网络攻击源追踪技术研究综述

姜建国[1,2], 王继志[1,2,3], 孔 斌[4], 胡 波[1,2], 刘吉强[4]

[1] 中国科学院信息工程研究所, 北京 中国 100093
[2] 中国科学院大学网络空间安全学院, 北京 中国 100093
[3] 山东省计算中心(国家超级计算济南中心), 济南 中国 250101
[4] 北京交通大学, 北京 中国 100093

摘要 在网络空间中, 网络攻击源追踪是指当检测到网络攻击事件发生后, 能够追踪定位真正的攻击者的主机, 以帮助司法人员对攻击者采取法律手段。近二十年, 研究人员对网络攻击源追踪技术进行了大量研究。本文对这些研究进行了综述。首先, 明确了问题的定义, 不同的攻击场景所采用的追踪技术也有所不同, 将问题分为 5 类: 虚假 IP 追踪、Botnet 追踪、匿名网络追踪、跳板追踪、局域网追踪, 分别总结了相关的研究成果, 明确各种方法所适用的场景。最后, 将各类方法归纳为 4 种类型, 对比了这 4 类方法的优缺点, 并讨论了未来的研究方向。

关键词 网络安全, IP 追踪, 跳板检测, 僵尸网络, 匿名网络
中图法分类号 TP393.0 DOI 号 10.19363/j.cnki.cn10-1380/tn.2018.01.008

On the Survey of Network Attack Source Traceback

JIANG Jianguo[1,2], WANG Jizhi [1,2,3], KONG Bin[4], HU Bo[1,2], LIU Jiqiang[4]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
[2]School of Cyber Security, University of Chinese Academy of Sciences Beijing 100093, China
[3] Shandong Computer Science Center (National Supercomputer Center in Jinan), Jinan 250101, China
[4]Beijing Jiaotong University, Beijing 100093, China

Abstract In cyberspace, the network attack sources traceback is to trace and locate the real hosts owned by attackers after network attack events are detected, so that judicial personals can take some legal action to arrest or prosecute these network attackers. In recently twenty years, there are lots of researches on the issue, which is investigated by the paper. Firstly, the definition of the issue is argued. Since different traceback technologies should be applied on different attack scenarios, the issue can be divided into 5 sub-problems: Spoofing IP Traceback, Botnet Traceback, Anonymous Network Traceback, Step Stone Traceback, and Local Network Traceback. In the 5 sub-problems, all kinds of methods are surveyed and their primitive conditions are discussed. Finally, these methods are sumed up 4 types, whose strengths and weaknesses are compared. Then the future research work is proposed.

Key words Network security, IP traceback, step-stone detection, Botnet, anonymous network

1 引言

近年来, 网络安全事件层出不穷, 各种网络攻击给国家、社会和个人带来了严重的危害, 如分布式拒绝服务攻击(DDoS)、基于僵尸网络(Botnet)的高级可持续攻击(APT)、利用远程控制木马的信息窃取等。在这些攻击方法中, 攻击者会向目标主机, 也就是受害主机, 发送特定的攻击数据包。从受害主机的角度来看, 能够观察到这些攻击数据包, 如果能追踪这些攻击数据包的来源, 定位攻击者的真正位置, 受害主机不但可以采用应对措施, 如在合适的位置过滤攻击数据包, 而且可以对攻击者采取法律手段。因此在网络取证领域, 网络攻击溯源一直是一个热点问题。

现有的 TCP/IP 协议在设计之初, 并没有考虑攻击源追踪问题, 不能提供对数据包来源的认证功能, 这使得在现有互联网基础设施条件下, 对攻击源追踪是很困难的事情。因为一般在 IP 数据包中, 只有源 IP 地址这一个字段能够表征该数据包的发送者。对于攻击者来说, 为了隐藏自身, 可以有意识的采

通讯作者: 王继志, 副研究员, Email: wangjizhi@iie.ac.cn。
本课题得到山东省重大科技创新工程(编号: 2017CXGC0704)资助。
收稿日期: 2016-06-14; 修改日期: 2016-08-19; 定稿日期: 2017-12-05

取措施避免数据包中的源 IP 地址暴露自己的真实 IP 地址。主要的方法有: 如果攻击者不需要接收受害主机的响应数据包, 如 DoS 攻击, 则可以采用虚假 IP 地址来填充源 IP 地址字段; 如果攻击者需要与受害主机建立连接, 则可以采用先攻击控制其他主机作为跳板, 而后从跳板主机发动攻击, 这样源 IP 地址字段中出现的是跳板主机的 IP 地址, 而非攻击者的 IP 地址。对于取证人员来说, 即使检测到攻击数据包, 也很难凭借这些攻击数据包去追踪定位真正的攻击者。

因此, 目前还不存在一种通用的攻击源追踪方法。已有的追踪方法都有严格的限定条件, 只能适用于一些特定的场景。目前, 对于攻击源追踪方法的分类主要依据所基于的理论, 但事实上, 这些方法的适用范围不同、前提条件不同、需要取证人员事先掌握的资源不同, 因此, 本文按照取证人员事先需要掌握的资源为依据, 从实际应用的角度, 将各种攻击溯源方法进行分类, 说明各类方法的适用范围。

全文组织如下: 第 2 节明确了问题的定义, 根据不同的场景将问题分为 5 类; 第 3 节分别对这 5 类问题现有的研究成果进行了综述; 第 4 节将已有方法大致归纳为 4 种类型, 对这 4 类的优缺点进行了对比; 第 5 节总结了全文并展望未来的研究方向。

2 问题定义

本文所说的网络攻击源追踪是指, 在网络空间中, 安全人员在检测到攻击行为发生的情况下, 如何追踪定位攻击者的主机? 虽然从司法取证的角度, 更希望能识别出攻击者本人, 以便于对嫌疑人采取法律手段, 但这需要将网络空间中的“主机”与现实物理空间中的“人”进行关联, 这超出了本文的范围。本文的内容限定于在网络空间中, 如何识别出攻击者直接使用的主机。

由于攻击者可以采用不同的形式来隐藏自身, 如虚假 IP 地址或跳板的方式, 那么可以将攻击源追踪问题分为下面 5 个子问题: 虚假 IP 溯源、僵尸网络溯源、匿名网络溯源、跳板溯源、局域网溯源。下面明确一下这 5 个子问题的具体含义。

1) 虚假 IP 溯源: 取证人员检测到的攻击数据包中, 其源 IP 地址是伪造的, 在这种情况下如何追踪定位攻击者的主机?

例如, 典型的 SYN Flood 攻击, 见图 1。

在这种攻击中, 攻击者将攻击数据包中的源 IP 地址替换为伪造的 IP 地址, 受害主机收到数据包后, 将响应数据包发送给伪造的 IP 地址主机, 这些主机可能存在也可能不存在。这样在受害主机端, 无法得到攻击主机的 IP 地址。

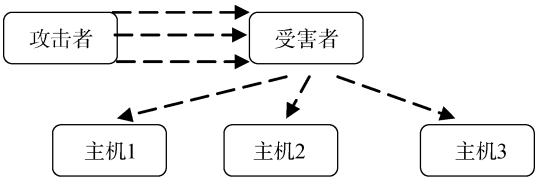

图 1 SYN Flood 攻击示意

Figure 1 The sketch of SYN flood attack

除此之外, 还有一种特殊的“反射攻击”, 如 Smurf 攻击、DNS 放大攻击等, 见图 2。

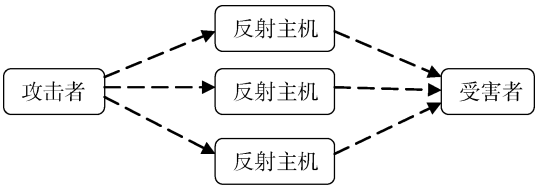

图 2 反射攻击示意

Figure 2 The sketch of reflection attack

在这种攻击中, 攻击者将攻击数据包中的源 IP 地址替换为受害者的 IP 地址, 将攻击数据包发送给反射主机, 反射主机收到数据包后, 响应数据包将发送给受害主机。从受害主机端观察, 只能判断这些数据包来自反射主机, 无法知道真正攻击者的 IP 地址。

2) 僵尸网络溯源: 攻击者利用僵尸网络发动攻击, 取证人员检测到攻击数据包中, 其源 IP 地址来自于 Botnet 中的 bot 主机, 在这种情况下如何追踪定位攻击者的主机?

典型的攻击场景如图 3。

在这种攻击中, 攻击者利用 C&C 型或 P2P 型 Botnet, 先发送控制指令, bot 主机接收到控制指令后, 向设定的攻击目标发动攻击。在受害主机端, 可以看到攻击数据包来自 bot 主机, 而无法识别出真正的 Botmaster。

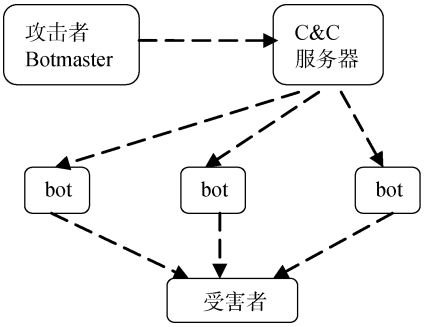

图 3 C&C 型 Botnet 攻击示意

Figure 3 The attack sketch of C&C type of Botnet

3) 匿名网络溯源: 攻击者利用匿名网络, 如 Tor, 发动攻击, 取证人员检测到攻击数据包中, 其源 IP 地址来自于匿名网络, 在这种情况下如何追踪定位攻击者的主机?

以 Tor 网络为例, 典型的攻击场景如图 4 所示。

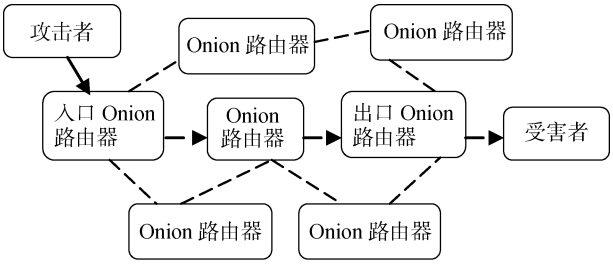

图 4 利用匿名网络的攻击示意

Figure 4 The attack sketch based on anonymous network

在这种攻击中, 攻击者的攻击数据包通过匿名网络进行转发, 在受害主机端, 只能观察到攻击数据包来自于出口路由器, 而不能发现真正的攻击者。

4) 跳板溯源: 攻击者利用多个“跳板主机”, 即通过控制多个主机转发攻击数据包, 取证人员检测到攻击数据包, 其源 IP 地址是最后一跳“跳板主机”的 IP 地址, 在这种情况下如何追踪定位攻击者的主机?

典型的攻击场景如图 5 所示。

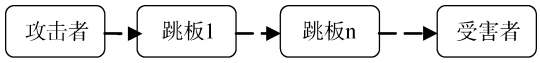

图 5 利用多个跳板的攻击示意

Figure 5 The attack sketch based on multi-step stones

在这种攻击中, 攻击者事先控制多个跳板主机, 利用跳板转发攻击数据包。在受害主机端, 只能看到攻击数据包来自于最后一跳的主机, 而不能识别出真正的攻击者。很显然, 跳板路径越长, 越难追踪攻击者。

5) 局域网络溯源: 攻击者位于私有网络内, 其攻击数据包中的源 IP 地址经过了网关的 NAT(Network Address Transform)地址转换, 在这种情况下如何在内网中追踪定位攻击者主机?

