 时链路使用率随攻击成本的变化趋势,横坐标为攻击者洪泛 $e(S_4,S_5)$ 所需要的攻击成本,攻击成本以每个节点需要部署 bots 的数量来呈现,纵坐标为 $e(S_4,S_5)$ 的链路使用率 CLR。根据实验结果,当 bots 的数量为 250 时,在 SPF 方案中 $e(S_4,S_5)$ 的使用率达到了 100%,而在 PrNet 中 $e(S_4,S_5)$ 的使用率仅为 65%,与 SPF 方案相比,PrNet 使攻击者需要多花费接近一倍的代价才能达到洪泛链路的目的。实验结果与预想相符合,说明 PrNet 在拓扑运行过程中会根据流量的走向和分布调整数据包的转发路径,使网络中的流量尽可能实现负载均衡,减少链路拥塞情况的出现,有效缓解了攻击者发起的盲攻击。

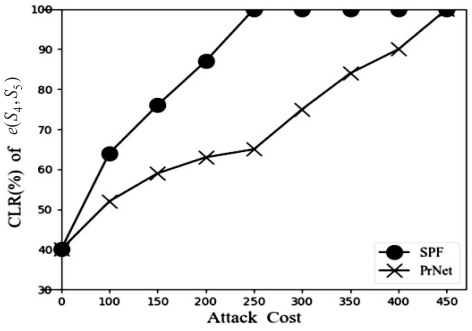

图 6 缓解盲攻击

3.3 PrNet 性能基准测试

通过测试系统在拓扑结构更新时的灵敏性和备选路径数量对数据包转发时延的影响来衡量 PrNet 的性能。

3.3.1 系统的灵敏性

当网络拓扑结构发生变化时,PrNet 需要根据实际拓扑结构调整各节点的静态度量指标值,以及重新生成混淆拓扑。在 3 种不同规模的网络拓扑中测量这两个步骤的处理时间,如表 2 所示。结果表明,对于规模较小的网络拓扑,PrNet 能够在 0.5 秒以内完成节点度量指标值和混淆拓扑的计算,而对于大型网络拓扑如 Iris,计算时间虽然对比前两者有明显增大,但是总时间仍控制在 1 分钟以内。因此,笔者认为 PrNet 具有良好的灵敏性,能够有效应对真实环境中拓扑结构更新的情况。

表 2 处理时间 ms

Types of calculation	Eunetworks	Agis	Iris
Static index calculation	20	408	48 318
Obfuscated topology calculation	13	97	10 257

3.3.2 数据包的转发时延

在此评估中,测量备选路径数量增多,对数据包转发时延的影响。在 3 种不同规模的网络拓扑中,随机挑选两个位于不同节点的主机进行通信,主机之间连续 30 秒每秒发送 10 个数据包,每个数据包大小为 5 000字节。当 PrNet 中备选路径数量为 1 时,代表此时系统使用最短路径策略分配转发路径。

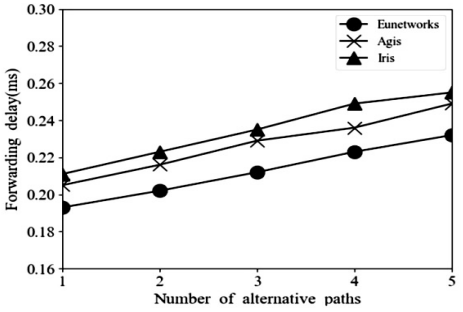

图 7 转发时延

如图 7 所示,当备选路径数量为 1 时,单个数据包在 3 种拓扑中的平均转发时延分别为 0.193 ms,0.205 ms,0.211 ms,当备选路径数量为 2 时,转发时延分别为 0.202 ms,0.216 ms,0.223 ms。在 PrNet 中,数据包的转发路径会是备选路径中的任意一条路径,而随着备选路径数量的增大,会有更长的路径纳入备选路径中,当数据包的转发路径为其中的一条较长的路径时,其转发延迟也会相应的增大,因此数据包的转发延迟会随着备选路径数量的增大而增大。但是在 3 种网络拓扑中,转发时延的平均涨幅分别为 0.010 5 ms,0.011 ms和0.012 ms,变化幅度较小,对数据包的正常转发造成的影响不大。

4 结束语

该文提出了一种应对链路洪泛攻击的机制 PrNet,

PrNet 的主要思想是从多角度定义形成网络瓶颈的度量指标,通过识别攻击者的测绘流量并为其分配一条安全的混淆路径,使得网络瓶颈在攻击者形成的攻击视图中具有较低的瓶颈度量值,并且通过概率路径转发算法,使网络拓扑中的流量尽可能实现负载均衡,减少网络瓶颈的产生。实验表明,PrNet 生成的混淆拓扑能够抵御攻击者在攻击前期实施的网络侦察行为,概率路径转发算法能够让网络具备更好的处理能力和灵活性,并且有效缓解攻击者不依赖测绘结果而发起的盲攻击。PrNet 仅考虑了将所有传统路由器变更为 SDN 交换机的实现方案,对于大型网络来说,更换所有路由器需要付出较大的代价,下一步将继续探索仅将网络中的部分路由器更换为 SDN 交换机的方案;对于大型网络拓扑,加入多个控制器,提升系统的数据处理能力;引入机器学习的算法,由算法根据当前拓扑的流量情况,决定备选路径的数量,为数据包分配最优的转发路径。

参考文献:

[1] YUAN X,LI C,LI X. DeepDefense:identifying DDoS attack via deep learning[C]//2017 IEEE international conference on smart computing (SMARTCOMP). Hong Kong:IEEE,2017:1-8.

