文章编号：2095-2163(2023)01-0055-05 中图分类号：TP393 文献标志码：A

一种基于常见性能计算机环境的快速探测方法

赵远方，张 睿，白 玮
（陆军工程大学 指挥控制工程学院，南京 210014）

摘 要：当前的互联网探测技术大都依赖于单节点性能和单个大带宽网络出口，这实际上限制了互联网探测技术的应用。本文提出一种基于常见性能计算机环境的网络状态快速探测方法，利用分布式探测架构和多级扫描策略使得互联网探测不再受限于单节点网络性能和网络带宽，在保证扫描准确率不低于现有方法的同时，性能消耗极大降低。实验证明，其分布式探测架构在保证快速全网扫描的基础上，大大降低了网络占用率及 CPU 占用率。

关键词：互联网探测；分布式探测架构；扫描策略

A fast network state detection method based on common performance computer environment

ZHAO Yuanfang, ZHANG Rui, BAI Wei
(College of Command and Control Engineering, Army Engineering University of PLA, Nanjing 210014, China)

【Abstract】 Most current Internet probing techniques rely on single-node performance and a single large bandwidth network outlet, which actually limits the application of Internet probing techniques. This paper proposes a fast network state detection method based on common performance computer environments, using a distributed detection architecture and multi-level scanning strategy makes Internet detection is no longer limited by single-node network performance and network bandwidth, while ensuring that the scan accuracy is not lower than existing methods, while greatly reducing performance consumption. Experiments prove that its distributed detection architecture greatly reduces the network occupancy and CPU occupancy on the basis of ensuring fast network-wide scanning.

【Key words】 internet detection; distributed detection architecture; scanning strategy

0 引 言

随着信息社会的不断发展，互联网逐渐在国家经济发展、国防军事、科学研究等领域扮演着愈发重要的角色。特别是随着信息时代的到来，网络空间已经成为“第五空间”，据统计 2020 年超过 250 亿台设备接入互联网。网络资产探测技术也随着互联网规模的变化不断更新迭代，为了能够快速探测互联网资产存活性，获取其服务内容及脆弱性信息，需要对网络资产进行快速探测且不断更新，这也是网络态势感知的基础工作。

当前的互联网探测技术大都依赖于单节点性能和单个大带宽网络出口，这实际上限制了互联网探测技术的应用，本文提出一种基于常见性能计算机环境的网络状态快速探测方法，使得互联网探测不再受限于单节点网络性能和网络带宽，在保证扫描准确率不低于现有方法的同时，性能消耗极大降低。

1 相关工作

1.1 主动探测

主动探测使用基于服务指纹的主动探测技术，其探测内容包括：存活性探测、服务内容探测、目标脆弱性探测和资产拓扑探测等，是网络资产探测的主要方法。

Nmap 是比较早期的主动探测工具，提供端口扫描、主机发现、操作系统识别等功能[1]。半开放扫描(Half-open scanning)是其经典的端口扫描模式，通过发送 SYN 数据包到目标端口，根据应答数据包的返回内容判断端口状态[2]；DNmap 引入分布式的 Nmap 扫描框架实现了大型网络探测能力[3]；ZMap 是美国密歇根大学研究者开发的一款开源探测工具，能在一个小时内扫描整个公共互联网，显示

作者简介：赵远方(1990-)，女，硕士研究生，主要研究方向：网络安全；张 睿(1977-)，男，博士，教授，主要研究方向：数据工程。
通讯作者：张 睿 Email：269973792@qq.com

收稿日期：2022-04-06

近 40 亿在线设备的信息[4]；Masscan 类似于 Nmap，是全网级别的端口扫描工具，速度更快但是在精确度上不如 Nmap[5]。Masscan 通过异步传输的方式进行，针对 TCP 端口进行扫描，同样使用 SYN 扫描的方式，不建立一个完全的 TCP 连接，允许自定义任意的地址范围和端口范围，对于非连续段 IP 的扫描更加高效，可以有效降低目标网络负载压力。Shamai 等[6]提出了单包随机理论，对 SYN/ACK 数据包的超时时间进行分析，建立了 116 个操作系统的数据库用于识别和分类操作系统。

主动探测方法虽然在近年来得到了广泛应用，但其必须保证与目标网络在同一可达网络之内。此外，由于需要向目标网络发送探测数据包，可能引发流量噪声对目标网络造成一定的影响，且只能掌握探测时网络的情况。