典型的攻击场景如图 6 所示。

图 6 隐藏在 NAT 服务器后的攻击者

Figure 6 The attack sketch hide behind NAT server

在这种攻击中, 由于攻击者的 IP 地址是私有 IP 地址, 在受害主机端只能看到 NAT 网关的 IP 地址。在大型私有网络内, 特别是无线局域网中, 寻找定位攻击者并不是一件简单的事情。

在实际的网络攻击事件中, 可能并不严格遵守上述各种攻击场景, 但大致可以归为上述某个或某几个问题。因此, 以上述 5 个问题为模型, 来探讨解决方案。下面分别对这 5 个问题的现有方法进行综述。

3 各种场景下的攻击源追踪

3.1 虚假 IP 地址攻击溯源问题

当攻击数据包中的源 IP 地址是伪造的时, 如何找到发送攻击数据包的真实 IP 地址? 这一问题也被称为 IP 追踪(IP Traceback)。文献[83]对该问题的早期研究进行了总结, 按照不同的分类方法, 将方法分为包标记/包记录、概率性方法/确定性方法等。本文根据攻击事件发生前, 事先需要采取的措施和知道的信息, 分为如下 5 种情况进行讨论。

图 7 IP 数据包中 IP 首部数据结构

Figure 7 The data structure of IP header in packets

1) 取证人员可以控制骨干网络上的全部或大部分路由器, 并且可以修改路由软件

这一条件的含义是, 取证人员可以在事先给骨干网络的路由器增加新的功能, 在不影响正常路由的情况下修改标准的 IP 协议, 以帮助发现真实的 IP 地址。

基于这一条件的方法主要有概率包标记算法、确定包标记算法、ICMP 标记算法等。分述如下:

(1) 概率包标记算法

概率包标记算法(Probabilistic Packet Marking, PPM)在文献[1]中被提出。文献[2]几乎在同时独立提出了类似的思路, 但相对来说, 文献[1]对该问题的研究更全面。假设攻击数据包的发包频率远高于正常的数据包, 其基本思想是让路由器对经过的每一

个数据包以一定概率在 IP 数据包首部中(IP 数据包结构见图 7)进行标记, 从而让受害主机能重构出攻击路径。

文献[1, 3]对包标记算法进行了详细研究, 提出了从简单到复杂的 3 个算法:

第一个算法是最简单的算法: 在每个数据包头中加入所经过的每个路由器的 IP 地址。但该方法显然不可行, 最主要的原因是由于路径长度事先不可知, 包头中不可能有充足的空间存放所经过的每个路由器的 IP 地址。这一思想也是最初的确定包标记算法的思想。

第二个算法称为节点采样算法, 即每个路由器以一定概率在数据包头中加入自己的 IP 地址。受害主机对收到的数据包头中具有相同 IP 地址的数据包进行计数, 然后按照计数大小排列相应的 IP 地址, 即得到攻击路径。该方法的理论基础是, 由于每个路由器都是以相同概率随机选择进行标记的数据包, 前面路由器已经标记过的数据包可能会被后面路由器的标记替换掉, 所以离受害主机越远, 受害主机所收到的该路由器标记过的数据包越少, 从而可以以数据包的个数来区分路由器的远近。但这种方法要达到很高的准确率, 需要大量的数据包。

第三个算法称为边采样算法, 主要目的是为了减少重构攻击路径所需的数据包。它在包头中插入三个域{开始, 结束, 跳数}, 记录数据包路径上的两个路由器的 IP 地址和跳数。当数据包经过中间路由器时, 路由器以一定的概率对数据包进行标记, 若数据包在此之前没有经过标记, 则把自己的 IP 地址填入包头中的“开始”域, 并把跳数设为 0。对于不需要标记的数据包, 若数据包已经经过标记, 并且跳数为 0, 则把自己的 IP 地址填入包头中的“结束”域, 并把跳数加 1; 若跳数不是 0, 则只把跳数加 1。该算法的实质是将相邻的两个路由器的 IP 地址存入包头中, 而跳数代表了这两个路由器离受害主机的远近程度。

由于边采样算法需要把 2 个 32bit 的 IP 地址和 8bit 的跳数共 72bit 加入包头中, 这仍然不可行, 因此对算法进行了改进。第一个措施是首先把“开始”域中的 IP 地址和“结束”域中的 IP 地址进行异或操作, 只在包头中加入异或后的结果。由于离受害主机最近的路由器, 没有后续的路由器进行异或操作, 因此它的 IP 地址保留下来, 受害主机利用它的 IP 地址和其他数据包包头中的信息进行异或操作, 就可以逐级恢复出上游路由器的 IP 地址。第二个措施是将路由器的 IP 地址分段, 将每个分段填入不同的数据包, 从而进一步缩小所需的包头空间。具体来说, 只需利用 IP 包头中的 16bit 的“标识”字段, 其中 3bit 作为分段的偏移量, 5bit 作为跳数, 8bit 作为分段。该算法为了识别出多个攻击源, 避免多个攻击源的分段发生冲突, 使用 hash 函数, 对 32bit 的 IP 地址计算 hash 值, 将 32bit 地址扩展为 64bit, 也就是说需要 8 个分段才能表达完整的地址信息, 这也是为什么需要 3bit 作为分段偏移量。该算法的实验结果表明, 这一算法仍然需要上千个数据包才能以高准确率重构出攻击路径。

上述算法被提出后, 文献[4, 5]分析了所存在的问题, 主要包括:

第一个问题: IP 包头空间受限问题。上述算法主要利用 IP 数据包头中的 16bit 的标识字段(identification)来填加路由器的标记。但由于 IP 数据包头中的空间不足以填写完整的路径信息, 因此路由器需将 IP 地址分段, 在多个 IP 数据包中写入分段的 IP 地址。但由于多个攻击数据包可能从不同的路径路由到受害主机, 这导致受害主机进行 IP 地址重组的时候, 必须考虑 IP 地址分段的所有组合情况, 计算负担较重, 错误率也较高。而且, 当数据包经过的跳数较长时, 恢复出完整的 IP 地址会更加困难。

第二个问题: 标记数据包的概率。主要的问题在于数据包经过的路由器的跳数越多, 受害者能够重构出攻击路径的可能性就越小, 因此就需要更多的攻击数据包才能保证重构出攻击路径。如果攻击者在一次攻击过程中发送的攻击数据包达不到要求, 则该方法失效。所以这一方法的关键是数据包作标记的概率与所需数据包个数的折中, 以及如何尽可能降低所需的数据包个数。

第三个问题: 如果骨干网络上只有部分路由器支持 PPM 算法, 受害主机能否重构出攻击路径?

第四个问题: 如果存在多个攻击源同时发送攻击数据包, PPM 算法能够识别出这多个攻击路径?

第五个问题: 如果攻击者了解路由器所执行的 PPM 算法, 有意识的在 IP 数据包中填加精心构造的虚假的包标记, 以误导受害主机, 那么 PPM 算法是否还能准确地重构出攻击路径?

针对这些问题, 很多文献进行了研究, 给出了一些解决方案:

针对第一问题, PPM 算法的提出者考虑到了这个问题, 他们是采用对 IP 地址进行分段的方式来解决。而其他的思路有:

文献[6]不是在包头中直接加入路由器的 IP 地址, 而是先对 IP 地址进行 hash 计算, 来减少数据的长度,

但这一方法需要受害主机知道每个 hash 值所代表的 IP 地址。同时针对第五个问题, 为防止攻击者发布虚假的标记, 文献[6]还提出一个采用密码学方法对标记进行认证的方法。

文献[7]也不采用直接的 IP 地址, 而是转化为多项式重构问题, 利用编码理论将路径信息嵌入包头, 在受害主机上利用多项式求解重构出攻击路径。他们给出了三种算法, 分别是确定性路径编码算法、随机化路径编码算法、边缘编码算法。

确定性路径编码算法: 每个路由器在数据包中插入两个数值, 一个是随机数 x_j; 另一个是 $y_j=(y_{j-1}\bullet x_j+IP_j)$ mod p, y 的初始值为 0。受害主机只要收到 d 个数据包, d 为路径长度, 求解线性矩阵, 就可以得出路径上的每个 IP 地址值。

随机化路径编码算法: 由于上述算法 y 初始值为 0, 暴露了第一个路由器, 因此在随机化路径编码算法中, 第一个路由器也随机选择 y, 但这将导致受害主机恢复出的路径含有不必要的前缀。

边缘编码算法: 由于当路径长度 d 很长时, 受害主机求解的计算量仍然很大, 因此借鉴原 PPM 算法中边采样的思想, 增加一个类似跳数的参数 l, 该参数 l 小于路径长度 d。第一个路由器在包头中加入参数 l, 随后的路由器将 l 逐跳减 1, 当参数 l 变为 0, 则路由器不再在包头中填加标记。该方法减少了受害主机进行重构的计算量, 同时该方法也能解决第五个问题。由于攻击者很难在事先知道路由器所选择的随机数, 因此很难伪造包标记。

文献[111,102]对文献[7]中基于网络编码的算法进行改进, 不但可以减少重构攻击路径所需的数据包, 还可以同时重构出多条攻击路径。

文献[8]提出了一种 ASPPM 算法, 该算法考虑到 PPM 算法可能会暴露骨干网络的拓扑结构, 导致骨干网络的运营商不愿意支持 PPM 算法, 因此提出给路由器分配 ID 号, 用 ID 号代替 IP 地址在数据包头中作标记。并且, 这一算法能在数据包头中存放完整的 ID 号, 避免了 IP 地址分段的问题。

文献[9,97]从理论上证明 PPM 算法中, 数据包头中进行标记所需的最小 bit 数为 1, 并研究了所需标记的 bit 数与所需数据包数的关系。

针对第二个问题, 文献[10, 11]指出 PPM 算法中, 由于每个路由器对数据包做标记的概率都一样, 这导致离受害主机最远的路由器所打的标记有可能被后续的路由器的标记所覆盖, 这使得受害主机需要更多的数据包来重构路径, 而且重构路径的时间较长, 因此提出了一种基于距离的变概率标记方法, 提高距离越远的路由器对数据包打标记的概率。文献[12, 13]基于数据包头中的 TTL 值来计算给数据包打标记的概率, 这样做的目的是降低攻击者伪造标记所造成的影响。文献[14]沿着这一思路, 对算法进行了改进。

针对第三个问题, 文献[6, 15]进行了研究。文献[6]利用其他工具, 如 traceroute, 来获得受害主机上游路由的拓扑结构来推测缺失的攻击路径。而文献[15]利用数据包标记来获得受害主机上游路由的拓扑结构。对于如何区分经过中间路由器的跳数, 文献[15]利用数据包头中的 1bit 来标记跳数。

针对第四个问题, 如果存在多个攻击源, 文献[8, 15-18]采用 PPM 算法以受害主机为根, 构造攻击路径构成的攻击树, 从而识别出多个攻击源。

针对第五个问题, 除了上述提到的文献[6, 7], 文献[19]从网络取证的角度, 利用 HMAC-SHA1 机制保证路由器标记的完整性, 防止其他人修改或者伪造包标记。

(2) 确定包标记算法

确定包标记算法(Deterministic Packet Marking, DPM)与 PPM 算法是类似的思想, 只是 DPM 算法对每个数据包都进行标记, 只需少量的数据包就可以发现攻击源。更确切的说, 该算法应被称为边缘路由器包标记算法。

文献[20]质疑 PPM 算法是否有必要重构出整个攻击路径, 因为溯源的最终目的是定位攻击源, 因此提出了算法 DPM, 只需要骨干网络的边缘路由器对数据包进行标记, 中间路由器不需要对数据包进行标记。该算法使用了 IP 包头的 16bit 数据包 ID 标识域和 1bit 的 flag 域。当每一个数据包进入骨干网络时都需要进行标记。由于所使用的数据包包头没有足够空间存放 32bit 的 IP 地址, 因此需要两个数据包来存放 IP 地址, 1bit 的 flag 用来表示是 IP 地址的前半部分还是后半部分。而数据包包头中存放 IP 地址前半部分还是后半部分是由路由器随机决定。受害主机只能知道攻击数据包来源的路由器 IP 地址, 而不知道数据包所经过的路径。

文献[21]对文献[20]中的 DPM 算法进行了改进。作者指出如果存在多个攻击源, 在原 DPM 算法中, 将会有多个路由器对 IP 地址进行分段, 这导致受害主机在重构 IP 地址时, 无法分清哪个分段来自哪个路由器, 不能准确地重构出路由器的 IP 地址。因此, 作者采用对 IP 地址计算 Hash 值, 将 IP 地址分段、Hash 值、分段偏移一起填入 IP 头部, 从而受害主机在重构 IP 地址时, 计算 IP 地址的 Hash 值可以验证

所重构 IP 地址的正确性。

同时, 文献[21]还考虑了数据包分片对 DPM 算法的影响。由于在不同的链路, 所允许的最大分组长度MTU不同, 当分组长度超过MTU时, 需要对数据包分片, 而分片使用的正是 IP 头部的“标识”字段。这一问题在 PPM 算法中没有太大的影响, 因为 PPM 算法采用概率标记, 遇到分片数据包的可能性可以忽略, 但在 DPM 算法中, 由于需要对每一个数据包作标记, 这一问题就需要郑重考虑。若路由器遇到分片的数据包, 在执行 DPM 算法时, 会更改 IP 头部的“标识”字段, 导致目的主机无法重组分片的数据包。针对这一问题, 路由器需要记录在第一个分片数据包中所填加的标记, 必须在后续的所有分片数据包中填加同样的标记, 这样目的主机就可以重组分片的数据包。

文献[18]给出一个新的 DPM 算法, 类似于文献[21]的标记编码方法, 但采用可变标记长度策略来适应不同的网络环境。同时, 结合 PPM 算法思想, 考虑到路由器工作负载的大小, 若路由器负载超过一定阀值, 则有选择的对数据包进行标记, 当低于该阀值, 才对全部数据包进行标记。

