CERNET2 网络环境中 IPv6 地址空间扫描行为检测与分析

Yiwei Li
2019311857
Tsinghua University

摘　　要

随着 IPv6 在世界范围内的逐渐流行，基于 IPv6 的流量和针对 IPv6 的网络攻击也呈现迅速增加的态势。在 CERNET2 网络中存在一个节点对多个节点的大量访问情况，我们使用 DNS 反向散射技术对 CERNET2 网络下这样的 IPv6 流量进行探测，确定了日均 1300 个这样的节点会访问其他 5.7 万个节点。根据其特征用机器学习方法对节点聚类，推断 83% 的节点为已知类型，剩余节点存在疑似 spam 或 scan 行为。本工作从数据上为 CERNET2 网络的安全和管理策略提供依据，希望针对性地提高 CERNET2 网络的安全性。

关键词： IPv6，DNS Backscatter，Scanning

1　简介

地址探测技术被广泛应用于互联网拓扑研究、IoT 应用识别、安全研究和互联网可靠性研究。随着 IPv6 在世界范围内的逐渐流行，使用 IPv6 的流量和针对 IPv6 的网络攻击也呈现迅速增加的态势。而作为中国典型的复杂网络之一，CERNET 早在 1998 年就成立工作小组开始了 IPv6 的实验和部署工作，之后称为 CERNET2 网络 [1]。由于 IPv6 地址具有更大的地址空间和更稀疏的活跃节点，因此基于 IPv6 的地址探测技术仍然是热门话题。另外，如何利用探测结果更好地理解 IPv6 的安全现状也需要更好的探索。

我们发现，在 CERNET2 网络中存在一个节点（originator）对多个节点（querier）的大量访问情况。我们使用 DNS 反向散射技术对 CERNET2 网络下 querier 报告的 originator 进行收集、筛选和分析工作。我们记录反向查询请求相关的 querier 和 originator，具体用于完成以下两个目标：

1) 对 originator 影响的 querier 范围和时间段进行分析，由此确定 originator 的流量对全网流量的影响程度
2) 对 originator 的类型进行分类，通过特征匹配的方式确定良性请求（如 CDN 服务、DNS 服务等），通过无监督聚类的机器学习方法推测存疑请求的类型（如可能的垃圾邮件发送端、可能的恶意爬虫等）

经过二十多天的跟踪记录，我们确定了日均 1300 个 originator 节点会被报告平均 24 万次，影响其他 5.7 万个 querier 节点。这样的 originator 节点里 83% 为已知的良性类型，剩余节点存在可疑的 spam 或 scan 行为。本工作从数据上为 CERNET2 网络的安全管理策略提供依据，希望针对性地提高 CERNET2 网络的安全性。

2 背景

2.1 IPv6 地址探测技术

IP 地址探测的工作流程主要包括活跃种子地址收集、地址建模分析和地址生成算法，以及地址验证和后续分析三大部分 [2]。其基本思想是对已知的种子地址进行统计分析，基于统计规律生成预测地址，再对预测地址进行探测。种子收集部分主要分为采用第三方公开数据的主动探测和通过探针抓包获取的被动探测。在地址建模分析中存在一些用于地址扩展的生成算法，且需要进一步对生成的地址进行验证工作。

2.2 DNS 反向散射

本工作采用 DNS 反向散射技术进行 IPv6 地址的收集。当节点（称为 originator）访问其它某些节点（称为 target）时，target 节点会记录 originator 的 IP 地址，委托一个节点（称为 querier）反向查询该地址，根据返回的结果进行后续的操作。如下图所示，一个垃圾邮件发送端（originator）对多个邮件服务器（target）发送 SMTP 包，在 Targets 处理请求的同时，会先委托 queriers 查询 originator 的身份，假如返回的域名位于黑名单，则拒绝处理这些 SMTP 包。Queriers 对 originator 的 IP 地址的反向查询可以被在 querier 和 DNS 权威服务器路径上的节点探测并记录，作为 IPv6 地址的来源。由于 target 和受其委托的 querier 通常处于同一子网，一般会拥有相同的父域名空间，因此后续实验中我们以 querier 直接指代受到影响的 target。