1.2 探测架构

网络资产探测的目标是整个 IPv4 地址空间中的网络资产，由于种类繁多、规模庞大，要在不影响目标网络正常运行的情况下，高效、快速地进行探测，需要考虑统一、有序、合理的探测架构。目前，分布式探测架构已成为各大网络空间探测系统的首选。分布式探测架构通过部署多个具备资产探测功能的探测结点，将 IP 地址划分为多个探测目标 IP 列表，通过构建探测结点选择策略模型，为各 IP 列表选择合理的探测结点，进而进行主机存活探测、端口扫描、获取资产信息等。具体步骤如下：

(1)部署多个具备资产探测功能的探测节点；
(2)控制节点根据策略通知探测节点扫描 IP；
(3)各探测节点对目标 IP 进行主机存活性探测；
(4)各探测节点对目标 IP 进行开放端口探测；
(5)各探测节点对目标 IP 开放端口发送探测报文，并与指纹库对比，获取网络资产信息。

DNmap 引入分布式的 Nmap 扫描框架实现了大型网络探测能力[7]。CENSYS 是最初由密歇根大学的研究人员发行，目前由谷歌提供支持的一款针对整个 IPv4 地址空间的探测系统[8]。该系统的多个扫描工作器先使用 ZMap 执行主机发现扫描，使用扫描插件 ZGrab 完成与主机握手，并从返回包中获得结构化字段，再将结构化字段存入 ZDb 中央数据库中，ZDb 维护每台主机的当前状态并将更新的记录发送到网络前端，研究人员可以在前端查询数据。Shodan 是 2009 年由一位美国学生 John Matherly 创建的，在功能上虽然与 CENSYS 相似，能通过 Web 页面通过特定语法(限定 IP、端口、HTML、证书、设备指纹等)搜索相关 IP 及探测到的所有 IP 对应资产信息，但其采集目标网络指纹的方式是通过拦截网络流量的非入侵式[9]。国内知名网络安全公司知道创宇开发了 ZoomEye 搜索引擎，在集成了多个探测工具如 Nmap、ZMap、Masscam 的基础上，根据探测频率、目标所在区域、防护情况等策略模版执行探测任务，同时构建了指纹规则库、特殊服务识别库、协议分析库等，为用户提供了设备指纹、Web 服务等搜索功能。

2 探测架构和通信流程

2.1 探测架构

多级分布式探测架构如图 1 所示，分为控制层和探测层，其主要特征是利用多个控制结点(图 1 中以 7 个为例)，构成分层路由网络，每一个控制层的控制节点负责一个探测集群的扫描控制，每个上层节点负责 n 个子节点(图 1 中 $n = 2$)，路由节点上有网络性能收集程序。

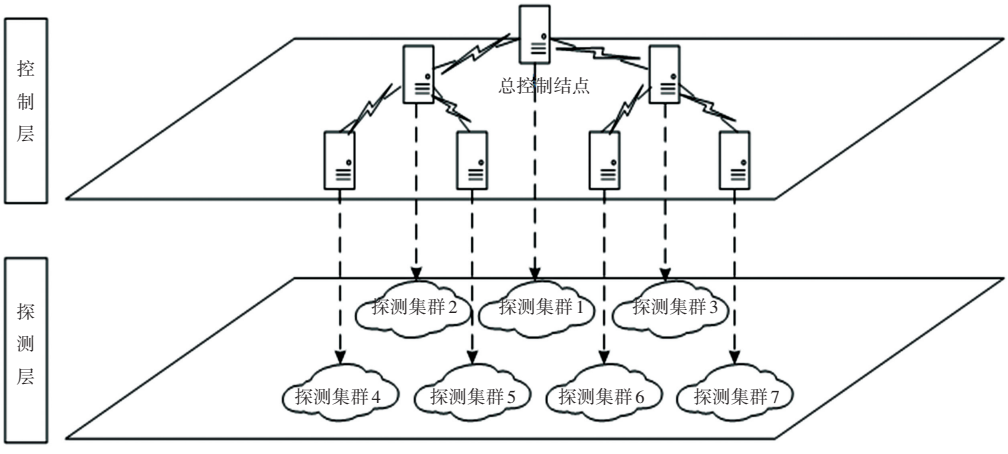

图 1 多级分布式探测架构

Fig. 1 Multilevel distributed detection architecture

该架构的主要优势是摒弃了传统的由一个任务调度节点负责所有探测节点任务分发的工作模式，采用多级控制结点架构，解决了在常见性能计算机环境下，中心结点发送速率受限于网卡发送速率的问题。此外，相比于传统单点扫描，由于采用分布式架构，扫描节点数量的增加可以提升整个探测效率。