文献[22]则采用了一种新的对 DPM 算法中包标记的编码方法, 路由器根据数据包来源的不同进行标记, 第一个路由器只标记 1bit 的 0, 随后的路由器在后面附上自己的标记, 由于每个路由器进行标记的 bit 数很少, 能够在 IP 头部存放整个路径的标记信息。当受害主机确定某个数据包是攻击数据包, 需要进行追踪时, 根据数据包头部的标记向上游路由器进行查询, 路由器依次告知该数据包的来源。

由于 IPv6 协议的包头结构完全不同于 IPv4 协议, 并且 IPv6 地址是 128bit, 远大于 IPv4 地址的 32bit, 因此文献[74]研究了在 IPv6 网络中利用 DPM 算法进行攻击溯源的方法。作者指出, IPv6 协议的基本头是固定的 20 字节, 不适合用于嵌入边缘路由器的 IPv6 地址, 而是采用目的选项扩展头 DOH。在该扩展头中有足够的空间可以嵌入全部的 128bit IPv6 地址, 大大简化了受害主机进行攻击路径重构的复杂度。

(3) ICMP 标记算法

ICMP 标记算法, 不同于 PPM 和 DPM 算法对 IP 数据包进行标记, 而是对 ICMP 协议数据包进行标记。文献[23, 24]提出了这一个思路。路由器以一定概率随机发送 ICMP 报文给目的 IP 主机, 告知该路由器的 IP 地址、前一跳路由器的 IP 地址、后一跳路由器的 IP 地址。当主机需要进行回溯攻击数据包的真正源时, 通过各个路由器发送过来的ICMP报文可以重构出攻击数据包的路径。文献[25]对这一思路进行了改进, 将整个路径进行编码保存在ICMP报文中, 提高了重构路径的性能。文献[99]针对这一方法在存在背景流量或多个攻击流量时, 攻击路径重构准确率下降的问题, 提出在 ICMP 报文中增加管理信息库(Management Information Base)信息, 从而提高准确率。

但这一方法的问题是, 目前很多路由器会过滤掉 ICMP 报文, 或者限制 ICMP 报文的发送频率, 并且攻击者很容易发送伪造的 ICMP 报文来迷惑受害主机。2000 年 IETF 成立一个工作组根据这一思路来制定标准草案[23], 但目前所制定的标准草案已被撤销。

(4) 组合方法

除了上述三种基本算法外, 还提出了一些组合的方法。文献[26-29]采用了数据包标记和数据包记录(该方法将在下面详细介绍)的混合方法。文献[30]综合了 ICMP 和 PPM 算法, 路由器对于 IP 数据包以一定概率进行标记, 并且同时把 IP 地址填入 ICMP 包中。ICMP 数据包用于区分 IP 数据包中的标记是否是攻击者伪造的, 能够抵抗攻击者伪造标记。文献[91]在文献[6]的基础上, 借鉴 PPM 算法的思想, 采用自适应概率进行标记, 若数据包已经经过了标记, 则减少再进行标记的概率, 从而减少后续节点标记覆盖前面节点标记的概率, 提高重构攻击路径的准确性。

对于跨自治域的追踪系统, 文献[85,73]进行了相关研究, 在 IP 头部插入自治域号, 设计了相应的系统框架。文献[106]同时在 IP 头部插入自治域号和路由器 IP 地址的 hash 值, 减少重构攻击路径所需的数据包数。文献[110]设计了一种域间的追踪 overlay 网络, 利用 BGP 信息和包标记思想, 实现攻击源追踪。

2) 取证人员可以控制骨干网络上的路由器, 但不能修改路由软件

这一条件的含义是, 取证人员可以事先观察记录流经骨干网络路由器的 IP 数据包, 但不能改变标准的路由协议。

主要思路是, 在路由器上记录所有流经的数据包, 当攻击发生时, 受害主机向其上游路由器进行查询, 路由器比对所记录的数据包, 可以构造出该数据包所经过的路径。该方法优点是可以回溯单个数据包, 但缺点是需要考虑路由器存储空间受限的问题。

文献[32, 33]设计了一个追踪系统 SPIE, 不是让路由器记录整个数据包, 而是利用 bloom filter 记录数据包的摘要, 大大减少了所需的存储空间。然后通过查询每个路由器上的数据包摘要, 可以重构出攻

击路径。具体来说, 每个路由器计算每个数据包的摘要值, 用来计算的数据包要除去 IP 包头中的服务类型、TTL、首部校验和、可选项这四个字段, 然后生成 32bit 摘要值来代表每个数据包。在保存时, 并不直接保存 32bit 的摘要值, 而是利用 bloom filter 将摘要值映射到一个连续的比特表, 并置为 1, 也就是说在该表中的位置即为摘要值。由于该表的存储空间有限, 需要设定一个刷新时间。刷新时间到, 则原有的记录被清空, 不再保存, 开始保存新的数据包记录。当受害主机要求追踪某个数据包时, 向其上游路由器进行查询, 路由器计算数据包的摘要值, 查看在bit表相应位置是否是1, 若是, 则该数据包经过了该路由器, 否则则说明没有经过该路由器。上游路由器逐个迭代该过程, 则重构出数据包经过的路径。由于无论采取什么 hash 函数总会存在冲突, 这一冲突会导致上述方法出现误报, 而增加存储空间会减少冲突的发生, 这就要求在存储空间和误报之间进行折中。文献[34]在文献[33]的基础上进一步减少所需的存储空间。

文献[35]研究了只有部分路由器进行数据包记录的情况, 并且对于多个攻击源问题, 该方法只需要追踪属于多个攻击源的数据包就可以识别出多个攻击源。

文献[100]提出一种基于Session的数据包记录方法, 即只记录 TCP 数据流中的连接建立请求 SYN 数据包和连接终止 FIN 数据包, 忽略掉流中间的数据包, 从而大大减少所需的存储空间。

文献[31]针对跨自治域的追踪问题, 利用路由器的IP包记录方法, 结合链路层的MAC地址来识别虚假 IP 地址, 实现了一个原型系统。文献[77]针对文献[31]中需要记录全部 IP 包, 导致路由器存储空间消耗过大的问题, 提出了一种概率包记录方法, 根据设定的概率只记录部分 IP 包, 减少对存储空间的消耗, 但这一方法不能追踪攻击数据包较少的情况。

文献[103, 104, 107]不是记录网络数据包, 而是利用路由器监控网络性能, 记录相应的特征, 利用这些特征来进行攻击追踪。

由于在这种情况下不能改变现有路由结构, 另外一个思路是在现有路由结构上建立一个覆盖网络(Overlay Network), 通过新设计的覆盖网络来实现数据包跟踪[36]。文献[87,90]给出一种新的信令协议, 用于路由器生成攻击路径记录数据包, 并转发给受害主机, 帮助受害主机识别出攻击源。文献[92,94,96]设计了一种跨域攻击追踪系统, 通过在骨干网上部署额外的组件来传递关于数据包来源的信息, 帮助受害主机识别攻击者。文献[101]利用 GRE 隧道建立与路由器的连接, 通过与路由器的信息交换来追踪攻击数据包的来源。

3) 取证人员不能控制骨干网络上的路由器, 但可以在网络上部署监控器

这一条件的含义是, 取证人员只能在网络合适的位置部署监控器收集数据包, 这里的网络不是指骨干网络, 而是指终端网络。

其基本思想是, 在大流量数据包情况下, 由于网络阻塞等各种原因, 路由器会有一定几率产生目标不可达的 ICMP 报文, 由于攻击数据包的源 IP 地址是虚假的, 一般是随机产生的, 这些 ICMP 报文会被发往这些虚假的 IP 地址, 其中包含路由器的 IP 地址以及原数据包的源和目的 IP 地址。因此部署在网络上的监控器会收到这些ICMP报文, 根据发送这些ICMP 报文的路由器, 可以构造出这些数据包的攻击路径。文献[78,37]提出了该方法, 利用 Network Telescope 项目(能够覆盖 1/256 的 IPv4 地址)收集的数据, 结合骨干网的拓扑结构, 可以在一定程度上发现攻击源。

这种方法要求攻击数据包的流量比较大, 并且在攻击正在进行的时候实施, 一旦攻击结束, 这种方法就无法找到真实的 IP 地址。

4) 取证人员既不能控制骨干路由器, 也不能部署监控器, 但知道骨干网络拓扑结构

这一条件的含义是, 取证人员只知道骨干网络的拓扑结构, 没有权限控制骨干网中的路由器, 也没有条件部署遍及全网的监控器。

文献[38]提出了一种链路测试的方法。在大流量数据包的情况下, 从被攻击目标出发, 由近及远, 依次对被攻击目标的上游路由器进行 UDP 泛洪。若某条链路上存在攻击流量, 由于泛洪流量的存在, 将导致攻击流量丢包。根据这一现象, 即可以判断出某条链路上是否存在攻击流量, 从而构造出攻击路径。

该方法只能对单个攻击流量进行检测, 若同时存在多个攻击流量, 则很难区分不同的攻击流量。这种方法同样要求攻击数据包流量较大, 并且一旦攻击结束, 方法也就失效了。另外, 这种方法本身就是一种 DoS 攻击, 会影响正常的数据流量。

文献[39]延续了链路测试的思想, 提出了一种基于蚁群的算法, 即受害主机发出一些蚁群, 这些蚁群根据链路中负载的程度来选择路径, 链路负载越大说明越可能是攻击流量, 因此蚁群选择该路径的概率越大。当所有蚁群达到所监控网络边缘时, 根据蚁群所走过的路径, 则可以构造出最有可能的攻击

路径。

事实上文献[38, 39]所提方法仍然需要中间路由器的支持, 并不符合“取证人员既不能控制骨干路由器, 也不能部署监控器, 但知道骨干网络拓扑结构”的前提条件。本文对文献[38]的方法稍加改进, 提出一种改进方法的思路, 就可以在满足前提条件的情况下, 对攻击源进行追踪, 如图 8 所示。

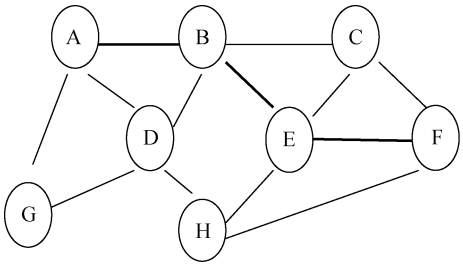

图 8 改进的链路测试方法

Figure 8 The improved method for link testing

假设上图所示为骨干网络路由器的拓扑结构, 其中 A 为离攻击者主机最近的路由器, F 为离受害者主机最近的路由器, 攻击路径为 A→B→E→F。

根据文献[38]的链路测试方法, 首先需要在路由器 F 上执行链路测试, 测试出 E→F 属于攻击路径。然后在路由器 E 上执行链路测试, 测试出 B→E 属于攻击路径。然后在路由器 B 上执行链路测试, 测试出 A→B 属于攻击路径。从而重构出整个攻击路径。

对该方法稍加改进, 只需在路由器 F 上执行链路测试。具体来说,

第一步对与路由器 F 直接相连的路由器进行测试, 即测试 F→H、F→E、F→C 三条链路, 发现 F→E 链路上存在攻击流量。

第二步由于知道整个网络的拓扑结构, 利用 IP 数据包的源路由功能, 测试 F→H→E 不存在攻击流量; 同样可知 F→C→E 也不是攻击路径。再测试 F→H→E→B 或者 F→C→E→B, 则可以发现 B→E 上存在攻击流量。

第三步依次类推, 则可以发现 F→C→B→A 上存在攻击流量, 则可以发现攻击源 A。

这种方法需要根据拓扑结构, 依次找到只包含攻击路径上一条链路的第二条路径, 利用源路由功能进行链路测试, 如果找不到这样的路径, 则该方法失效。

5) 取证人员既不能控制骨干路由器、不能部署监控器, 也不知道拓扑结构

这一条件的含义是, 取证人员不掌握任何资源, 在这一条件下似乎不可能追踪到真实的 IP 地址。但可以采取某种方法, 获得骨干网络的拓扑结构, 从而将问题转化为(d)中的情况, 如文献[6]提到的利用 traceroute 获取网路拓扑结构。

对一般虚假 IP 溯源问题的解决方案总结如表 1 所示。

表 1 一般虚假 IP 溯源问题的解决方案

Table 1 The general solution for false IP traceback

网络安全人员所掌握的资源	检测方法
控制路由器, 修改路由软件	给数据包打标记
控制路由器, 但不能修改路由软件	记录数据包
不能控制路由器, 但能在网络上部署监控器	利用路由器自然产生的 ICMP 错误报文
仅知道网络拓扑结构	链路泛洪测试
不了解任何信息	获得网络拓扑, 再采用链路泛洪测试

对于特殊的“反射攻击”, 由于受害主机收到的攻击数据包中的源 IP 地址是真实的 IP 地址, 因此一种方法是定位某个反射跳板所在的网络, 以反射跳板为起点, 将问题转换为上述一般性虚假 IP 溯源问题; 另外一种方法是越过反射跳板, 直接追踪定位攻击者主机。