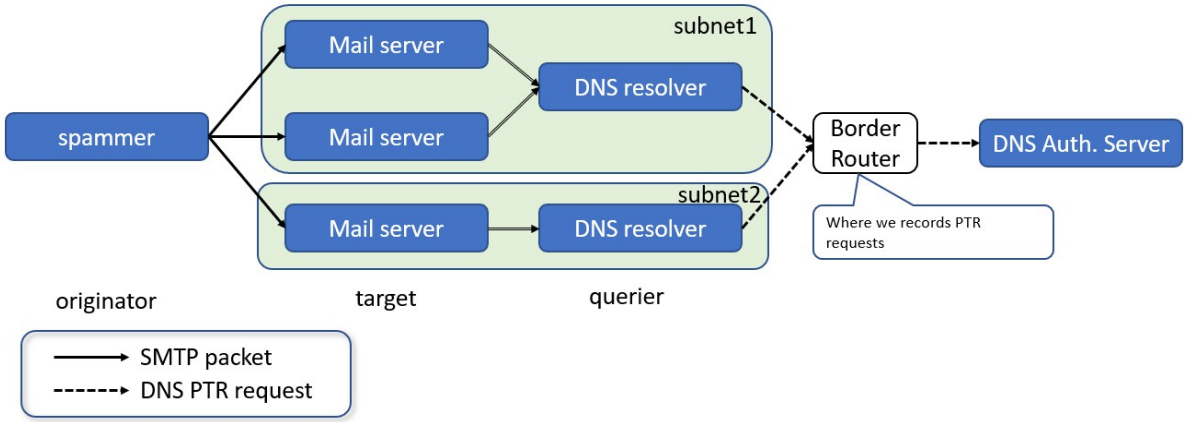

图 1: 利用 DNS 反向散射进行地址探测流程

3 Methodology

我们主要通过对 IPv6 活动进行收集、预处理、特征匹配和聚类分析进行网络内 IPv6 的分析，工作流程如图一所示。

本工作采用 Python 完成所有的环节，一共约 760 行代码，代码将公开于https://github.com/leepoly/cernet2-dns-analysis。下面按逻辑顺序对各处理环节进行细节说明。

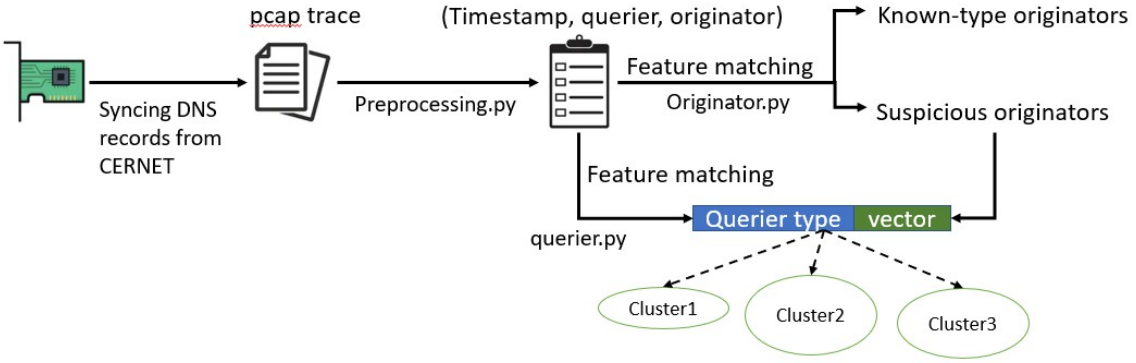

图 2: 利用 DNS 反向散射进行地址探测流程

3.1 数据收集与预处理

本工作从 CERNET2 的边界路由器中进行采集 DNS 请求包。所有 CERNET2 内部节点对外部网络的 DNS 请求与响应包均被 rsync 工具同步到节点中，保存为 pcap 文件。通过 tcpdump 工具解析 pcap 文件，根据 DNS 请求类型是否为 PTR，只保留所有 IPv6 节点发出的反向 DNS 查询请求。由于网络内部的对私有 IP 地址的 DNS 查询不能保证 IP 地址的真实性，影响实验精度，因此，根据对请求包中是否包含 lan 关键字过滤对网络内部的 DNS 查询。由于网络延时、防火墙等各种因素导致远程的 DNS 服务器不能及时接收并响应这些 DNS 请求，因此本工作不记录 DNS 回复。排除不合理 IP 区间（如本地地址、多播地址）后，所有反向 DNS 请求被记录为 (timestamp, querier_ip, originator_ip) 三元组的形式。

由于 CERNET2 网络的复杂性，不可避免地存在偶发 DNS 反向请求，因此本工作只关注一段时间内多次访问 querier 的节点。根据 CERNET2 网络的规模，我们将可供分析的 originator 的阈值设为 20 次/天，且对 originator 访问的不同的 querier 数设下限为 10 个/天，即只对同一天内被十个以上 querier 报告 20 次及以上的 originator 进行后续类型分析，以降低数据的噪声。关于下限选择的细节，我们在第四章进行进一步讨论。

3.2 特征匹配

我们对在预处理中出现过的所有 originator，使用启发式特征匹配的方式进行初步的类型判断。可以利用的信息包括：反向查找 DNS 得到的域名、使用 WHOIS 工具查询 IP 所属自治域（AS）信息、IP 地址本身区段信息和第三方数据库给出的标识信息。不同的类型查询其域名或 AS 信息的难度也有所不同，如 spam 很少自带反向域名信息。通过第三方数据库 [3] 也可辅助确定某些没有 AS 记录的 IP 区段，如未被官方记录的 Google 的某些 IP 区段和国内一些高校的 IP 区段。我们将 originator 的类型主要分为表 1 中的若干类。