控制层与探测层的功能如下：

控制层：处于探测层上层，负责探测任务分配，向探测节点下发探测任务。由于其分层结构，顶层控制结点为总控制节点，负责向其下层控制节点分发探测任务。

探测层：主要负责扫描任务的执行，根据其上级探测节点发送的任务及探测策略，开展相应的探测行为。

控制层、探测层之间使用流量放大机制以提高带宽利用效率。所谓流量放大机制是指下一级可以使用从上一级获取的少量数据来产生不与其他节点冲突的大量数据并发往下一级。

2.2　通信过程

多级分布式探测通信过程如图 2 所示，主要包括各个模块之间的信息通信以及数据流向。广谱扫描系统由分布式任务分发、扫描节点、数据存储、数据解析、数据同步 5 个功能模块组成。

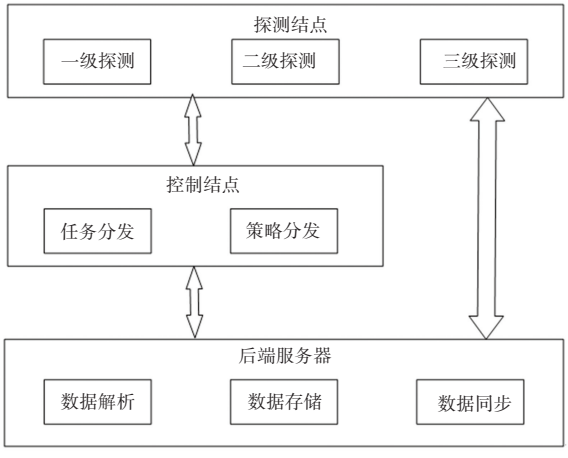

图 2　多级分布式探测通信过程

Fig. 2　Multistage distributed detection communication process

控制节点的任务分发模块负责接收或者生成随机 IP 地址扫描策略，分发给一级探测节点。

探测节点按照策略开展探测，将结果回传至指定位置，通过数据自动回传加密通道，回传至存储服务器中。探测节点按照探测类型不同，可分为一级探测节点、二级探测节点和三级探测节点。一级探测节点负责探测端口开放情况；二级探测节点担负协议交互和 banner 数据的获取；三级探测节点在二级探测的基础上实现 web 应用的深度识别。

后端数据存储负责存储探测过程中产生的原始扫描数据和中间过程数据。后端数据存储分为一级存储和二级存储。一级存储使用非关系型数据库，用以快速存储前端探测回传的非结构化数据；二级存储综合使用关系型、非关系型数据库，用以支撑二级、深度扫描策略生成，为节点脆弱性分析和资源管理与塑造系统提供支持。

数据解析模块根据一级探测结果生成二级探测策略，推送给二级探测节点并根据二级探测结果生成深度探测策略推送至深度探测节点。

数据同步实现探测结果和策略在前端节点和后端服务器之间加密自动回传。

3　流量放大机制

3.1　循环乘群在 IPv4 地址空间的应用

任何主动探测都需要有探测数据包，而 IPv4 域的网络探测需要针对每个探测节点构造不同的探测包，由于 IPv4 地址有 40 多亿个，生成全部的探测包将消耗大量的存储空间，打乱这些探测包的顺序也需要消耗大量的时间。为解决这些问题，ZMap 提出了一种基于循环乘群的解决方式。这不是本文的原创方法，但是是后文的技术基础，所以在此简要阐述。

群：数学上，把由一种集合以及一个二元运算所组成的符合群公理的代数结构称之为群。群公理包含以下 4 个性质：封闭性、结合律、单位元和逆元。比如，所有整数和加法构成一个群。

循环群：一个群中所有元素能由其中的一个元素生成，称之为循环群。比如，正整数加法群的所有元素可以由 1 生成，正整数加法群就是一个循环群。

循环乘群：一个循环群，其二元运算是乘法，那这个群就是循环乘群。

在实际应用中，ZMap 使用整数模 n 乘法群算法建立覆盖 IPv4 地址空间的全排列。IPv4 地址为 128 bit 换算成十进制即 1～232 的整数。所以 ZMap 通过选择 p 为 232 + 15，再从需要遍历的数列中选择一个本源根 u，通过将待排序的数字 n 与 u 相乘之后再与 p 取模，经测试，3 是 p 的本源根。这样，从该群中的任何一个生成元出发，都可以不重复的遍历该群中的所有元素，而所有 IPv4 地址都可以映射到该群上，从而可以通过对群元素的遍历来遍历