文献[37]提到了一种越过反射跳板, 直接定位攻击主机的方法。攻击数据包在经路由器进行路由过程中, 由于这些数据包的数量巨大, 会有少量数据包因为各种原因, 如网络阻塞, 不能送达目的 IP 地址, 因此路由器会生成 ICMP 错误报文。由于攻击主机在发送反射攻击数据包时, 其源 IP 地址是受害主机的 IP 地址, 因此这些 ICMP 错误报文会被送到受害主机。这些 ICMP 错误报文中包含了发送路由器的 IP 地址, 受害主机结合骨干网络的拓扑结构, 可以在一定程度上发现这些攻击数据包的发送者。

3.2 僵尸网络中的攻击溯源问题

对于僵尸网络中的攻击溯源问题, 取证人员首先需要对 bot 程序能够进行分析, 了解 bot 与 C&C 服务器通信的机制, 在此基础上, 对 C&C 服务器及 Botmaster 进行追踪。根据取证人员事先掌握的资源, 分为以下 4 种情况进行讨论:

1)botnet 中的主机, 包括 bot 和 C&C 服务器, 位于取证人员的管辖范围之内, 取证人员可以物理或远程访问这些主机。

文献[108]设计了一个基于主机对 botnet 进行追踪的框架, 可以完成两个功能: 一是从受害主机到 bot 的追踪, 二是从 bot 到 C&C 服务器的追踪。该框架要求 botnet 中的所有节点都位于取证人员的控制范围之内, 并且在每台主机上安装有监控软件。当受害主机检测到攻击后, 通知管理中心, 管理中心将受害主机的 IP 地址下发给所有的节点, 各个节点检查是否曾经向受害主机发送过数据包, 若发送过, 则被认为是 bot 节点。然后, 采用类似的方式, 从 bot 节点追踪到 C&C 服务器。

2) 取证人员能够控制至少一个 bot 主机, 并且能够在骨干网络上部署多个监控器

bot 一般要回传信息给攻击者, 如果 C&C 服务器只是简单中继 bot 的回传信息, 或者 bot 回传的信息不经过 C&C 服务器直接传送给攻击者, 由于网络安全人员可以控制 bot 主机, 则可以在 bot 的回传信息中填加某种标记, 然后对标记进行跟踪。但如果 bot 回传的信息可能会在 C&C 服务器上进行聚合, 例如延迟一段时间将多个 bot 的回传信息统一发送给攻击者, 则需要考虑其他方法。

文献[41]对于基于 IRC 的 botnet 进行了研究, 假设网络安全人员可以控制一个 bot, 并可以修改 bot 回传给攻击者的信息。若该信息没有经过加密, 则在数据包中填加一些字符, 改变数据包的长度, 在骨干网络上部署监控器, 检测数据包中的这些特征; 若信息经过了加密, 则采用长度-时间混合方法, 即改变数据包的长度和发送数据包之间的间隔时间, 同样在骨干网络上部署监控器, 检测数据流中的这些水印信息。该方法要求 C&C 服务器不会对 bot 发送的信息进行聚合或者延迟, 否则网络安全人员所填加的水印信息将完全丢失。

当由 bot 追踪到 C&C 服务器后, 由于 C&C 服务器必然会向攻击者回传一些信息, 则取证人员可以利用类似的方法, 即利用这些回传数据包, 对其内容进行修改, 加入一些特征, 如特殊字符串(针对明文信息)、数据包的大小、数据包发送的时序(针对密文信息)等。或者通过某些通信特征进行识别, 因此可以通过部署在骨干网络上的监控器, 对数据流量进行检测, 发现这些流量特征, 构造出数据包传输路径, 从而识别出攻击者的主机。

文献[116]设计了一种从 C&C 服务器到 Botmaster 的追踪机制。该机制研究了 C&C 服务器的通信模式, 利用这些特征来追踪真正的攻击者。文献[117]利用类似的思想, 研究了 P2P 型 botnet 中的通信模式, 通过部署监控器来追踪数据流的来源。

3) 取证人员虽然能控制至少一个 bot 主机, 但不具有在骨干网络部署监控器的资源

在这种情况下, 同样可以利用 C&C 服务器向攻击者回传信息, 但网络安全人员需要了解攻击者用什么样的程序来接收信息, 研究该软件是否存在可远程执行代码的漏洞, 如存在, 可以在回传信息中构造可执行代码, 利用该漏洞使得代码在攻击者的主机上执行, 向网络安全人员发送攻击者主机的信息。

另外, 由于攻击者在 bot 上可能会搜集一些敏感文件, 如果这些文件是 PDF、word 等可嵌入可执行代码的格式文件, 则可以在这些文件中嵌入可执行代码。对于这些文件 C&C 服务器一般不会进行加工处理, 可以保证这些文件原样到达攻击者。当攻击者打开这些文件时, 内嵌的可执行代码执行, 可以获得攻击者主机的信息, 发送给取证人员。

文献[42, 43]利用类似的感染攻击者的可执行代码来定位攻击者的位置。作者搭建了一个实验性的 Zeus botnet, 首先控制了一台 bot, 利用内存取证的方法在其内存中获得 bot 与 C&C 服务器之间加密通信的加密密钥, 然后在 C&C 服务器上上传了一个可执行代码, 当攻击者访问 C&C 服务器时, 利用 botnet 软件本身的漏洞, 该可执行代码将会被下载到攻击者本地执行, 从而获得攻击者主机的一些信息。

4) 取证人员不能控制任何已有的 bot 主机

在这种情况下, 主要的思路是利用 honeypot 主动感染 bot 程序, 成为僵尸网络中的一员[40], 然后将问题转化为上述 3)中的场景, 再进行追踪。

对僵尸网络溯源问题的解决方案总结如下表所示:

表 2 僵尸网络溯源问题的解决方案

Table 2 The solution for botnet traceback

网络安全人员所掌握的资源		检测方法
能控制 bot 主机和 C&C 服务器		检查各主机日志是否有访问受害主机的记录
仅能控制 bot 主机, 但能够在网络上部署监控器	bot 的回传信息是明文	在明文中填加特殊字符串作为水印信息
	bot 的回传信息是密文	在密文数据包流中填加流水印
仅能控制 bot 主机, 且不能在网上部署监控器		1) 制造虚假的敏感文件, 嵌入可执行代码; 2) 利用 botnet 本身的软件漏洞
不能控制 botnet 中任何已有的主机		先利用 Honeypot 成为一个 bot, 再进行追踪

3.3 匿名网络中的攻击溯源问题

匿名网络, 如 Tor, 通过保护通信双方的身份信息, 能有效防止用户个人信息的泄露, 成为一种新的网络访问方式。但同时, 攻击者也看到了匿名网络所带来的匿名性, 可以利用匿名网络发动网络攻击, 逃避司法取证人员的追踪调查, 因此研究匿名网络中的攻击溯源问题也就成为了近年的一个研究热点。

文献[71]综述了匿名网络中的攻击溯源问题, 将攻击溯源方法分为两类: 匿名网络调制追踪和匿名网络渗透追踪。前者是指取证人员在匿名网络流量中添加流水印信息, 通过检测流水印信息将不同的网络流量关联起来, 从而识别网络流量的来源。后者是指取证人员控制部分匿名网络的节点, 通过破坏或查看通过这些节点的流量, 来识别网络流量的源头。这两类方法需要取证人员掌握的资源有所不同, 因此从取证人员的角度, 本文将匿名网络中的攻击溯源方法分为如下 2 种情况:

1) 取证人员能够控制全部或部分匿名网络的节点

文献[80]采用数据包标记方法[1]来识别匿名流量。由于匿名网络 Tor 隐藏了数据发送端的 IP 地址, 而且中间节点会剥去数据包的外层包头, 因此该方法提出利用 Tor 协议中网络层的 32bit 的 GMT(包终止时间)字段, 将其改造为 5bit 的距离和 19bit 的 IP 地址 hash 值。中间节点填入相应的信息, 受害主机根据这些信息可以推算出数据流的来源。

文献[112]针对 Tor 网络的追踪问题, 利用 Tor 网络所采用的 AES 算法的计数器加密模式, 需要计数器进行同步的特征, 控制多个 Tor 网络的节点, 人为改变节点的计数器, 导致后续节点解密失败, 从而将节点间的通信流进行关联。

文献[75]研究了高延时匿名网络、匿名内容发布网络 Freenet 中的匿名追踪问题, 指出可以利用网络中部分受控制的节点作为监控节点, 通过监控节点所观察到的消息来推断内容请求者的真实主机, 但该方法没有解决内容发布者的匿名追踪问题。文献[86]研究了 BitTorrent 网络中初始种子的识别问题, 给出一些识别特征。文献[115]研究了在种子初始传播阶段识别第一个上传者的规则。

2) 取证人员能够控制匿名网络的部分通信流量, 并能部署检测传感器

文献[88,82]采用数据包发包速率来作为载频的方式嵌入水印, 如用较高的发包速率表示二进制 1, 用较低的发包速率表示二进制 0。该方法可以有效抵抗由于各种原因引起的数据包的网络时延、丢包、增加包、重打包等对水印检测准确率的影响。

文献[113]针对采用发包流率作为载频, 而在实际应用中发包流速不稳定的问题, 提出了一种以时隙质心的载频方法, 即将时间分为多个时间片, 以数据包落在时间片中的位置为质心, 当数据包数量足够多时, 无论发包速率如何波动, 时隙质心是稳定的。这样可以提高流水印的健壮性和隐秘性。

文献[114]将直序扩频机制 DSSS 应用到流水印中, 不是直接在流中嵌入特征码, 而是利用伪噪声码 PN 码进行扩频, 嵌入到数据流中, 提高了流水印的隐蔽性。但文献[95]对该方法的隐蔽性进行了充分实验验证, 指出该方法的隐蔽性需要进一步提高。

文献[84]综合利用时隙质心和 PN 码的方法提出一种新的流水印方法。作者指出长 PN 码可以提高水印检测的准确率, 这一点在文献[81]中得到验证。文献[81]采用基于 DSSS 的长 PN 码来嵌入水印, 该方法能够阻止攻击者检测到流中可能存在水印信息, 并且可以用于多个数据流嵌入水印的同时检测。文献[98]更进一步采用双重时隙质心和 PN 码的方法, 即将相邻时隙的质心联合进行考虑, 可以更好的提高水印的隐蔽性。

文献[118]不直接针对时隙进行水印的嵌入, 而是使用包数来表达嵌入的信号, 即用连续的 1 个数据包代表“0”; 用连续的 3 个数据包代表“1”。考虑到网络延时及缓冲队列造成的数据包合并、分离, 提出了鲁棒的算法来提高检测率。这样, 对于流持续时间较短的情况, 仍然可以添加水印并能得到较好的检测效果。文献[119]进一步指出, 虽然 Tor 匿名网络把应用层数据分割为固定大小的 512 字节(含头部信息, 真正数据是 498 个字节), 但这一机制并不会导致网络层数据包分组也是固定大小。利用这一现象, 文献[119]给出了一个新的嵌入水印的方法, 即利用真正数据的 498 个字节代表“0”, 2444 个字节代表“1”。前者将被分割为 1 个数据包, 而后者考虑到链路层的最大数据帧长度 1500 字节, 将被分割为 5 个数据包。然后研究了数据包在网络层的分割、融合的各种情况, 给出了水印检测算法。实验结果显示能够使用较少的数据包以较高的准确率检测到嵌入的水印信号。文献[120,121]将类似的思想应用到另一匿名网络 Anonymizer 中, 可以识别出访问网站的匿名客户端。

上述方法是针对单个数据流嵌入水印并进行检测的, 而如果同时存在多个数据流, 在多个数据流中嵌入水印, 将导致检测准确率急剧下降。文献[79]针对这一问题, 提出了一种针对多个数据流嵌入水印的方法, 即事先生成一个种子序列, 对每个数据

流随机选择不同的种子, 在数据流中不同的时隙段内嵌入水印。当进行检测的时候, 解码器尝试采用每一个可能的种子来检测水印, 找出最匹配的值, 从而解码出嵌入的水印值。

文献[76]对于为什么可以采用时隙特征来嵌入水印的理论原因进行了解释, 得出了影响追踪效果的三个因素: 报文长度、干扰数据流中数据包的数量、转发节点的数据包缓冲时间, 并给出了在给定检测准确率的情况下, 所嵌入水印的最小延迟时间的计算方法。

一般的方法采用在固定时间长度的数据包间隔中嵌入水印, 常用的水印检测方法效率较差, 需要大量的数据包才能获得较高的检测准确率。文献[72]针对这一问题, 基于 SPRT(Sequential Probability Ratio Test)检验方法, 提出三种不同的假设检验构造方法, 在达到同样检测准确率的情况下, 明显减少了所需的数据包数量。

这一类方法仅适用于低延时匿名网络, 对于高延时匿名网络将导致所添加的流水印信息丢失。

对匿名网络溯源问题的解决方案总结如下表所示:

表 3 匿名网络溯源问题的解决方案

Table 3 The solution for anonymous network traceback

网络安全人员所掌握的资源		检测方法
能够控制部分匿名网络节点	低延时匿名网络, 如 Tor	1) 在匿名网络协议包中插入标记; 2) 利用匿名网络协议的某些特征;
	高延时匿名网络, 如 Freenet	通过监控匿名分发的内容特征
不能控制匿名网络节点, 但能够控制部分匿名网络流量, 并部署检测传感器		在数据流中添加水印信息, 仅适用于低延时匿名网络

3.4 利用跳板主机的攻击溯源问题

在这一问题中, 由于在受害主机端可以观察到最后一跳的 IP 地址, 因此追踪问题就转化为如何沿着攻击路径上的跳板, 逐跳验证是否确实存在“跳板主机”。

因此, 问题 4 可以分为两个子问题:

问题 4.1: 跳板检测(Step-stone Detection), 即如何确定本地网络中存在攻击者的跳板?