通过上表，可以确定大部分正常的访问请求，不能确定的 Originator 类型记为存疑，在之后进行聚类分析。

表 1: Originator 的类型

类型	举例	匹配难度	静态特征
DNS	网络边缘的域名解析服务器	低	域名中包含 dns, resolv, name, cns, ns cache 关键字
Home	家用小型路由器	低	域名中包含 ap, cable, cpe, customer, dsl, dynamic, pop, fiber, flets, home, host, ip, pool, retail, user 关键字
Cernet	学术目的（如爬虫服务）的监测节点	高	域名中包含 cernet 关键字，AS 号属于 133111，23910（CERNET2），133512（IANA）等
Commercial Service	如 Google 的虚拟专用服务器（VPS）	低	域名中包含 vps, cloud 关键字，或 AS 号属于 Google（如 15169）
CDN	网络加速服务器	低	域名中包含 cdn, mip（Mobile Instant Pages）关键字，或 AS 号属于 CloudFlare（如 13335）
Common Services	包括网页服务、授时服务、隧道服务、VPN 服务	低	域名中包含 tunnel, tor, www, ntp, time 关键字，或 IP 符合某些商业隧道区段，如 Teredo（2001::/32）[4] 和 6to4（2002::/16）[5]
Spam	垃圾邮件发送节点	高	较难匹配
Scan	IP 地址空间扫描节点	高	较难匹配

3.3 聚类分析

针对上一阶段未知类型的 originator，本工作记录更多的和 querier 相关的动态信息，以辅助聚类。我们首先使用上一节类似的方式，对 querier 进行类型匹配，但由于 querier 一般为服务器节点，所以划分的类型略有差别。而且相比 originator，querier 更容易查阅到 AS 或域名记录，因此匹配工作的难度相对 originator 大大降低。对于仍然无记录的 querier，我们简单的将其类型视为“未知”并舍弃。

除了判断 querier 类型外，我们还记录 originator 的更多时间相关的信息和空间相关的信息。时间相关的信息包括 originator 被请求的总数及平均被访问的频率。空间相关的信息主要和报告该 originator 的所有 querier 相关，具体如下：

相关 querier 的总数（空间属性）：记录报告该 originator 的所有 querier（称为相关的 querier）的个数，重复的 querier 不予记录。

被报告的请求总数（时间属性）：记录报告该 originator 的所有请求的条数，重复的请求也会记录。

被报告的频率（时间属性）：记录一天内被报告的请求数与 querier 数比值，作为较长期的被报告频率。以及按照半小时为粒度，统计每半小时被报告的请求数与 querier 数比值，取最大值，作为短期的被报告频率，以此统计突发特征。

相关 querier 的 IP 前缀特征（空间属性）：分别记录相关 querier 的 IP 前缀为 8 和 24 的香农熵，统计 originator 是否对 querier 选取有 IP 相关的局部或全局的特征。

相关 querier 的特征向量（空间属性）：记录相关 querier 的类型，归一化后以向量的形式存储。例如，当某个 originator 在一天内被所有 DNS 类型的 querier 报告 14 次，被所有 CDN 类型

表 2: Querier 的类型

类型	举例	特征
Mail	邮件服务器	域名中包含 mail, mx, smtp, post, correo, poczta, send, lists, newsletter, zimbra, mta, pop, imap, hinet(一家位于中国台湾的商业邮件服务器)关键字
Home	家用小型路由器	域名中包含 ap, cable, cpe, customer, dsl, dynamic, pop, fiber, flets, home, host, ip, pool, retail, user 关键字
Cernet	学术目的（如爬虫服务）的被访问节点	域名中包含 cernet 关键字，AS 号属于 133111，23910（CERNET2），133512（IANA）等
Commercial Service	如 Google 的虚拟专用服务器（VPS）	域名中包含 vps, cloud 关键字，或 AS 号属于 Google（如 15169）
CDN	网络加速服务器	域名中包含 cdn, mip（Mobile Instant Pages）关键字，或 AS 号属于 CloudFlare（如 13335）
Common Services	包括网页服务、授时服务、隧道服务、VPN 服务	域名中包含 tunnel, tor, www, ntp, time 关键字，或 IP 符合某些商业隧道区段，如 Teredo（2001::/32）[4] 和 6to4（2002::/16）[5]
Firewall	防火墙	域名中包含 wall, fw 关键字
Antispam	反垃圾邮件服务	域名中包含 ironport, spam 关键字
DNS	网络边缘的域名解析服务器	域名中包含 dns, resolv, name, cns, ns cache 关键字