IPv4 地址。循环乘群相对于循环加群有一个优势,循环乘群从生成元遍历整个群元素的时候,遍历顺序相对来说规律性没有循环加群那么明显,带有一定随机性。

3.2 任务调度算法

总控制节点任务调度算法见表 1,总控制节点首先生成最大质数 p 和本原根 u,通过随机生成数 c,以及 p,u, 可以以一种随机排列的方式访问整个 IPv4 地址空间;主控制节点通过计算迭代次数来控制向下一级控制节点发送的任务, n 可以看作发送给每个子控制节点的任务数量。通过迭代计算 n 次后,即算出发往下一个控制节点的地址起点,但是任务数量 n 保持不变。这样就可以将整个探测任务平均分配给下一层控制节点。

表 1 总控制节点任务调度算法

Tab. 1 Task scheduling algorithm of master control node

算法	总控制节点任务调度算法
输入	p
输出	四元组(p,u,c,x)
步骤 1	取大质数 p,判断 p 是否大于 2^{32}
步骤 2	计算本源根 u,可通过 u 与乘法运算得到$[1,p-1]$ 中所有的整数
步骤 3	在$[1,p-1]$ 中随机选择一个开始元素 c
步骤 4	获取下层节点数 m,计算 $n=2^{32}/m$,n 向上取整,i 初始化为 0
步骤 5	当 $i<m$ 时,将(p,u,c,x) 发送至层节点 i,否则转至步骤 2
步骤 6	迭代计算 $c=c*u \bmod P$,计算 n 次后 $i=i+1$,转至步骤 5

其他控制节点在接收到上层节点发送的任务后,通过当前数字 c 获知探测任务开始的起点,如果还有下一层控制节点,根据算法将任务平均分配给下一层控制节点,如果没有下一层控制节点,则向探测集群下发扫描任务,其任务调度算法见表 2。

表 2 控制节点任务调度算法

Tab. 2 Task scheduling algorithm of control node

算法	总控制节点任务调度算法
输入	四元组(p,u,c,x)
输出	四元组 (p,u,c,x)
步骤 1	接收上层节点发送的任务四元组(p,u,c,x)
步骤 2	获取下层节点数 m,计算 $x=n/m$, x 向上取整,i 初始化为 0
步骤 3	当 $i<m$ 时,(p,u,c,x) 发送给下层节点 i,否则转至步骤 1
步骤 4	迭代结算 $c=c*u \bmod P$, 计算 x 次后 $i=i+1$,转至步骤 3

任务调度管理服务在任务分发前,会将生成的探测 IP 地址同黑名单进行比对,如果命中,就丢弃,重新生成 IP 地址,如此往复。白名单是任务调度管理服务根据后台管理的临时需求,增加的优先级较高的扫描任务。通过黑白名单机制,可以规避到风险较高的 IP 地址段,也可以重点开展某段地址扫描,见表 3。

表 3 探测节点任务调度算法

Tab. 3 Task scheduling algorithm of detecting node

算法	探测节点任务调度算法
输入	四元组(p,u,c,x)
输出	探测 IP 地址
步骤 1	接收上层节点发送的任务 (p,u,c,x), i 初始化为 0
步骤 2	检查 c 是否在范围$[1,2^{32}-1]$之间,若是,转向步骤 3,否则,到步骤 6
步骤 3	检查 c 是否在白名单中,若是,转至步骤 5,否则转至步骤 4
步骤 4	检查 c 是否在黑名单中,若是,转至步骤 6,否则转至步骤 5
步骤 5	将 c 转化 IPv4 格式,发送给扫描任务拼接程序
步骤 6	迭代计算 $c=c*u \bmod P$,计算 x 次后 $i=i+1$,转至步骤 5

4 实验验证与结果分析

4.1 实验环境

本方法具备快速扫描、分布部署、动态扩充和高容错性的特点,由于 Nmap、ZMap 和 Masscan 不具备分布式扫描的功能,都为单机版运行软件。因此本系统任务分发程序与 Nmap、ZMap、Masscan 于同一台服务器同一网络条件下,并关闭服务器上所有非必要软件。

实验环境搭建于云服务器,操作系统版本为 Ubuntu 16.04 LTS x64,配置信息为 4 个 E5 2620 v3 核心,4 G 内存,100 Mbps 网络带宽,通过 docker 建立一个目标集群,每次测试选取不同的节点作为探测节点进行探测,探测目标为全网络,每个工具进行 10 次探测实验,比较扫描的耗时、网络占用、处理器占用、内存占用、扫描稳定性,实验数据取平均值,各参数意义与具体测算方法见表 4。