问题 4.2: 在本地网络中发现跳板后, 如何追踪定位攻击者主机?

3.4.1 问题 4.1

在这一问题中, 假设网络安全人员完全控制本地网络, 能够对进出本地网络的网络流量进行监控。若攻击者的跳板位于本地网络内, 由于跳板只是起一个攻击数据包转发的功能, 一般不会对攻击数据包进行修改, 因此进出网络的攻击数据包会具有相似性, 通过对进出网络的网络流量进行监控, 检测这种攻击数据包的相似性, 即可判断本地网络内是否存在跳板。若攻击者的主机就位于本地网络内, 则在一定时间内, 攻击者会发出攻击数据包, 而不会有进入本地网络的攻击数据包, 因此也可以通过对进出网络的网络流量进行监控, 如发现只有出的攻击数据包而没有进入的网络数据包, 则可以判断攻击者主机位于本地网络内。

在跳板攻击检测中, 存在一个前提条件, 即进入的攻击流量和转发的流量间隔时间不能很长, 否则很难将入的流量和出的流量关联在一起。这一间隔时间用攻击者的最大容忍时间[48]来表示。如果间隔时间大于最大容忍时间, 例如攻击者向跳板发送命令后, 跳板不是立即转发攻击者的命令, 而是采用计划任务等方式, 让跳板在设定的时间再转发数据, 则在这种情况下现有的方法都将失效。

由于攻击者为了逃避检测, 可以将数据流量进行加密, 因此将情况分为流量未加密情况和加密的情况。

1) 数据包未加密

文献[44]首次提出跳板检测问题, 使用数据包的明文内容的指纹来判断不同的数据包是否具有相同的内容, 从而建立流量间的关联。

2) 数据包加密

由于数据包进行了加密, 无法对数据包的内容进行检查, 因此主要思想是对数据流的特征进行检测, 如数据包的时序特征。

(1) 假设攻击者不会有意识改变数据包的特征

文献[45]首次提出基于数据包时序特征的检测方法, 他们观察到在一个数据流中存在没有数据传输的时间间隔, 将这一时间间隔定义为“关”周期, 而在相似的数据流中, “关”周期的特征是相同的, 因此通过这一特征来关联入的流量和出的流量。但该方法要求连接是同步的。

文献[46]定义了数据传输的平均延迟和最小延迟两个指标来识别数据流的模式, 通过这两个延迟时间计算两个数据流的偏离程度, 如偏离程度小于一定阀值, 则认为两个数据流具有较高的关联度。

文献[47]定义了一个滑动窗口, 计算滑动窗口内数据包之间的间隔时间, 根据数据包间隔时间的特征来进行关联。

文献[89]基于关联规则挖掘算法, 若“入”数据包和“出”数据包的时间差值小于预设的数值, 则将这两个数据包进行关联, 根据流中数据包关联的置信度和支持度。来判断“入”的数据流与“出”的数据流是否具有关联关系。

(2) 假设攻击者有意识改变数据包特征

①攻击者改变数据包的时序特征

文献[48]首次考虑攻击者可能会有意识改变数据包的时序特征, 但假设有一个攻击者的最大延迟容忍时间。它基于小波变换来检测流量的关联性。假设攻击者数据包的到达服从泊松分布或者帕累托分布, 进行了一些理论分析, 但没有给出需要捕获多少数据包才能以一定概率得到正确检测结果的分析。文献[49]在文献[48]的基础上, 去掉了攻击者数据包的概率分布假设, 并给出了需要捕获多少数据包才能以一定概率得到正确检测结果的理论分析。

文献[50]给出了一个基于水印的方法检测流量关联性, 在入流量中填加水印信息, 在出流量中检测是否存在水印信息, 但它假设攻击者数据包之间的时间间隔是独立同分布的。文献[122]同样采用嵌入水印的思想, 通过记录入流量数据包的到达时间, 在一个初始延迟时间的基础上, 再增加或减小一个微小的时间来作为水印信号, 能够获得更好的鲁棒性。

文献[51]假设攻击者随机延迟数据包, 跳板中继的数据包不能丢包、不能乱序、不能增加数据包, 不依赖于数据包大小, 给出了检测方法和理论分析。该方法根据两个约束条件: 出的数据包时间大于入的数据包时间; 出的数据包时间减去入的数据包时间小于最大延迟容忍时间, 将入的数据包关联到出的数据包, 然后假设数据包的顺序不会发生变化, 降低算法的复杂度。文献[52]继续这一思路对保序的方法进行改进和分析。

②攻击者增加额外的数据包

攻击者可以有意识的在中继后的数据流中插入额外的无用数据包, 来破坏输入和输出数据流之间的关联关系。文献[53, 54]对这个问题进行了研究, 假设攻击者在一段时间内能够插入的多余数据包数目是有限的, 通过匹配输入与输出的数据包, 给出了可以抵抗增加多余数据包的跳板检测算法。

③攻击者同时改变时序特征和增加额外数据包

文献[55]假设攻击者在跳板中同时增加数据包的随机延时和多余的数据包, 并假设这两种改变是统计独立的, 基于数据包匹配的方法给出了检测算法。

前述方法都是针对攻击者主机到受害者主机的流量通过跳板时的入流量与出流量的关联分析, 而文献[56]是将攻击者主机到受害者主机的流量通过跳板时的出流量与受害者主机回传给攻击者的流量通过跳板时的出流量进行关联, 定义前者的数据包数为 Send, 后者的数据包数为 Echo, 指出如果两个流量具有相关性, 则(Echo-Send)和(Echo+Send)具有线性关系, 而如果两个流量没有关系, 则不具有这种线性关系。论文通过实验表明, 即使攻击者有意增加数据延时和增加多余数据包, 该方法也能检测出两个流量的相关性。

对于问题 4.1 的解决方案总结如下表所示:

表 4 问题 4.1 的解决方案

Table 4 The solution for the problem 4.1

攻击者能力	检测方法
数据包未加密	根据数据包明文内容指纹进行关联
数据包仅加密, 不改变其特征	根据到达和发出数据包的时序特征进行关联
数据包加密, 仅改变其时序特征	1) 对时序特征进行变换 2) 在数据流中填加水印特征 3) 根据时间约束, 对入数据包和出数据包进行匹配
数据包加密, 仅增加额外数据包	根据数据包大小, 对入数据包和出数据包进行匹配
数据包加密, 不仅改变其时序特征, 且增加额外数据包	1) 数据包匹配 2) 数据流中数据包包数的相关性

上述方法一般都假设攻击者不会有意识的在跳板上丢弃某些数据包, 文献[57]研究了丢包问题, 通过仿真实验指出丢包实际上会严重影响两个数据流的关联性, 但没有给出在这种情况下如何检测跳板的方法。

3.4.2 问题 4.2

若检测人员发现本地网路中存在跳板, 要想识别真正的攻击源, 需要和其上游的网络管理域进行

协作, 如图 9 所示, 按照相同的方法进行检测, 直至发现真正的网络攻击源。

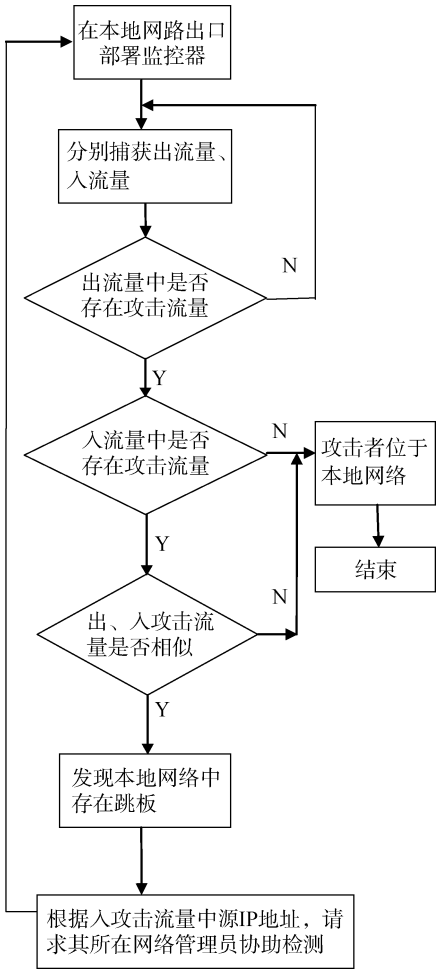

图 9　检测流程

Figure 9　The testing flow

由于这一过程需要人工参与, 文献[58]给出了一个自动协作的框架机制。该方法综合各个网络域中的网络设备, 如防火墙、路由器等, 通过多个网络域的协作来追踪攻击源。

文献[47]指出, 可以对攻击路径上的各个跳板采用计算机取证的方法, 通过各个跳板的日志记录来进行追踪。但这种方法受限较大, 因为攻击者可能已经完全控制了跳板主机, 那么这些跳板主机上的日志记录也可能已被攻击者所修改。

上述方法需要各个网络域的协作, 但如果上游攻击路径上有一个或多个网络管理域无法进行合作, 则很难定位真正的攻击者。根据检测人员所掌握的资源, 可以分为两种情况:

1) 检测人员可以在网络上部署传感器

根据攻击者的数据包是否加密, 可以分为两种情况

(1) 数据包未加密

文献[59]给出了一个主动的方法, 即在回传给攻击者的数据包中注入水印特征, 在攻击路径上部署传感器, 从而追踪到攻击者。

(2) 数据包加密

文献[60,61]对这种情况进行了研究, 所提方法实际上是文献[50]在这种情况下的自然延伸。通过改变回送给攻击者的数据包的时序特征, 也就是以数据包之间的间隔时间为载体来填加水印信息, 从而追踪数据包的流向。

文献[62-64]对数据包的来回时间进行测量。该方法基于这样一个假设, 正常的数据包发送和回复时间相差应该在一个有限的时间内, 而回传数据包经过跳板进行中继, 这个时间必然远高于正常的时间差。这个时间越长, 说明检测点离攻击者主机的距离越远。

2) 检测人员无法在网络上部署传感器

在这种情况下, 攻击源定位问题仍然是一个开放的问题。目前未发现有文献在这一约束条件下进行研究。

但我们一个朴素的想法是, 类似于针对僵尸网络溯源的[42,43], 如果检测人员在攻击者未察觉的情况下能控制跳板主机, 在跳板主机回传给攻击者的消息中插入可执行代码, 当攻击者接收到消息后, 可执行代码在攻击者主机上运行, 把攻击者主机的信息发送给检测人员。这一思路需要根据攻击者所使用的软件工具, 挖掘其漏洞, 难度较大, 且不具有通用性。

对于问题 4.2 的解决方案总结如下表所示:

表 5　问题 4.2 的解决方案

Table 5　The solution for the problem 4.2

网络安全人员所掌握的资源		检测方法
各个网络域能够相互协作		利用问题 4.1 的解决方案, 逐跳进行跳板检测, 直至到达攻击者主机所在的网络
各个网络域不能相互协作	能在网络上部署传感器	1) 在数据包中(针对明文)或数据流中(针对密文)填加水印信息 2) 测量攻击者发送数据包和受害主机应答数据包的时间差
	不能在网络上部署传感器	利用攻击者所使用软件的漏洞或可执行代码, 感染攻击者主机

对跳板攻击更多的研究综述可以参考文献[65]。

另外, 在前文所述方法中, 很多方法要求在网络上部署传感器, 然而在骨干网络上部署传感器是一个费时、费力、高成本的任务, 如何能尽可能少的部署传感器, 同时能得到较好的监控效果, 是一个难题。文献[66]对该问题进行了研究。

3.5 局域网中的攻击溯源问题

由于目前 NAT 技术的大量使用, 若攻击者主机位于 NAT 后面, 使用私网 IP 地址, 对于攻击源的追踪只能到攻击者的 NAT 网关, 而无法穿透 NAT 网关。因此, 假设已知攻击者来自于某个 NAT 网关保护的私有网络, 如何定位攻击者主机在私网中的位置?