的 querier 报告 6 次，则其特征向量可表示为（DNS=0.7, CDN=0.3）。

我们将上述信息统一为一个高维元组，使用 sklearn 库里的 K-平均算法对这些 originator 进行聚类。关于聚类数的合理选择和详细结果，将在第四章进行说明。

4 结果与分析

本工作收集了自 2019 年 12 月 16 日 0 时到 2020 年 1 月 11 日下午的 CERNET2 网出口流量。我们分三部分展示结果：第一部分展示 originator 的影响范围和总体情况；第二部分是对可疑 originator 的聚类结果和分析；第三部分是 backscatter 活动随时间变化的趋势。

4.1 影响范围和总体情况

本工作一共收集到约 400 万条有效反向请求，平均每天独立的 originator 数目为 1300 多个。下图是各 originator 所占比例，从图中可见，存在较多 CERNET2 内部已注册的节点进行科研工作，其次是 CDN 和 DNS 服务，一共约 8 万条记录。图中仍有 17% 的 originator 无法根据特征匹配简单的归类，称为可疑节点。

进一步，我们按照一天内被报告的频率进行降序排序，我们得到了热度前 10，前 100 和前 1000 的 originator 类型分布图。从下图中可以发现，不同类型的 originator 的请求频率存在差别，

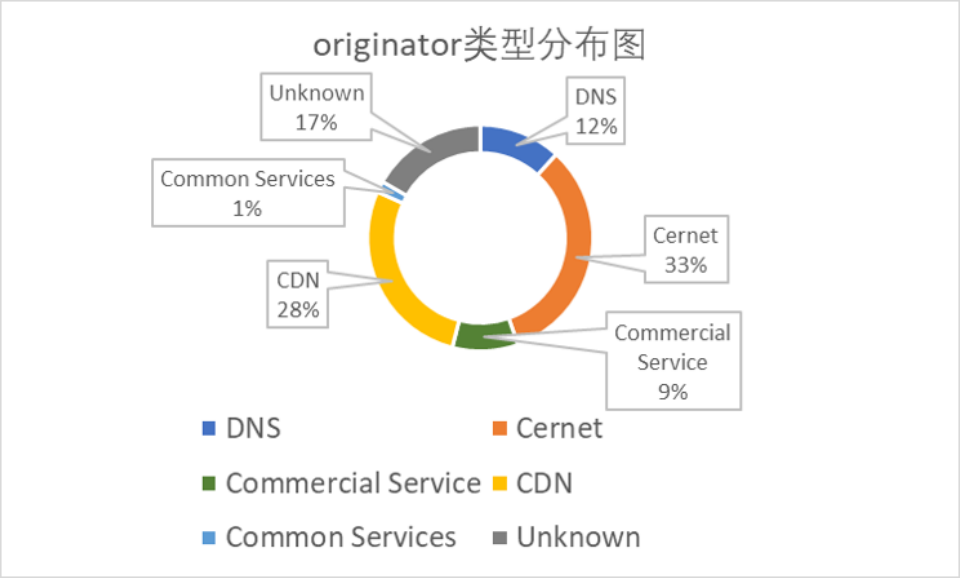

图 3: originator 请求分布图

如 Cernet 类型的请求集中在前 100 个 originator 里，而 CDN 和 DNS 类型的 originator 更容易在更靠后的 100-1000 的 originator 中产生。这间接说明了使用频率作为动态参数的一部分用于类型聚类的合理性。

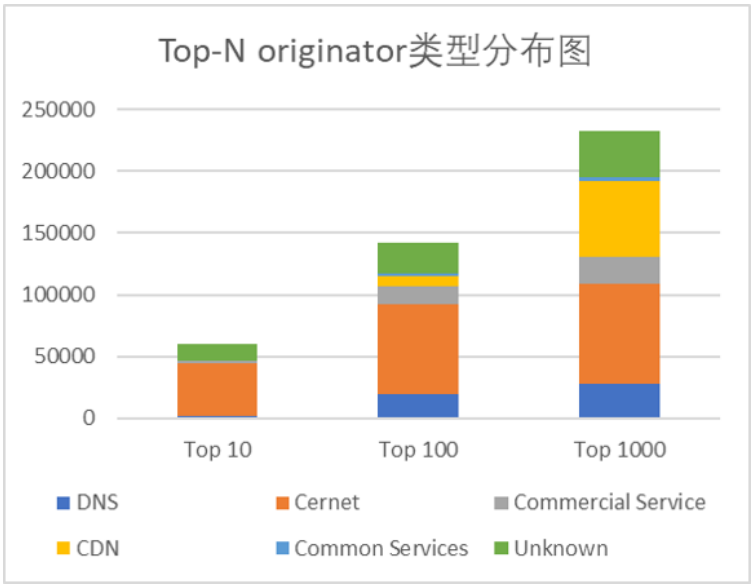

图 4: 访问频率前 N 个 originator 的请求分布图

4.2 可分析的相关 querier 数目下限选择

合适的（相关 querier 下限，被报告的请求次数下限）二元对，可以有效平衡工作的分析开销和噪点。我们收集某一天内的所有 DNS 请求，映射到以相关 querier 下限和平均被报告的请求频率为坐标轴的图上，发现随着相关 querier 数目增多，对应 originator 的平均请求数也大致正相关的增加。因此相关 querier 数目下限确实筛选掉了噪声点（低报告次数）。最终我们考虑实际分析工作量，参考图上的红点选择了（10，20）作为我们的下限。