4.2 探测时间分析

当使用单个结点对 4 096 个 IP 进行扫描时,本方法在扫描耗时上并没有优势,但是,当使用 64 个结点计算等效全网耗时后,其分布式探测架构缩小了全网扫描的时间,并从网络占用率上优于其他 3 种扫描方式,见表 5。

表 4　各参数意义与具体测算方法

Tab. 4　Significance of each parameter and specific calculation method

序号	参数	意义	测试方法
1	扫描耗时	扫描任务开始到数据落盘所经历时间，单位 s	日志记录(去除程序初始化时间)
2	等效全网扫描时间	按测算扫描耗时估计的对全网进行扫描所需的时间	扫描耗时
3	网络占用	网络流量占网络连接速度比例	系统监视程序显示
4	处理器占用	软件运行时对处理器处理能力的占用比例	系统监视程序显示
5	内存占用	软件运行时占用的物理内存的大小，单位兆字节	系统监视程序显示

表 5　扫描时间对比

Tab. 5　Comparison of scanning time

名称	扫描耗时/s	等效全网耗时	网络占用
本方法	48.348 7	39 天	0.1
Nmap	88 542.72	约 11 776.226 7 年	1.4
Zmap	1.052 5	51 天	98.7
Masscan	1.42	61 天	1.3

4.3　资源消耗分析

资源占用率对比见表 6，本方法无论是从处理器占有率还是内存占用率方面都优于其他扫描方式，说明分布式探测架构相比传统探测方法可降低资源占用率，达到节省资源的目的。

表 6　资源占用率对比

Tab. 6　Resources utilization percentage comparison

名称	处理器占用/%	内存占用/MB	扫描稳定性
本方法	2.8	10.2	稳定
Nmap	20.1	61.6	稳定
ZMap	50	13.9	不稳定
Masscan	3.2	37.0	稳定

从以上测试结果可以得出结论，本系统在扫描速度、资源占用、结果稳定性方面均优于现有软件，并具有可动态扩充的特点。

但是从测试数据看，系统的实测扫描时间与理论计算值有大概 20%的延长，该延长的产生原因为超时时间设置导致，试验中超时设置为 10 s，实际发包时间大概 38 s，这 10 s 可通过流水线运行等并行化技术进行优化，系统实际部署后性能可提高 20%左右。

5　结束语

为了解决现有网络资产探测工具带宽消耗及资源消耗过大的问题，本文提出了一种基于常见性能计算机环境的快速探测方法，使用控制层和探测层两层探测架构，在控制层和探测层各节点之间利用流量放大机制提高带宽利用率从而实现快速全网探测。下一步将继续开展网络探测相关研究，提高探测精准度和扫描效率。

参考文献

[1] LYON G F. Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning[M]. Insecure, 2009:11-89.

[2] 洪宏, 张玉清, 胡予濮,等. 网络安全扫描技术研究[J]. 计算机工程, 2004, 30(10):3.

[3] GARCIAS. DNmap: the distributed nmap [EB/OL]. http//mateslab.weekly.com/dnmap-the-distributed-nmap.html

[4] DURUMERIC Z, WUSTROW E, HALDERMAN J A. ZMap: Fast Internet-wide scanning and its security applications[C]// Proceedings of the 22nd USENIX conference on Security. USENIX Association, 2013:605-620.

[5] 冉世伟. 基于 Masscan 漏洞扫描技术的研究[D].天津:南开大学,2016.

[6] SHAMSI Z, NANDWANI A, LEONARD D, et al. Hershel: Single-Packet OS Fingerprinting [J]. ACM SIGMETRICS Performance Evaluation Review, 2014, 42(1):1.

[7] BARNES J, CROWLEY P. K-p0f: A high-throughput kernel passive OS fingerprinter[C]// Architectures for Networking and Communications Systems (ANCS), 2013 ACM/IEEE Symposium on. ACM, 2013:113-114.

[8] FALCH P B. Investigating passive operating system detection[D]. Oslo:University of Oslo, 2011.

[9] KLEPSLAND M E. Passive Asset Detection using NetFlow[D]. Oslo:University of Oslo, 2012.

(上接第 54 页)

[14] MOUSTAFA N, SLAY J. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)[C]//2015 Military Communications and Information Systems Conference (MilCIS). 2015: 1-6.

[15] KUTUB T, MEIKANG Q, KEKE G, et al. An investigation on cyber security threats and security models[C]//2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing. IEEE, 2015: 307-311.

[16] IEC 61375-1:2012 | IEC Webstore[EB/OL]. [2012-06-21]. https://webstore.iec.ch/publication/5397.