这一问题在有线网络比较容易解决, 因为 NAT 网关只进行 IP 地址和端口的转换, 对数据包的内容和大部分头部信息并不进行修改, 即使数据包的内容经过了加密。因此只要对公网的数据流和私网的数据流进行监控, 根据 IP 头部中的信息, 如序列号, 就可以把公网数据流和私网数据流关联起来, 从而知道攻击者的私网 IP 地址和 MAC 地址。

文献[105]研究了经 NAT 地址转换的数据包来源识别问题, 指出 NAT 服务器一般不会更改原 IP 数据包头中的序列号。对于 Windows 系统, 数据流中的包头的序列号通过每次加 1 的方式进行增长, 这可以用于判断来自同一台主机的数据包。而对于 Linux 系统, 由于采用序列号随机化的方式, 前述特征无法用于 Linux 主机, 但可以通过 Http 协议报头中的时戳、cookie 等来判断。

文献[67]基于文献[33]的数据包记录的方法, 提出了一种 2 层网络的攻击源追踪方法, 即在已知离攻击者主机最近的路由器的情况下, 在内网中确定攻击者的主机。该方法通过在路由器上记录数据包的 MAC 地址、来自交换机的哪个端口、来自路由器的哪个端口, 通过建立这些信息的摘要表, 从而能快速识别出攻击者主机所在的子网。随后, 文献[109]认为文献[67]中的方法在实际 2 层交换网络中部署困难, 基于文献[35]中的方法结合交换机中的审计记录, 提出一种新的攻击溯源方法, 实现即使只有一个攻击数据包, 也可以进行追踪。

在无线局域网中, 这一问题就比较困难了, 因为无线路由器不仅要进行 IP 地址的转换, 而且会对 IP 数据包, 包括 IP 头部进行加密, 如 WPA 算法, 这样就无法通过内、外网数据流的观察来进行关联。文献[68]对这一问题进行了研究, 利用数据包的大小在数据流中填加水印, 从而对内、外网数据流进行关联。具体来说, 该方法是针对数据流从外网流入内网的攻击者主机的情况, 在外网中能够控制相关的数据流, 选择一个作为水印的特征码, 然后随机选择数据流中的多个数据包, 用大小为 700 字节的数据包代表码元 0, 1000 字节的数据包代表码元 1, 若选择的数据包超过所代表码元的数据包大小, 则把该数据包按码元大小进行分组; 若小于, 则重新选择下一个数据包。之所以选择这两个数据包字节大小, 是因为经过对 802.11 数据帧进行统计, 具有 500—1000 字节大小的数据帧很少, 可以避免误报。这样在内网中检测数据流中所嵌入的特征码, 则可以将内、外网数据关联起来。

但这一方法不能用于从内网流出到外网的情况, 例如在内网中的攻击者向外发动攻击的时候在外网发现攻击数据流, 需要定位内网的主机位置。目前还未发现有相关文献对这个问题进行研究。

文献[68]能够获得内网中攻击者主机的 IP 地址和 MAC 地址, 但在大型公共场合的无线网络中, 如机场、车站、宾馆, 如何定位攻击者主机的物理位置仍然是一个问题。文献[69]对此问题进行了研究, 假设已知攻击者主机的 MAC 地址, 设计了一套系统来定位目标的物理位置。该系统利用定向天线捕获无线数据帧, 识别出源 MAC 地址为目标 MAC 地址的数据帧, 利用数据信号强度定位信号的来源方向, 通过多个地点的测量, 即可以定位出目标的物理位置。该系统能够对三维空间进行定位, 即使攻击者主机位于高楼中, 测量地点在楼外, 也可以定位出攻击者主机所在的楼层房间。

对问题 5 的解决方法总结如下表所示:

表 6 问题 5 的解决方案

Table 6 The solution for the problem 5

网络安全人员所掌握资源	检测方法
能够控制内、外网数据流	利用数据包特征、数据包标记或数据流水印方法, 识别攻击者主机的 MAC 地址
无线网络, 且不能控制内网数据流	利用定向天线定位攻击者主机的物理位置

4 溯源方法分类

通过对上述各种场景下的攻击溯源方法的分析, 可以看到, 所有的方法大致基于以下 4 种思想:

1) 在数据包中打标记

这种方法需要修改现有的网络协议, 在协议数据包中添加标记, 通过对数据包中标记的追踪, 识别出数据包的传播路径, 从而追踪到数据包的源头。

目前大多数网络协议都已经标准化, 修改现有协议显然不是一件容易实施的工程, 需要考虑成本、性能、兼容等各种工程问题, 但在技术上难度较小, 准确性较高。

2) 在数据流中加入流水印

这种方法不需要修改网络协议, 即使对于加密的数据包也可以适用, 而且不依赖于单个的数据包, 而是整个数据流, 即使流中出现重打包、丢包等现象, 也能以一定的准确率检测到水印信息。但同时, 网络时延、数据包的转发时延, 以及攻击者有意识的破坏水印信息, 都可能影响流水印的检测准确率。

这种方法仍然要求在较大范围网络中部署检测传感器, 以检测可能出现的水印信息, 这需要网络基础设施的支持, 仍然不是一项容易实施的工程。在技术上要保证水印检测的准确率, 有一定难度。

3) 日志记录的方法

这种方法可以是对网络数据包的日志记录, 也可以是对主机的日志记录, 通过日志记录查找曾经进行攻击数据包传播的痕迹, 从而重构出攻击路径。

这种方法不影响现有网络的功能, 可以以补丁的形式增加额外日志记录的功能, 以记录必要的信息, 在工程上实施难度不大, 但这种方法属于事后的审计, 可能会丢失一些关键的信息, 而且日志记录也可能会被攻击者修改。

4) 渗透测试的方法

这种方法实质是用攻击方法对抗攻击, 即寻找攻击者所使用系统、网络的安全漏洞, 利用这些漏洞编写相应的可执行代码, 使得攻击者感染这些代码, 由这些可执行代码向取证人员主动报告攻击者的信息。

这种方法在工程上容易实施, 甚至单个取证人员就可以执行, 但在技术上难度较大, 需要挖掘各种漏洞。而且由于这是一种通过攻击获取证据的方法, 在司法上的可信性方面存在疑问。

这 4 类方法的比较见表 7。

由上述分析可以看出, 这 4 类思想方法各有优势和不足, 需要根据具体情况并综合考虑各种因素来选择合适的攻击溯源方法。并且, 在实际的攻击中,

表 7 4 类方法的对比

Table 7 The comparison among 4 kinds of methods

方法	技术难度	实施难度	证据可信性
包标记	易	难	高
流水印	较易	较难	较高
日志记录	较难	较易	较低
渗透测试	难	易	低

可能存在更复杂的场景, 如图 10 所示。攻击者主机可能位于内网中, 可能在网关处经过了 NAT 转换, 这是对攻击源进行隐藏的第一个措施。攻击者可能控制了 n 个中间跳板, 来发动网络攻击, 这是对攻击源进行隐藏的第二个措施。攻击者可能控制了僵尸网络, 这是对攻击源进行隐藏的第三个措施。在这种情况下, 需要综合运用各种攻击溯源方法[93]。

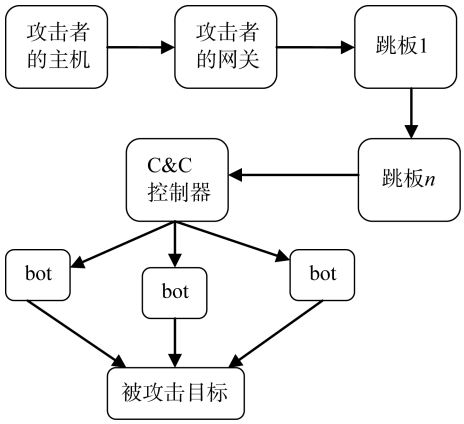

图 10 实际的网络攻击场景

Figure 10 The real network attack scene

由此可见, 在大多数场景下直接追踪到真正的网络攻击源几乎是不可能的, 需要先将困难的问题进行分解, 逐步迭代进行追踪, 才能得出最后的结果。如, 先追踪虚假 IP 地址数据包的真正地址; 再以此 IP 地址为起点, 或追踪僵尸网络中的 C&C 服务器、或溯源匿名网络、或溯源跳板链, 追踪到攻击者所在网络的 NAT 网关; 再以 NAT 网关为起点, 追踪内网中的真正攻击者。

5 总结及展望

虽然司法取证人员可以通过法官授权, 对嫌疑主机或网络进行必要的司法取证, 但对于攻击溯源问题, 由于攻击路径可能分布在整个互联网上, 可能涉及到跨司法管辖权的问题, 因此司法取证人员不一定能掌握实现攻击溯源所需的全部资源。从目前的溯源方法来看, 取证人员需要掌握的资源越少, 则实施起来越容易, 但技术难度越大; 反之, 实施起

来越困难, 但技术难度越小。目前, 还无法做到自动化的对攻击进行追踪, 更多的是依赖取证人员手工的分析。

通过对现有网络攻击源追踪技术的研究可以发现, 大部分的方法依赖于网络基础设施的帮助, 否则几乎不可能发现真正的网络攻击者的主机, 这就给下一代互联网基础设施的设计带来一个挑战, 即是否要支持网络攻击源追踪技术, 并在网络安全结构或协议中有所考虑。虽然对于下一代互联网, 是采用重新顶层设计还是自然演进的路线, 还存在很大的争议[70], 但已有的研究表明现有的网络基础设施不足以支撑有效的网络攻击源追踪技术, 需要作出改变。在下一代互联网体系结构中是否需要嵌入网络攻击源追踪机制, 以及如何嵌入这种机制, 是需要慎重考虑的问题。

由于目前攻击溯源方法的实用性较差, 实际效果不理想, 随着网络安全领域威胁情报(Threat Intelligence)研究的兴起, 利用共享的威胁情报来进行攻击溯源成为可能[123]。威胁情报能够为攻击溯源提供更多的数据支撑, 并且利用威胁情报有可能追踪到实施攻击的真正的个人或组织, 这将是下一步研究的方向。

参考文献

[1] Stefan Savage, David Wetherall, Anna Karlin, Tom Anderson, Practical network support for IP traceback, *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, 2000, pp.295-306

[2] Thomas W.Doeppner, Philip N.Klein, Andrew Koyfman, Using router stamping to identify the source of IP packets, *The 7th ACM Conference on Computer and Communications Security*, 2000, pp.184-189

[3] Stefan Savage, David Wetherall, Anna Karlin, Tom Anderson, Network support for IP traceback, *IEEE/ACM Transactions on networking*, 2001, 9(3), pp.226-237

[4] Kihong Park, Heejo Lee, On the effectiveness of probabilistic packet marking for IP traceback under denial of service attack, *INFOCOM*, 2001, pp.338-347

[5] Li De-quan, Su Pu-rui, Feng Deng-guo, Notes on packet marking for IP traceback, *Journal of Software*, 2004, 15(2), pp.250-258

[6] Dawn Xiaodong Song, Adrian Perrig, Advanced and authenticated marking schemes for IP traceback, *IEEE INFOCOM*, 2001, pp.878-886

[7] Drew Dean, Matt Franklin, Adam Stubblefield, An algebraic approach to IP traceback, *ACM Transactions on Information and System Security*, 2002, 5(2), pp.119-137

[8] Chao Gong, Kamil Sarac, Toward a practical packet marking approach for IP traceback, *International Journal of Network Security*, 2009, 8(3) pp.271-281

[9] Micah Adler, Trade-offs in probabilistic packet marking for IP traceback, *Journal of the ACM*, 2005, 52(2), pp.217-244

[10] Tao Peng, Christopher Leckie, Kotagiri Ramamohanarao, Adjusted Probabilistic packet marking for IP traceback, *NETWORKING* 2002, LNCS 2345, pp.697-708

[11] Byungryong Kim, Efficient technique for fast IP traceback, *Cooperative Design, Visualization, and Engineering*, 2006, LNCS 4101, pp.211-218

[12] Vamsi Paruchuri, Arjan Durresi, Sriram Chellappan, TTL based packet marking for IP traceback, *IEEE Global Telecommunications Conference*, 2008, pp. 1-5

[13] Jenshiuh Liu, Zhi-Jian Lee, Yeh-Ching Chung, Dynamic probabilistic packet marking for efficient IP traceback, *Computer Networks*, 2007, 51(3), pp.866-882

[14] Hongcheng Tian, Jun Bi, Xiaohe Jiang, An adaptive probabilistic marking scheme for fast and secure traceback, *Networking Science*, 2013, 2(1-2) ,pp.42-51