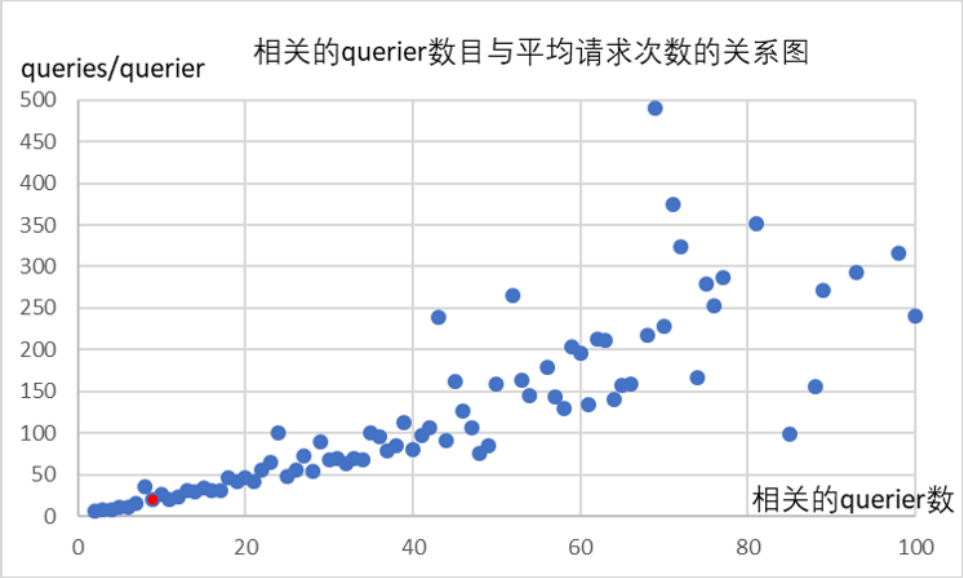

图 5: 相关 querier 数目与平均请求次数关系图

4.3 对可疑节点的聚类分析

对于第三章里无法被辨别的可疑节点，我们根据其相关的 querier 特征进行 K-平均算法聚类。聚类的数目需要人为确定。我们使用 calinski_harabaz 公式评估某一个聚类数的聚类效果分数。分数越高，聚类的效果越好。但由于选取的特征只能覆盖到 querier 的部分行为，因此理论上不存在一个合理的聚类数值使得聚类效果最好。也就是说，calinski_harabaz 分数随着聚类划分增多而增大，不存在极大值点。下图展示了 calinski_harabaz 分数随聚类数 n 的关系：

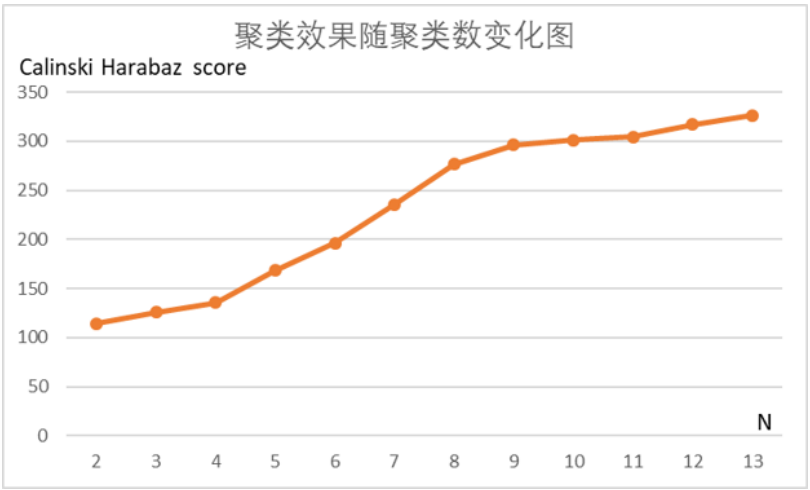

图 6: 聚类效果随聚类数变化图

尽管无法确定一个最优的聚类数值，但从上图看出当 N=8 或 9 时存在着聚类效果的拐点，其后的聚类效果增量明显变小。因此我们选择 N=8 作为我们的聚类数目。

确定聚类数目后，我们发现聚类可以说明一些可疑的 originator 的请求特征，下图是采用主成分分析（Principal component analysis）降维后的聚类效果展示，黑色为数据点，红色为聚类后的中心点。由于我们先聚类后降维，因此聚类数在二维图中表现不合理。但聚类必须在高维进行，如果先降维再聚类会丢失特征信息。我们对各聚类所属的可疑 originator 占比进行了统计，

