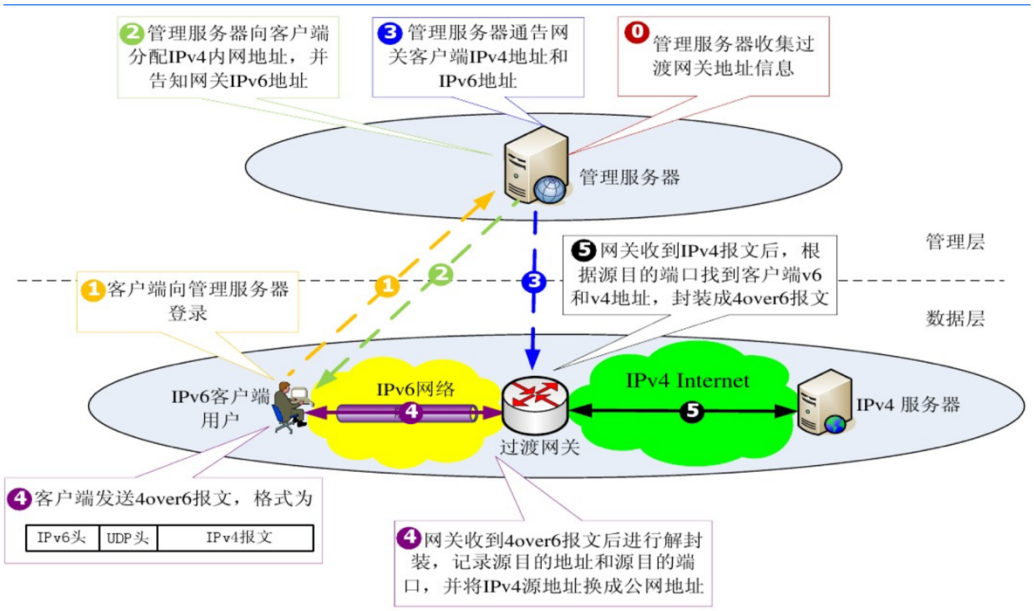
4 over 6 原理验证系统开发及流量数据分析

设计文档

一、 所在系列及赛项：网络技术挑战赛之创意系列（A）

二、 目标问题

IPv6 是下一代互联网协议，它的设计用以解决 IPv4 中的许多问题（最重要的一个就是解决了 IP 地址不够分配的问题）为目标。但是，在互联网大规模支持 IPv6 之前，IPv4 到 IPv6 如何平滑过渡是需要人们仔细考虑的一个问题。例如，由于互联网服务提供商的现有网络设备大部分都只支持 IPv4，我们有必要设计一种机制允许 IPv6 用户访问历史遗留的 IPv4 服务，而 IPv4 over IPv6 就是这样的一种机制。其大致原理是采用了“IPv4 over IPv6”隧道方案来实现，IPv6 网络中的用户在希望访问 IPv4 服务时，将 IPv4 包封装在 IPv6 包中发送给一台能够直接访问 IPv4 服务的服务器。该服务器将进行实际 IPv4 资源的访问，然后再将得到的 IPv4 响应封装在 IPv6 包中发送回用户。为了能够通过 IPv4 地址将 IPv4 响应返回给相应的用户，IPv4 over IPv6 服务器需要维护一个 IPv4 和 IPv6 地址的映射表并为需要 IPv4 over IPv6 服务的用户分配 IPv4 地址。原理大致如下图所示：



“马上 6”平台是中国目前最全的 IPv6 资源网站，为广大用户提供最简单快捷、免费高速的 IPv6 网络访问 IPv4 网络的接口，用户无需其他操作即可在 IPv6 网络中访问 IPv4 的资源。我们与“马上 6”公司取得了合作，项目第一阶段为开发一个为用户提供 4 over 6 服务的实验系统，能够通过手机 APP 在 IPv6 环境下访问 IPv4 资源。进一步，“马上 6”产生的数据流量非常大，在第二阶段，我们计划采集“马上 6”网关上的流量，利用机器学习的算法来对其进行 offline 的解析和分类。第三阶段，我们希望能够在“马上 6”的网络设备上部署我们的分类器，对途经的流量进行 online 的管理和控制，以达到提高其产品的服务质量的目的。