[2] THILAGAM T,ARTHI K,AMUTHADEVI C. A survey on security and privacy issues in cloud computing[J]. International Journal of Engineering & Technology,2018,7(2):88-92.

[3] KANG M S,LEE S B,GLIGOR V D. The crossfire attack [C]//2013 IEEE symposium on security and privacy. Berkeley:IEEE,2013:127-141.

[4] KATKAR V,BHIRUD S G. Novel DoS/DDoS attack detection and signature generation[J]. International Journal of Computer Applications,2012,47(10):18-24.

[5] DHAMDHERE A,DOVROLIS C. The internet is flat:modeling the transition from a transit hierarchy to a peering mesh [C]//Proceedings of the 6th international conference(Co-NEXT '10). New York:Association for Computing Machinery,2010:1-12.

[6] SHIN S,XU L,HONG S,et al. Enhancing network security through software defined networking (SDN)[C]//2016 25th international conference on computer communication and networks (ICCCN). Waikoloa:IEEE,2016:1-9.

[7] WANG J,WEN R,LI J,et al. Detecting and mitigating target link-flooding attacks using SDN[J]. IEEE Transactions on Dependable and Secure Computing,2018,16(6):944-956.

[8] KANG M S,GLIGOR V D,SEKAR V. SPIFFY:inducing cost-detectability tradeoffs for persistent link-flooding attacks[C]//Network and distributed system security symposium (NDSS). San Diego:NDSS,2016:53-55.

[9] AYDEGER A,SAPUTRO N,AKKAYA K,et al. Mitigating crossfire attacks using SDN-based moving target defense [C]//2016 IEEE 41st conference on local computer networks (LCN). Dubai:IEEE,2016:627-630.

[10] TRASSARES T,BEVERLY R,ALDERSON D. A technique for network topology deception[C]//MILCOM 2013-2013 IEEE military communications conference. San Diego:IEEE,2013:1795-1800.

[11] 刘亚群,邢长友,高雅卓,等. TopoObfu:一种对抗网络侦察的网络拓扑混淆机制[J]. 计算机科学,2021,48(10):278-285.

[12] KIM J,SHIN S. Software-defined HoneyNet:towards mitigating link flooding attacks[C]//2017 47th annual IEEE/IFIP international conference on dependable systems and networks workshops (DSN-W). Denver:IEEE,2017:99-100.

[13] LIU Y,ZHAO J,ZHANG G,et al. NetObfu:a lightweight and efficient network topology obfuscation defense scheme [J]. Computers & Security,2021,110:102447.

[14] DING X,XIAO F,ZHOU M,et al. Active link obfuscation to thwart link-flooding attacks for internet of things[C]//2020 IEEE 19th international conference on trust,security and privacy in computing and communications (TrustCom). Guangzhou:IEEE,2020:217-224.

[15] KIM J,NAM J,LEE S,et al. BottleNet:hiding network bottlenecks using SDN-based topology deception[J]. IEEE Transactions on Information Forensics and Security,2021,16:3138-3153.

[16] DAN O,PARIKH V,DAVISON B D. IP geolocation using traceroute location propagation and IP range location interpolation[C]//Companion proceedings of the web conference 2021. New York:Association for Computing Machinery,2021:332-338.

[17] AGUSRIANDI A,ELIHAMI E. Developing delay jitter,throughput,and package lost:IPERF3 for learning islamic education[J]. Jutkel:Jurnal Telekomunikasi,Kendali dan Listrik,2020,2(1):23-30.

[18] HUANG Y,RABINOVICH M,AL-DALKY R. FlashRoute:efficient traceroute on a massive scale[C]//Proceedings of the ACM internet measurement conference. New York:Association for Computing Machinery,2020:443-455.

[19] CHEKOORY Y K,MUNGUR A U. Use of Openflow to manage network devices[C]//International conference on electrical and electronics engineering. Singapore:Springer,2022:376-386.

[20] ZHOU W,LI L,LUO M,et al. REST API design patterns for SDN northbound API[C]//2014 28th international conference on advanced information networking and applications workshops. Victoria:IEEE,2014:358-365.

[21] KNIGHT S,NGUYEN H X,FALKNER N,et al. The internet topology zoo[J]. IEEE Journal on Selected Areas in Communications,2011,29(9):1765-1775.