将各聚类中心点的相关 querier 特征向量用下图表示。

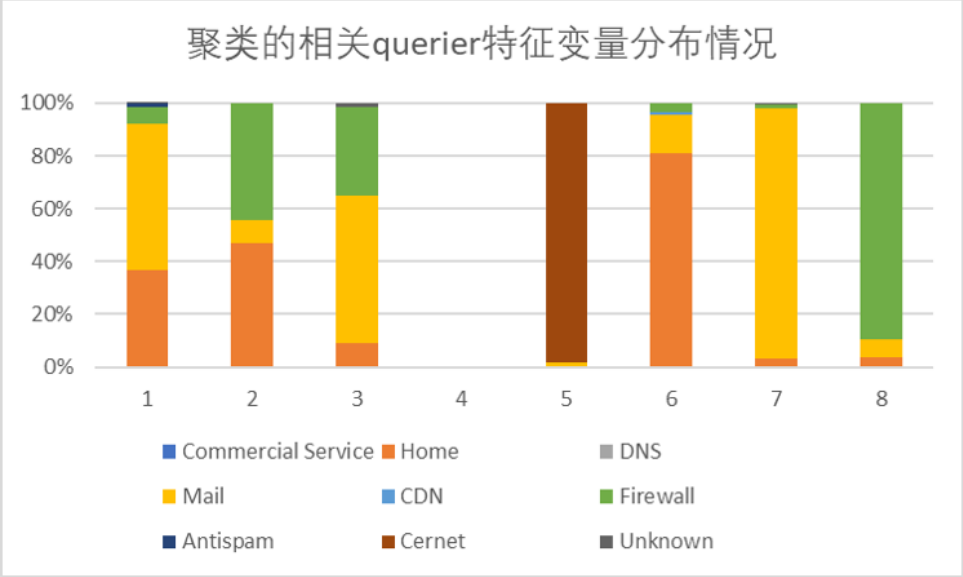

图 7: 聚类效果随聚类数变化图

我们对其中一些类别进行一些说明:

表 3: 聚类的说明

聚类编号	querier 特征向量	占比	说明
1	(Mail=0.369, Home=0.553, ...)	19.7%	疑似 spammer
2	(Home=0.469, Firewall=0.441, ...)	6.4%	疑似 scanner
3	(Mail=0.558, Firewall=0.335, ...)	15.3%	疑似 spammer
4	(0, 0, ..., 0)	7.6%	未被已有特征向量覆盖，仍无法判断
5	(Cernet=0.982, ...)	3.2%	疑似攻击
6	(DNS=0.809, Home=0.148, ...)	14.0%	疑似 scanner
7	(Mail=0.947, ...)	26.8%	疑似 spammer
8	(Firewall=0.895, ...)	7.0%	疑似攻击

1) 如 7 号聚类的中心点 querier 特征向量为（Mail=0.9468, ...），说明这类 originator 访问 mail server 远高于其他 querier。在过滤掉正常邮件请求后，可作为疑似 spammer 进行进一步地调查。
2) 如 5 号聚类的中心点 querier 特征向量为（Cernet=0.9822, ...），这类 originator 基本只访问校内服务器。由于校内用作学术研究的个人一般可以查阅到 IP 的 AS 信息，所以该类 originator 可能来自于校外的攻击。这类 originator 日均发生十多起。
3) 如 6 号聚类的中心点 querier 特征向量为（DNS=0.809, Home=0.148, ...），这类 originator 访问大量 DNS 服务器。可能为个人架设的 DNS resolver，或疑似 scanner（例如，处理本工作的服务器自身也产生大量 DNS 请求）。