[15] Abraham Yaar, Adrian Perrig, Dawn Song, FIT: fast internet traceback, *IEEE INFOCOM*, 2005, pp.1395-1406

[16] Michael T.Goodrich, Efficient packet marking for large-scale IP traceback, *The 9th ACM Conference on Computer and Communications Security*, 2002, pp.117-126

[17] K.T.Law, John C.S.Lui, David K.Y.Yau, You can run, but you can't hide: an effective methodology to traceback DDoS attackers, *IEEE Transactions on Parallel and Distributed Systems*, 2005, 16(9), pp.799-813

[18] Yang Xiang, Wanlei Zhou, Minyi Guo, Flexible deterministic packet marking: an IP traceback system to find the real source of attacks, *IEEE Transactions on Parallel and Distributed Systems*, 2009, 20(4), pp.567-580

[19] Hyungseok Kim, Eunjin Kim, Seungmo Kang, Huy Kang Kim, Network forensic evidence generation and verification scheme, *Telecommunication Systems*, 2015

[20] Andrey Belenky, Nirwan Ansari, IP traceback with deterministic packet marking, *IEEE Communications Letters*, 2003, 7(4), pp.162-164

[21] Andrey Belenky, Nirwan Ansari, On deterministic packet marking, *Computer Networks*, 2007, 51(10), pp.2677-2700

[22] M.Vijayalakshmi, N.Nithya, S.Mercy Shalinie, A novel algorithm on IP traceback to find the real source of spoofed IP packets, *Artificial Intelligence and Evolutionary Algorithms in Engineering Systems*, 2015, LNCS 325, pp.79-87

[23] Steve Bellovin, Marcus Leech, Tom Taylor, ICMP traceback message, *draft-ietf-itrace-04*, 2003

[24] Allison Mankin, Dan Massey, Chien-Lung Wu, S.Felix Wu, Lixia Zhang, On design and evaluation of “intention-driven” ICMP traceback, *the 10th International Conference on Computer Communications and Networks*, 2001, pp.159-165

[25] Henry C.J.Lee, Vrizlynn L.L.Thing, Yi Xu, Miao Ma, ICMP traceback with cumulative path, an efficient solution for IP traceback, *Information and Communications Security*, 2003, LNCS 2836, pp.124-135

[26] Basheer Al-Duwairi, Manimaran Govindarasu, Novel hybrid schemes employing packet marking and logging for IP traceback, *IEEE Transactions on Parallel and Distributed Systems*, 2006, 17(5), pp.403-418

[27] Chao Gong, Kamil Sarac, IP traceback based on packet marking and logging, *IEEE International Conference on Communications*, 2005, pp.1043-1047

[28] Ming-Hour Yang, Ming-Chien Yang, RIHT: a novel hybrid IP traceback scheme, *IEEE Transactions on Information Forensics and Security*, 2012, 7(2), pp.789-797

[29] Ning Lu, Yulong Wang, Sen Su, Fangchun Yang, A novel path-based approach for single-packet IP traceback, *Security and Communication Networks*, 2014, 7(2), pp.309-321

[30] Fan Min, Jun-yan Zhang, Guo-wie Yang, An IP traceback scheme integrating DPM and PPM, *ACNS* 2003, LNCS 2846, pp.76-85

[31] Tatsuya Baha, Shigeyuki Matsuda, Tracing network attacks to their sources, *IEEE Internet Computing*, 2002, 6(2), pp.20-26

[32] Alex C.Snoeren, Craig Partridge, Luis A.Sanchez, Christine E.Jones, Fabrice Tchakountio, Beverly Schwartz, Stephen T.Kent, W.Timothy strayer, Single-packet IP traceback, *IEEE/ACM Transactions on Networking*, 2002, 10(6), pp.721-734

[33] Alex C.Snoeren, Craig Partridge, Luis A.Sanchez, Christine E.Jones, Fabrice Tchakountio, Stephen T.Kent, W.Timothy Strayer, Hash-based IP traceback, *Proceedings of ACM SIGCOMM*, 2001

[34] Jun Li, Minho Sung, Jun Xu, Li Li, Large-scale IP traceback in high-speed internet: practical techniques and theoretical foundation, *IEEE Symposium on Security and Privacy*, 2004, pp.115-129

[35] Chao Gong, Trinh Le, Turgay Korkmaz, Kamil Sarac, Single packet IP traceback in AS-level partial deployment scenario, *IEEE Global Telecommunications Conference*, 2005

[36] Robert Stone, Centertrack: an IP overlay network for tracking DoS floods, *Proceedings of the 9th Conference on USENIX Security Symposium*, 2000, pp.15-28

[37] Guang Yao, Jun Bi, Athanasios V.Vasilakos, Passive IP traceback: disclosing the locations of IP spoofers from path backscatter, *IEEE Transactions on Information Forensics and Security*, 2015, 10(3), pp.471-484

[38] Hal Burch, Bill Cheswick, Tracing anonymous packets to their approximate source, *Proceedings of the 14th Systems Administration Conference*, 2000, pp.319-327

[39] Gu Hsin Lai, Chia-Mei Chen, Bing-Chiang Jeng, Willams Chao, Ant-based IP traceback, *Expert Systems with Applications*, 2008, 34, pp.3071-3080

[40] Zaiyao Yi, Liuqing Pan, Xinmei Wang, Chen Huang, Benxiong Huang, IP traceback using digital watermark and honeypot, *UIC* 2008, LNCS 5061, pp.426-438

[41] Daniel Ramsbrock, Xinyuan Wang, Xuxian Jiang, A first step towards live botmaster traceback, *RAID* 2008, LNCS 5230, pp.59-77

[42] Wenjie Lin, David Lee, Traceback attacks in cloud - pebbletrace botnet, *The 32nd International Conference on Distributed Computing Systems Workshops*, 2012, pp.417-426

[43] Fang Yu, David Lee, Internet attack traceback cross-validation and pebble tracing, *IEEE Conference on Technologies for Homeland Security*, 2008, pp.378-383

[44] S.Staniford-Chen, L.Heberlein, Holding intruders accountable on the internet, *IEEE Symposium on Security and Privacy*, 1995, pp.39-39

[45] Y.Zhang, V.Paxson, Detecing stepping stones, *the 9th USENIX Security Symposium*, 2000, pp.171-184

[46] K.Yoda, H.Etoh, Finding a connection chain for tracing intruders, *The 6th European Symposium on Research in Computer Security*, 2000, LNCS 1895, pp.191-205

[47] X.Wang, D.Reeves, S.Wu, Inter-packet delay-based correlation for tracing encrypted connections through stepping stone, *The 7th European Symposium on Research in Computer Security*, 2002, LNCS 2502, pp.244-263

[48] David L.Donoho, Ana Geogfina Flesia, Umesh Shankar, Vern Paxson, Jason Coit, Stuart Staniford, Multiscale stepping-stone detection: detecting pairs of jittered interactive streams by exploiting maximum tolerable delay, *The Fifth International Symposium on Recent Advances in Intrusion Detection*, 2002, LNCS 2516

[49] Avrim Blum, Dawn Song, Shobha Venkataraman, Detection of Interactive stepping stones: algorithms and configence bounds, *Recent Advances in Intrusion Detection*, 2004, LNCS 3224, pp.258-277

[50] Xinyuan Wang, Douglas S.Reeves, Robust correlation of encrypted attack traffic through stepping stones by manipulation of inter-packet delays, *Proceedings of the 2003 ACM Conference on Computer and Communications Security*, 2003, pp.20-29

[51] Ting He, Lang Tong, A signal processing perspective to stepping-stone detection, *The 40th Annual Conference on Information Sciences and Systems*, 2006, pp.687-692

[52] Ying-Wei Kuo, Shou-Hsuan Stephen Huang, Stepping-stone detection algorithm based on order preserving mapping, *International Confer-*

ence on Parallel and Distributed Systems, 2007, pp.1-8

[53] Han-Ching Wu, Shou-Hsuan Stephen Huang, Detecting stepping-stone with chaff perturbations, *The 21st International Conference on Advanced Information Networking and Applications Workshops*, 2007, pp.85-90

[54] Baris Coskun, Nasir Memon, Online sketching of network flows for real-time stepping-stone detection, *2009 Annual Computer Security Applications Conference*, 2009, pp.473-483

[55] Linfeng Zhang, Anthony G.Persaud, Alan Johnson, Yong Guan, Detection of stepping stone attack under delay and chaff perturbations, *The 25th IEEE International Performance, Computing, and Communications Conference*, 2006, pp. 247-256

[56] Shou-Hsuan Stephen Huang, Robert Lychev, Jianhua Yang, Stepping-stone detection via request-response traffic analysis, *Autonomic and Trusted Computing*, 2007, LNCS 4610, pp.276-285

[57] Mohd Nizam Omar, Lebyzar Siregar, Rahmat Budiarto, Dropped packet problems in stepping stone detection method, *International Journal of Computer Science and Network Security*, 2008, 8(2), pp.109-115

[58] Dan Schnackenberg, Harley Holliday, Randall Smith, Kelly Djahandari, Dan Sterne, Cooperative intrusion traceback and response architecture (CITRA), *Proceedings of the DARPA Information Survivability Conference and Exposition*, 2001, pp.56-68

[59] X.Wang, D.Reeves, S.Wu, J.Yuill, Sleepy watermark tracing: an active network-based intrusion response framework, *The 16th International Information Security Conference*, 2001, pp.369-384

[60] Xinyuan Wang, Douglas S.Reeves, Robust correlation of encrypted attack traffic through stepping stones by flow watermarking, *IEEE Transactions on Dependable and Secure Computing*, 2011, 8(3), pp.434-449

[61] Khyamling A.Parane, G.R.Udupi, Manoj A.Patil, Correlation of encrypted attack traffic through network flow watermarking, *Proceedings of International Conference on Emerging Research in Computing, Information, Communication and Applications*, 2013, pp.245-250

[62] Ping Li, Wanlei Zhou, Yini Wang, Getting the real-time precise round-trip time for stepping stone detection, *The 4th International Conference on Network and System Security*, 2010, pp.377-382

[63] Jianhua Yang, Shou-Hsuan Stephen Huang, Matching TCP/IP packets to detect stepping-stone intrusion, *International Journal of Computer Science and Network Security*, 2006, 6(10), pp.269-277

[64] Kwong H.Yung, Detecting long connection chains of interactive terminal sessions, *Proceedings of International Symposium on Recent Advance in Intrusion Detection*, 2002, LNCS 2516, pp.1-16

[65] Robert Shullich, Jie Chu, Ping Ji, Weifeng Chen, A survey of research in stepping-stone detection, *International Journal of Electronic Commerce Studies*, 2011, 2(2) pp.103-126

[66] Young June Pyun, Douglas S.Reeves, Strategic deployment of network monitors for attack attribution, *The 4th International Conference on Broadband Communications, Networks and Systems*, 2007, pp.525-534

[67] Hiroaki Hazeyama, Masafumi Oe, Youki Kadobayashi, A layer-2 extension to hash-based IP traceback, *IEICE Transactions Information and Systems*, 2003, E86-D(11), pp.2325-2333

[68] Yinjie Chen, Zhongli Liu, Benyuan Liu, Xinwen Fu, Wei Zhao, Identifying mobiles hiding behind wireless routers, *IEEE INFOCOM* 2011, pp.2651-2659

[69] Jizhi Wang, Yingjie Chen, Xinwen Fu, Jie Wang, Wei Yu, Nan Zhang, 3DLoc: three dimensional wireless localization toolkit, *IEEE ICDCS* 2010

[70] Jennifer Rexford, Constantine Dovrolis, Future Internet architecture: clean-slate versus evolutionary research, *Communications of The ACM*, 2010, 53(9), pp.36-40

[71] Chen Zhouguo, Shi Pu, Shixiong Zhu, Traceback technology for anonymous network, *Journal of Computer Research & Development*, 2012, 2012, 49(Suppl.)pp.111-117
(陈周国、蒲石、祝世雄, 匿名网络追踪溯源综述, *计算机研究与发展*, 2012, 49(Suppl.): 111-117)

[72] Xiaogang Wang, Ming Yang, Junzhou Luo, A novel sequential watermark detection model for efficient traceback of secret network attack flows, *Journal of Network and Computer Applications*, 2013, 36, pp.1660-1670

[73] Bjorn Stelte, ISP traceback - attack path detection, *IEEE Conference on Communications and Network Security*, 2013, pp.363-364

[74] You-ye Sun, Cui Zhang, Shao-qing Meng, Kai-ning Lu, Modified deterministic packet marking for DDoS attack traceback in IPv6 Network, *The 11th IEEE International Conference on Computer and Information Technology*, 2011, pp.245-248