三、 思路规划与作品设计

（1） 第一阶段（已经完成）：

本阶段我们主要完成的工作如下：

a） Centos 上抓包并存储为 pcap 程序的开发:本程序采用 C 语言开发，具体实现可以参考所附代码中的 pcapTest.c 文件，所完成的具体功能为在 Centos 环境下指定特定网口进行数据包的抓取，考虑到存储和后阶段解析的方便性，我们选择每 1000000 万个数据包存储为 1 个 pcap 文件，其中每个数据包只截取真实数据包的前 110 字节（这已经满足了我们后面阶段的解析分类需求）。

b） 多篇顶级会议“流分类”相关论文的调研:磨刀不误砍柴工，在正式进行数据分析流分类之前我们还进行了充分的调研，阅读了数篇国内外顶级会议中发表的有关“流分类”的论文，并从中总结出了后阶段的大致思路。

c） “4 over 6”原理验证系统客户端和服务器端的开发：该部分工作通过实现 4over6 隧道最小原型验证系统，以实现在安卓手机上，通过 IPV6 网络访问 IPV4 网站的功能。分为客户端和服务器端实现，其中客户端实现在安卓设备上实现一个 4over6 隧道系统的客户端程序：实现了安卓界面程序用以显示隧道报文收发状态（java 语言）；启用安卓 VPN 服务（java 语言）；实现底层通信程序对 4over6 隧道系统控制消息和数据消息的处理（C 语言）。此外，服务器端实现在 linux 系统下，实现 4over6 隧道系统服务端程序：实现服务器端与客户端之间控制通道的建立与维护；实现对客户端网络接口的配置；实现对 4over6 隧道系统数据报文的封装和解封装。

（2） 第二阶段（已经完成）：

本阶段我们主要完成的工作如下：

a） 在“马上 6”公司的出口网关上采集流量数据：为获取机器学习所需数据，我们自行开发了基于 pcap 的抓包程序，在“马上 6”公司提供的交换机上进行抓包，数据保存为 pcap 格式并存储到服务器上。采集时长半个小时，共抓到约 15 G 原始流量数据(数据包只保留前 110 个字节，截断多余负载)。

b） 对获取的数据进行解析并进行分流：在采集到原始数据之后，我们开发自行 python 脚本对原始数据进行了解析和分流，即将原始包数据分成一条条的数据流，用以后面阶段的机器学习算法的训练之用。

c） 使用整条数据流进行训练，进行有监督机器学习：在进行无监督学习之前，我们使用了有监督学习进行了预训练。这里的实现可以发现在有准确 label 的情况下（这里的 label 基于端口号给出），有监督机器学习的准确率是非常高的，可以达到 98%以上。

d） 使用整条数据流进行训练，进行无监督聚类学习：整个项目的机器学习部分的重点是在于无监督的机器学习聚类上，但由于我们所采取的真实数据中未知应用类型太多（动态端口号等技术的影响），所以我们目前对 unknown 的流仅是采取了预聚类的方法，目前无监督机器学习的准确率可以达到 60%左右（当然我们还没有做应用层的

分析)。

e） 尝试扩展到 online 分类（聚类）：在有了无监督机器学习的 demo 之后，我们又尝试将我们的 demo 扩展到 online 版本，即不是使用整条流的全部数据而仅仅使用流的前几个包来确定流的类别，实验发现在 online 的情况下，用每条流的前 5 个包来识别整条流效果和使用整条流的效果大致一样，这也证明了我们的项目完全扩展为 online 版本后准确率应该不会有大的影响。

（3） 第三阶段（下一步计划）：

下一阶段我们的规划如下：

a) 离线分类测试完成后，计划在“马上 6”的网关上进行部署：由于我们前面阶段已经完成了 online（根据一条流的前几个包进行分类）的扩展工作，所以我们在项目的后期只需要将这个版本部署到真实的网关上即可，进行实时的测试与分析。

b) 通过 QoS 和协议类型对流量进行综合分析：由于我们无监督机器学习的部分准确率仍然有待提升，其主要原因是我们所获取的数据中“未知应用类型”占比过多（几乎到了 50%的占比），导致我们的聚类算法准确率不是很高。因此在项目后期可以想到的途径就是通过 QoS 和协议类型对流量进行综合分析，以此获得流的准确应用标签类型，进而提高无监督机器学习聚类准确率。

c) 有针对性地改变网络的行为，实现对数据流的动态识别和管理：在利用了 Qos 和协议类型分析之后，我们还计划对我们的数据进行网络行为的分析，进一步地提升我们的模型的预测准确率。

d) 数据集匿名化并予以公布：未来我团队计划将我们的数据处理为公开的数据集，因为我们的数据集具有典型意义（未知流量占比很多的真实数据集），因此十分具有公开以供国内外学者共同研究的意义。