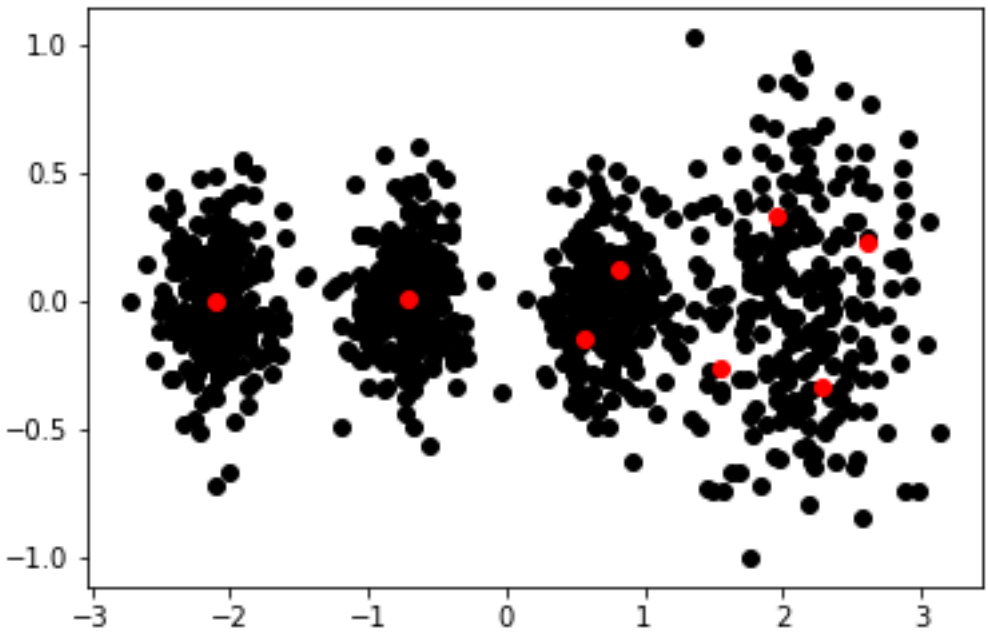

图 8: 聚类效果的可视化

4.4 backsactter 活动随日期变化的趋势

根据观测情况，CERNET2 网络平均总 DNS 请求量（包含 v4 和 v6 请求）为日均 48460.3 万条，其中总的 PTR 请求（包含 v4 和 v6）为日均 6972.4 万条，而 v6 的 PTR 请求只有 50.7 万条，相比总 PTR 请求仅占 0.7%，说明 originator 在 IPv6 空间远没有 IPv4 空间活跃。经过我们筛选后，超过阈值的有分析价值的 PTR 请求为日均 24.9 万条，对应不同的 originator 数目 1380 个，相关的 querier 数目为 5.7 万个。我们整理了近半个月的 originator 数和各类型占比随日期的变化趋势，从图 9 中可以看出，可疑类型的波动较明显，说明归类工作仍不够细致。另外 1 月 2 日的采集量异常小，跟踪查到是当天的预处理出现失误，漏掉了大部分 DNS 请求。11 日的数据尚未收集完全，因此也出现了类似的情况。

5 相关工作

传统方法如 Beverly 等提出的 [6] 对 IPv6 探测需要收集地址种子、生成可能地址、对新地址验证三个步骤。由于 IPv6 具有 128 位的地址空间，远大于 IPv4 的 32 位空间，且地址分布稀疏，因此 IPv6 地址探测的效果严重依赖于生成算法的有效性。而且，由于地址验证环节需要目标节点的应答，因此存在节点针对这类探测做针对性的拒绝回答以隐藏自己，造成了假阴性探测。本工作采用 DNS 反向散射对网络的请求进行探针式的被动探测，并不解决探测所有 IPv6 空间的问题，跳过生成地址的环节，转而根据已知的部分节点的请求行为推测分析节点的类型，提供网络安全方面的建议。这样可以保证在不考虑源地址真实性的情况下，所有的节点都是真实存在的，没有假阴性的探测。使用 DNS 反向散射进行探测，由于和 originator 并无直接连接，因此