[75] Guanyu Tian, Zhenhai Duan, Todd Baumeister, Yingfei Dong, A traceback attack on Freenet, *The Proceedings of IEEE INFOCOM*, 2013, pp.1797-1805

[76] Gaofeng He, Ming Yang, Junzhou Luo, Lu Zhang, Yuanyuan Ma, Modeling and analysis of time characteristics used in onion routing traceback techniques, *Chinese Journal of Computers*, 2014, 37(2), pp.356-372
(何高峰、杨明、罗军舟、张璐、马媛媛, 洋葱路由追踪技术中时间特征的建模与分析, *计算机学报*, 2014, 37(2): 356-372)

[77] Wang yu, Li yichao, Zhang xiaoshong, Zeng jiazhi, A method of IP traceback for DoS, *The Proceedings of the 4th International Conference on Parallel and Distributed Computing, Applications and Technologies*, 2003, pp.762-764

[78] Guang Yao, Jun Bi, Zijian Zhou, Passive IP traceback: capturing the origin of anonymous traffic through network telescopes,

SIGCOMM, 2010, pp.413-414

[79] Liancheng Zhang, Zhenxing Wang, Jing Xu, Qian Wang, Multi-flow attack resistant interval-based watermarks for tracing multiple network flows, *ICCIC* 2011, pp.166-173

[80] Zhenqiang Wu, Bo Yang, An advanced marking scheme and realization for onion routing traceback, *Journal of China Institute of Communications*, 2002, 23(5), pp. 96-102
(吴振强, 杨波, 追踪洋葱包的高级标记方案与实现, *通信学报*, 2002, 23(5): 96-102)

[81] Xian Pan, Junwei Huang, Zhen Ling, Xinwen Fu, Long PN code based traceback in wireless networks, *International Journal of Performability Engineering*, 2012, 8(2), pp.173-191

[82] Zhen Ling, Junzhou Luo, Kui Wu, Wei Yu, Xinwen Fu, TorWard: discovery, blocking, and traceback of malicious traffic over Tor, *IEEE Transactions on Information Forensics and Security*, 2015, 10(12), pp.2525-2530

[83] Zhiqiang Gao, Nirwan Ansari, Tracing cyber attacks from the practical perspective, *IEEE Communications Magazine*, 2005, 5, pp.123-131

[84] Lu Zhang, Junzhou Luo, Ming Yang, Gaofeng He, Interval centroid based flow watermarking technique for anonymous communication traceback, *Journal of Software*, 2011, 22(10), pp.2358-231
(张璐、罗军舟、杨明、何高峰, 基于时隙质心流水印的匿名通信追踪技术, *软件学报*, 2011, 22(10): 2358-231)

[85] Masafumi OE, Youki Kadobayashi, Suguru Yamaguchi, An implementation of a hierarchical IP traceback architecture, *International Symposium on Applications and the Internet Workshops*, 2003, pp.250-253

[86] Young Hwan Lim, Dong Hwi Lee, Won Hyung Park, Kwang Ho Kook, Detection and traceback of illegal users based on anonymous network in bittorrent environment, *Wireless Pers Commu*., 2013, 73, pp.319-328

[87] Ahmad Fadlallah, Ahmed Serhrouchni, PSAT: proactive signaling architecture for IP traceback, *Proceedings of the 4th Annual Communication Networks and Services Research Conference*, 2006, pp.299-305

[88] Zongbin Liu, Jiwu Jing, Peng Liu, Rate-based watermark traceback: a new approach, *ISPEC* 2010, LNCS 6047, 172-186

[89] Ahmad Almulhem, Issa Traore, Mining and detecting connection-chains in network traffic, *IFIP International Federation for Information Processing*, 2007, pp. 47-57

[90] Mouna Gassara, Faouzi Zarai, Ikbel Daly, Mohammad S. Obaidat, Kuei-Fang Hsiao, A new scheme for proactive out of band signaling solution for IP traceback in wireless mesh network, *International Conference on Computer, Information and Telecommunication Systems*, 2015, pp.1-6

[91] Yinan Jing, Jingtao Li, Gendu Zhang, An adaptive edge marking based hierarchical IP traceback system, *ICCNMC* 2005, LNCS 3619, pp. 1188-1197

[92] Hiroaki Hazeyama, Youki Kadobayashi, Masafumi Oe, Ryo Kaizaki, InterTrack: a federation of IP traceback systems across borders of network operation domains, *Proceedings of Annual Computer Security Application Conference*, 2005

[93] W.Timothy Strayer, Christine E. Jones, Beverly I.Schwartz, Joanne Mikkelson, Carl Livadas, Architecture for multi-stage network attack traceback, *The Proceedings of the the 30th Anniversary IEEE Conference on Local Computer Networks*, 2005, pp.785-792

[94] Hiroaki Hazeyama, Youki Kadobayashi, Daisuke Miyamoto, Masafumi Oe, An autonomous architecture for inter-domain traceback across the borders of network operation, *Proceedings of the 11th IEEE Symposium on Computers and Communications*, 2006, pp.38-385

[95] Xiapu Luo, Junjie Zhang, Roberto Perdisci, Wenke Lee, On the secrecy of spread-spectrum flow watermarks, *ESORICS* 2010, LNCS 6345, pp.232-248

[96] Hiroaki Hazeyama, Yoshihide Matsumoto, Youki kadobayashi, Message forwarding strategies for inter-AS packet traceback network, *The Proceedings of the 2nd Joint Workshop on Information Security*, 2007

[97] Micah Adler, Tradeoffs in probabilistic packet marking for IP traceback, *STOC* 2002, pp.407-418

[98] Xiaogang Wang, Junzhou Luo, Ming Yang, A double interval centroid-based watermark for network flow traceback, *The Proceedings of the 14th International Conference on Computer Supported Cooperative Work in Design*, 2010, pp. 146-151

[99] Bo-Chao Cheng, Guo-Tan Liao, Ching-Kai Lin, Shih-Chun Hsu, Ping-Hai Hsu, Jong Hyuk Park, MIB-ITrace-CP: an improvement of ICMP-based traceback efficiency in network forensic analysis, *IFIP International Federation for Information Processing*, 2012, pp.101-109

[100] Omer Demir, Ping Ji, Jinwoo Kim, Session based logging (SBL) for IP-traceback on network forensics, *International Conference on Security & Management*, 2006, pp.233-239

[101] Guangwu Hu, Jianping Wu, Ke Xu, Wenlong Chen, SAVT: a practical scheme for source address validation and traceback in campus network, *The Proceedings of 20th International Conference on Computer Communications and Networks*, 2011, pp. 1-8

[102] Pegah Sattari, Athina Markopoulou, Algebraic traceback meets network coding, *International Symposium on Network Coding,* 2011, pp. 1-7

[103] Wei-Tsung Su, Tzu-Chieh Lin, Chun-Yi Wu, Jang-Pong Hsu, Yau-Hwang Kuo, An on-line DDoS attack traceback and mitiga-

tion system based on network performance monitoring, *Proceedings of 10th International Conference on Advanced Communication Technology*, 2008, pp.1467-1472

[104] Wei-Tsung Su, Yi-Hsun Chuang, Zong-Bing Wu, Yau-Hwang Kuo, A table-driven approach for IP traceback based on network statistic analysis, *ICACT* 2009, pp.1633-1637

[105] M.I.Cohen, Source attribution for network address translated forensic captures, *Digital Investigation*, 2009, 5, pp.138-145

[106] Emmanuel S.Pilli, R.C.Joshi, Rajdeep Niyogi, An IP traceback model for network forensics, *ICEF2C* 2010, pp.129-136

[107] Xiaoyong Li, Dongxi Liu, Dawu Gu, Caiying Bai, Research on intrusion traceback in controlled network environment, *Journal of Computer Research & Development*, 2003, 40(6), pp.808-812
(李小勇、刘东喜、谷大武、白彩英, 可控网络攻击源追踪技术研究, *计算机研究与发展*, 2003, 40(6): 808-812)

[108] Keisuke Takemori, Masahiko Fujinaga, Toshiya Sayama, Masakatsu Nishigaki, Host-based traceback, tracking bot and C&C server, *International Conference on Ubiquitous Information Managements & Communications*, 2009, pp.400-405

[109] Marios S.Andreou, Aad van Moorsel, IP traceback in a switched ethernet network, *Technical Report Series No.CS-TR-1040*, University of Newcastle upon Tyne, 2007

[110] Andre Castelucio, Artur Ziviani, Ronaldo M.Salles, An AS-level overlay network for IP traceback, *IEEE Network*, 2009, 23(1), pp.36-41

[111] Pegah Sattari, Minas Gjoka, Athina Markopoulou, A network coding approach to IP traceback, *IEEE International Symposium on Network Coding*, 2010, pp.1-6

[112] Ryan Pries, Wei Yu, Xinwen Fu, Wei Zhao, A new replay attack against anonymous communication networks, *IEEE International Conference on Communications*, 2008, pp.1578-1582

[113] Xinyuan Wang, Shiping Chen, Sushil Jajodia, Network flow wartermarking attack on low-latency anonymous communication systems, *Proceedings of the IEEE Security & Privacy Symposium*, 2007, pp.116-130

[114] Wei Yu, Xinwen Fu, Steve Graham, Dong Xuan, Wei Zhao, DSSS-based flow marking technique for invisible traceback, *Proceedings of the IEEE Security & Privacy Symposium*, 2007, pp.18-32

[115] R.Ieong, P.Lai, K.P.Chow, M.Kwan, F.Law, Is it an initial seeder? derived rules that indicate a seeder is within the slow-rising period, *The 6th IFIP WG 11.9 International Conference on Digital Forensics*, 2010

[116] Seiichiro Mizoguchi, Keisuke Takemori, Yutaka Miyake, Yoshiaki Hori, Kouichi Sakurai, Traceback framework against botmaster by sharing network communication pattern information, *The 5th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, 2011, pp.639-644

[117] Xiaolei Wang, Yuexiang Yang, Jie He, A collaborative traceback against P2P botnet using information sharing and correlation analysis, *International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, 2014, pp.132-138

[118] Zhen Ling, Junzhou Luo, Wei Yu, Xinwen Fu, Dong Xuan, Weijia Jia, A new cell counter based attack against Tor, *In Proceedings of the 16th ACM Conference on Computer and Communications Security*, 2009, pp.578-589

[119] Zhen Ling, Junzhou Luo, Wei Yu, Xinwen Fu, Equal-sized cells mean equal-sized packets in Tor? *In Proceedings of IEEE International Conference on Communication and Information System Security Symposium*, 2011

[120] Zhen Ling, Xinwen Fu, Weijia Jia, Wei Yu, Dong Xuan, A novel packet size based covert channel attack against anonymizer, *IEEE INFOCOM*, 2011, pp.186-190

[121] Zhen Ling, Xinwen Fu, Weijia Jia, Wei Yu, Dong Xuan, Junzhou Luo, Novel packet size based covert channel attacks against anonymizer, *IEEE Transactions on Computers*, 2013, 62(12), pp. 2411-2426

[122] Amir Houmansadr, Negar Kiyavash, Nikita Borisov, RAINBOW: a robust and invisible non-blind watermark for network flows, *In Proceedings of Network and Distributed System Security Symposium*, 2009

[123] Yang Zeming, Li Qiang, Liu Junrong, Liu Baoxu, Research of threat intelligence sharing and using for cyber attack attribution, *Journal of Information Security Research*, 2015, 1(1), pp.31-36
(杨泽明, 李强, 刘俊荣, 刘宝旭, 面向攻击溯源的威胁情报共享利用研究, 信息安全研究, 2015, 1(1): 31-36)

姜建国 2003 年在四川大学信息安全与应用密码专业获博士学位, 现任中国科学院信息工程研究所研究员, 研究领域为信息安全与保密技术。Email: jiangjianguo@iie.ac.cn

王继志 2003 年在山东科技大学控制理论与控制工程获硕士学位, 现在中国科学院信息工程研究所攻读博士学位, 现任山东省计算中心(国家超级计算济南中心)副研究员, 研究领域为网络安全、安全协议。Email: wangjzh@sdas.org

孔斌 2002 年在华中科技大学计算机学院获得硕士学位。现在北京交通大学攻读博士学位, 现任国家保密科技测评中心高级工程师。研究领域为计算机网络安全。研究兴趣包括: 网络安全标准研究、计算机检查与网络风险评估技术、云计算。Email: pingpangfan@163.com

刘吉强 1999 年在北京师范大学获理学博士, 现任北京交通大学教授, 研究领域为可信计算、应用密码学、安全协议、隐私保护。Email: jqliu@bjtu.edu.cn

胡波 2007 年在北京工业大学电子信息与控制工程学院获得硕士学位。现任中国科学院信息工程研究所高级工程师。研究领域为计算机网络安全。研究兴趣包括: 云计算、虚拟化安全、安全风险评估等。Email: hubo@iie.ac.com