四、 作品实现

（1） “4 over 6”原理验证系统客户端和服务器端的开发：关于本部分客户端和服务器端的开发请详见附件中“4over6 客户端与服务器端开发文档”，其中有对本部分的原理、内容、设计与实现、遇到的问题、运行方式及效果的详细介绍。

（2） 原始数据获取：为获取机器学习所需数据，我们自行开发了基于 pcap 的抓包程序，在“马上 6”公司提供的交换机上进行抓包，数据保存为 pcap 格式并存储到服务器上。采集时长半个小时，共抓到约 15 G 原始流量数据(数据包只保留前 110 个字节，截断多余负载)，处理后筛选出约 140 万条流，用以实现本项目提出的 offline 机器学习下的网络流量检测分析。pcap 抓包程序运行环境为 Centos，代码使用 C 编写，具体代码参见 pcapTest.c。在抓取了数据之后的项目代码均使用 python 编写。

（3） 数据包分流阶段：这里值得一提的是我们在分流的时候进行了三种种方法的分流对比，第一种是采取前人比较通用流行的方法即五元组（源地址，源端口，目的地址，目的端口，协议类型）的方法来区分流；第二

种是单纯地采取数据包的三元组（源端口号、目的 IP 地址、目的端口号）来区分流；第三种是采取上述三元组+握手信息（SYN/FIN）相结合的方法来分流。实验显示，单纯使用三元组方法即上述第二种方法试验分流效果就已经很好了（这里的实验分流效果指分出来流的数量和大小特征）。因此我们在后面阶段均采用“三元组”分流得出的数据来进行机器学习的算法训练，而这一点也是我们的挑战创新所在。

（4） 流特征提取：获取的原始 pcap 数据文件包含了一部分分类所需的流量信息，如包长度、流量方向等，但是另外的一些信息需要进行特征提取，对原始数据进行预处理才可获得，如包间隔等，这些信息在流量分类中是很重要的。接下来的工作就是根据我们的预备方案在原始数据集上提取特征，整理出机器学习算法的输入数据集。最终筛选出的数据流特征包括总包数、流持续时间、平均包长度、平均负载长度、平均包间隔等。由于数据采集环境在 NAT 网关下，因此一些传统方法中用到的特征如数据包源地址等无法使用，这也为我们的项目增添了挑战性。特征的提取和数据预处理代码参见 FlowSolve 文件夹下的代码。

（5） 机器学习算法应用：在使用机器学习进行流量分类阶段，我们借助于 python 的 sklearn 机器学习库，分别采用了有监督的机器学习和无监督的聚类两种方法。有监督的学习包括 KNN、逻辑回归、随机森林、C4.5 决策树、朴素贝叶斯等，参见 single_machine_learning.py。无监督的学习是我们的主要关注点，主要包括一些非常成熟的聚类方法，如 K-Means、DBSCAN、AutoClass 等，参见 cluster.py。目前基本已经实现的主要是使用 K-Means 方法的聚类分析。而 DBSCAN 由于本身算法空间复杂度过高，服务器内存不足以容纳全部采集到的数据，因此正在考虑将数据集分批进行训练，关于 DBSCAN 聚类算法的实现将在项目的后面阶段继续延伸。目前我们训练使用的流类别标签是通过端口分析生成的，然而随着动态端口和端口混淆技术的广泛使用，端口分析生成的标签存在不准确性，同时存在很多使用未注册端口的流量数据。因此我们正在尝试使用一些 DPI（Deep Packet Inspection）工具来进行流类别的标记，如 Qosmos 等。