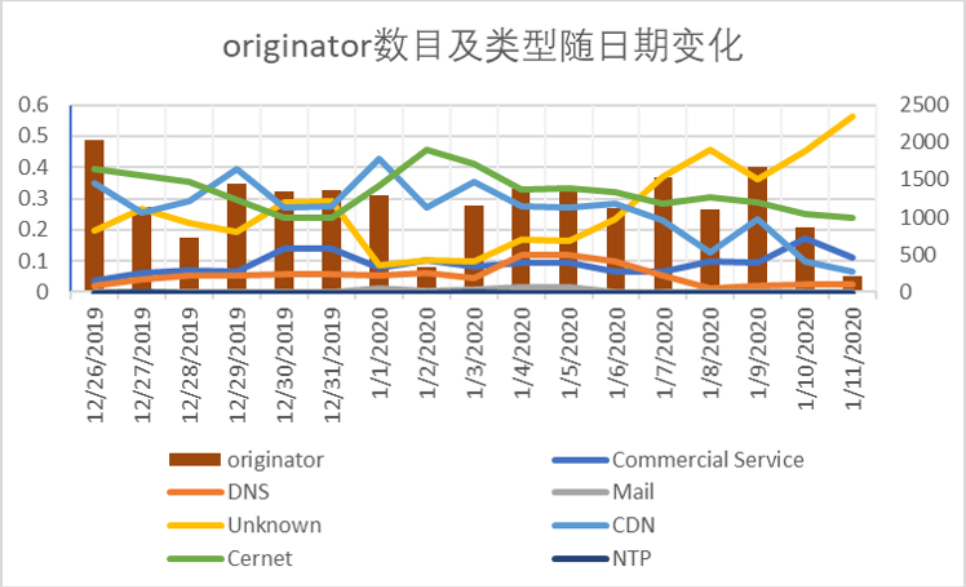

图 9: originator 数目及类型随日期变化

originator 很难针对这种探测做针对性的隐藏。

以往的工作更多将 DNS 反向散射作为直接评估网络安全的工具。如 Herwig 等在 [7] 中利用了正常的节点对攻击行为进行 DNS 反向查询这一特点，进行非成功探测的跟踪。本工作更多地利用 DNS 请求所呈现的流量分布，先确定可疑节点，再对其进行类型分析。[8] 中提到利用到对网络内 DNS 访问聚类进行流量分析，但和本工作不同，他们并未限定跟踪 PTR 请求，因此无法归类可疑节点并进行后续安全性分析。

和本工作想法及设计最相近的是 Fukuda 和 Heidemann 的工作 [9]，但我们在其工作基础上有一些不同：我们使用 CERNET2 出口路由器处探测所有 DNS 请求，而不是使用公开但相对陈旧的 DNS 权威服务器记录，简化了复杂 DNS 层次结构带来的 DNS 缓存等问题，可以得到更为可靠的时间戳和请求数。我们使用无监督方式对可疑节点进行聚类，避免了随机森林方法需要长期跟踪（如通过 darknet）确定一部分 originators 的类型标签，使得实验更为可行。他们在 IPv6 下的实验方法 [10] 对 scanner 的检测依赖于第三方的公开黑名单匹配，这样的黑名单一般更新不及时或很难匹配，因此本工作没有考虑黑名单匹配。

6 总结

本工作使用 DNS 反向散射技术，对 CERNET2 出口 IPv6 流量中的 DNS 反向查询请求作统计和分析，得到当前 CERNET2 网络 IPv6 的 originator 分布情况和影响程度，并采用特征匹配和聚类分析相结合的方法，试图对 originator 的类型做推断。本工作确定了 originator 日均约 1300 多个，总的有效 PTR 请求数日均 24 万条。除去 Cernet、CDN 等类型后，剩余 17% 的可疑类型具有可聚类的特性，存在疑似 spammer 和 scanner 等行为。本工作在 Fukuda 和 Heidemann[9] 的基础上，使用无监督聚类分析省去了构建标签的开销。

未来如果需要深入继续该思路，我认为可以尝试更复杂的无监督模型，如：使用概率图模型，将 querier 的特征作隐变量，根据 originator 和 querier 的关系使用 PLSA 主题模型聚类。另

外，应尝试更长期的监控，得到更完整的趋势图。

致谢

感谢刘莹老师一学期以来对网络体系结构、IPv6 中的协议、安全性和可审计性方面的介绍，让我作为一名非网络安全一级学科的学生也收获很多。感谢保君学长和伟彬学长在论文选题、实验平台、实验数据以及论文写作上的有效指导和无私帮助。

参考文献

[1] 严程, 李星, 陈茂科, 等. CERNET IPv6 试验床[J]. 电信科学, 2002, 18(3):35-38.

[2] 王之梁. IPv6 网络地址空间智能探测研究与实践[R]. [出版地不详]: 清华大学网络研究院, 2019.

[3] LSY.CN. IPv6 地址查询工具[EB/OL]. 2020. https://ip.zxinc.org/ipquery/.

[4] HUITEMA C. Teredo: Tunneling ipv6 over udp through network address translations (nats)[J]. 2006.

[5] CARPENTER B, MOORE K. Connection of ipv6 domains via ipv4 clouds[M]. [S.l.]: RFC 3056, February, 2001.

[6] BEVERLY R, DURAIRAJAN R, PLONKA D, et al. In the ip of the beholder: Strategies for active ipv6 topology discovery[C]//Proceedings of the Internet Measurement Conference 2018. [S.l.]: ACM, 2018: 308-321.

[7] HERWIG S, HARVEY K, HUGHEY G, et al. Measurement and analysis of hajime, a peer-to-peer iot botnet.[C]//NDSS. [S.l.: s.n.], 2019.

[8] PLONKA D, BARFORD P. Context-aware clustering of dns query traffic[C]//Proceedings of the 8th ACM SIGCOMM conference on Internet measurement. [S.l.]: ACM, 2008: 217-230.

[9] FUKUDA K, HEIDEMANN J. Detecting malicious activity with dns backscatter[C]//Proceedings of the 2015 Internet Measurement Conference. [S.l.]: ACM, 2015: 197-210.

[10] FUKUDA K, HEIDEMANN J. Who knocks at the ipv6 door?: Detecting ipv6 scanning[C]//Proceedings of the Internet Measurement Conference 2018. [S.l.]: ACM, 2018: 231-237.