（6） 地址匿名化公布：由于本项目抓取的是真实的用户数据，因此在实现的过程中要注意保护用户的隐私，因此我们计划在数据集发布之前对其进行匿名化处理。主要进行以下三个方面的工作：首先，将 IP 地址匿名化，其次，消除数据中的用户通信内容，最后，重新计算 IP 头中的校验和。本项目中，使用基于 crypto-PAn 算法进行 IP 地址匿名化的 C 库——“CryptopANT”（written by Yuri Pradkin, University of Southern California, Information Sciences Institute, CA.）来进行我们的 IP 地址匿名化服务。其满足的功能包括：“IPv4 class awareness”（IPv4 地址类感知）、“ Optional partial anonymization”（可选择部分匿名化）、“ IPv4 IPv6 MAC encrypt”（支持 IPv4、IPv6、MAC 地址加密）、“ decryption”（支持反匿名化）、“ Key generation”（支持自动生成密钥）。基于以上功能，我团队认为“CryptopANT”开源工具是满足我们的需求的。

五、 创新与特色

（1） 4over6 客户端与服务器端创新：本部分通过实现 IPV4 over IPV6 隧道最小原型验证系统，以实现在安卓手机上通过 IPV6 网络访问 IPV4 网站的功能。最大的创新与特色就是实现了从客户端到服务器端一整套的流程，实现了服务器端到客户端之间控制通道的建立与维护，实现了对 4over6 隧道系统数据报文的封装与解封装。用户可完全依靠我们的项目实现校园网 IPV6 环境下畅通无阻的上网服务。

（2） “三元组流分类”方法的创新：在流量数据的分类过程中，我们参考了一些以往的工作。以往的流量分类方面的工作通常都是利用五元组即(源地址，源端口，目的地址，目的端口，协议类型)来区分一条流，在流特征筛选时往往也需要双方的地址信息。这样的做法在理想的网络环境下是没有问题的，但是在实际应用中，往往会遇到一些特殊情况，如 NAT 网关屏蔽了源地址或目的地址等等。我们的项目在前人的基础上，拓展实现了这样的非理想网络内部的流量分类方法，经实验发现同样取得了不错的效果。

（3） “二次聚类”方法的创新：由于我们获得的真实数据集本身非常的不规整，经过统计其中未知应用类型（unknown）网络流占比超过了 50%，这对我们的无监督机器学习算法的准确率带来了极大的挑战，由于 unknown 占比太多导致最终聚类结果绝大多数类中最多的流都是 unknown，这就在一定程度上失去了聚类本身的意义。因此我们创新性地提出了对 unknown 部分数据进行预聚类（即第一次聚类），使得 unknown 在正式聚类算法开始之前能够降解为更小的维度，获得更多的类，以此提高最终聚类算法的准确率，经过实验“二次聚类”比“单次聚类”至少提高了 20%左右的准确率。

（4） Qos、应用层协议类型、网络行为分析：虽然上述方法很大地提高了我们聚类算法的准确率，但最终结果还是存在很大的提升空间。因此我团队在今后项目中的重点在于对数据进行精确程度更深的 Qos、应用层协议类型、网络行为等的分析，相信在未来的工作中我们的模型性能能够获得更大的提升。

六、 未来工作

未来，我团队将继续在老师的指导下，与“马上 6”公司进行更为深入紧密的联系与合作。对比多种机器学习算法的有效性、准确性，并从网络流分类结果出发，根据我们积累的经验，提炼出网络数据流量优化的方案与思路，争取将本项目真正落地部署到实际网络环境中，计划在“马上 6”的网关上进行部署我们的模型，以此提高“马上 6”公司的产品服务质量。同时，通过 QoS 和协议类型对流量进行综合分析，有针对性地改变网络的行为，实现对数据流的动态识别和管理，最终争取能够数据集匿名化并予以公布，给国内外学者提供宝贵的经验与总结。

七、 参考文献

[1] Thuy T. T. Nguyen, Grenville Armitage, Member, IEEE, ACM, Philip Branch, and Sebastian Zander, “Timely and Continuous Machine-Learning-Based Classification for Interactive IP Traffic”

[2] Jeffrey Erman, Martin Arlitt, Anirban Mahanti , University of Calgary, 2500 University Drive NW, Calgary, AB, Canada, “Traffic Classification Using Clustering Algorithms”

[3] Gal Frishman, Yaniv Ben-Itzhak , Oded Margalit , “Cluster-Based Load Balancing for Better Network Security”

[4] Laurent Bernaille , Renata Teixeira , Ismael Akodkenou , Augustin Soule , Kave Salamatian, Universit´e Pierre et Marie Curie, Thomson Paris Lab Paris, FRANCE, “Traffic Classification On The Fly”