Design and Implementation of Background Traffic Management System in Network Scene

Ningyi Liu[1,2,a], Xiaorui Gong[1,2,b], Wei Huo[1,2,c] and Zhenyu Song[1,d,*]

[1] Institute of Information Engineering, Key Laboratory of Network Assessment Technology, Chinese Academy of Science, Beijing Key Laboratory of Network Security and Protection Technology, Beijing, China

[2]School Of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

[a]liuningyi@iie.ac.cn, [b]gongxiaorui@iie.ac.cn, [c]huowei@iie.ac.cn, [d]songzhenyu@iie.ac.cn

*Corresponding author

Keywords: Network testbed, Traffic generate, parallel scenes.

Abstract. We design an administration system to solve the isolation problem in a multi-user shared network testbed which holds multiple background traffic generators simultaneously running in parallel experiment scenes. Traffic generators embedded in an experiment scene are dedicated to dynamically generate background traffic for that scene based on session analysis, re-stamping and recombination of the captured traffic from the real network. This administration system has been applied to a network testbed and can stably and reliably manage traffic generators in 4 parallel experiment scenes.

网络场景背景流量管理系统的设计与实现[1]

刘宁逸[1,2,a]，龚晓锐[1,2,b]，霍玮[1,2,c]，宋振宇[1,d,*]

[1]中国科学院信息工程研究所 中国科学院网络测评技术重点实验室 网络安全防护技术北京市重点实验室，北京，中国

[2]中国科学院大学 网络空间安全学院，北京，中国

[a]liuningyi@iie.ac.cn, [b]gongxiaorui@iie.ac.cn, [c]huowei@iie.ac.cn, [d]songzhenyu@iie.ac.cn

[*]通讯作者

关键词: 网络测试床; 流量生成; 并发场景

中文摘要. 本文通过对真实网络流量进行会话分析、重新标记并重组，合成为背景流量，通过网络配置解决背景流量在并发场景中的网络隔离性问题，实现网络测试床中的背景流量管理系统，并通过4个并发场景的实验测试验证了该系统的可靠性与稳定性。

1. 引言

随着互联网的高速发展，网络研究正面临一个关键问题：难以在现实网络环境中安全可靠地测试新开发的功能[1, 2]。现有的测试设施限制了开发、测试和培训的时间与资源成本，在安全领域，面对现实世界威胁的快速演变时，这一点显得尤为不利[3]。为了解决这一问题，网络测试床被设计为一个提供与其他网络相隔离的、代表真实世界场景的测试台，提供实践

包括渗透测试、网络保护、攻击应对等网络操作的环境，同时支持网络安全产品的实验和测试[4]。

在网络测试床中，场景网络位于最上层应用层，被用户直接使用，其真实性对用户体验起重要作用；背景流量既是场景节点可达性的表述者，同时也是场景模拟环境的氛围制造者，是场景真实性的重要表现。本文是对测试床中网络场景背景流量管理系统设计的一次有意义的实践。

2．背景流量问题分析

背景流量的管理主要包括流量生成与流量投放两个部分。

一方面，网络测试床允许多个并发网络场景分别独立工作，而同一场景内也可能存在多个相互独立的子网。重放流量时，需要与每个场景乃至每个子网分别连接以提供用于重放的流量包，同时能保证所有的场景与子网不会由此而被联通。因此，流量投放模块的主要任务是在每个场景乃至每个独立子网中进行背景流量的重放，同时保持各自网络的隔离性。

另一方面，网络场景中的每个节点都会分配对应的节点属性，用于模拟现实网络中不同的网络设备，如FTP服务器、数据库服务器或普通网络用户等等。由于测试场景多变复杂，在实际网络中录制获得的真实流量难以与场景中的节点一一对应，直接重放的效果会近似使用测试仪等工具生成的无意义流量，失去了重放真实流量的优势。因此，流量生成模块的主要任务是根据场景中节点的属性将真实流量重组为合适的背景流量，使得场景更为贴近现实。

3．背景流量管理系统的设计与实现

3.1　流量投放

网络测试床允许多个并发网络场景分别独立工作。不同的场景相互隔离，同一场景内也可能存在多个相互独立的子网，需要分别投放背景流量。而背景流量模块位于平台层，所有的流量投放都须接受统一管理。为避免背景流量系统将隔离网络串通，需要将背景流量模块的控制流与数据流分离。

本系统流量投放模块逻辑网络配置如图1：

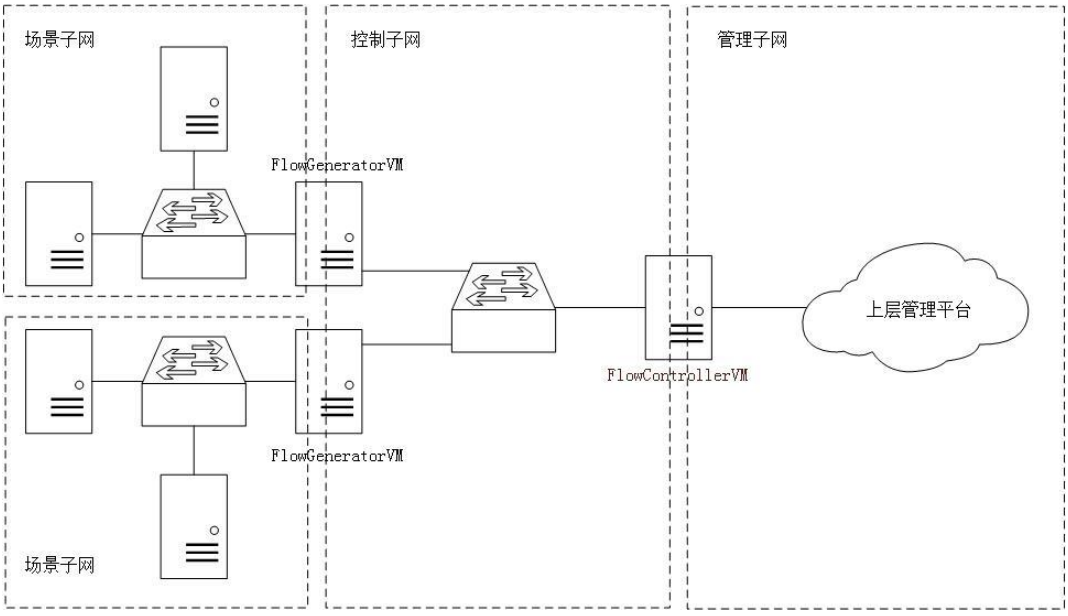

图1 背景流量子系统逻辑网络配置图

每个场景分配一个流量控制器Flow Controller VM，并为每个子网生成对应的流量投放器Flow Generator VM；每个Flow Controller VM与其所拥有的Flow Generator VM组成控制子网，

并进行任务协调；每个Flow Generator VM单向接入场景子网中，实现各个隔离子网中背景流量别分重放；管理平台与所有Flow Controller VM相连，组成管理子网下发流量任务，从而在保持场景隔离的前提下实现流量任务的统一管理。

其物理网络配置如图2：

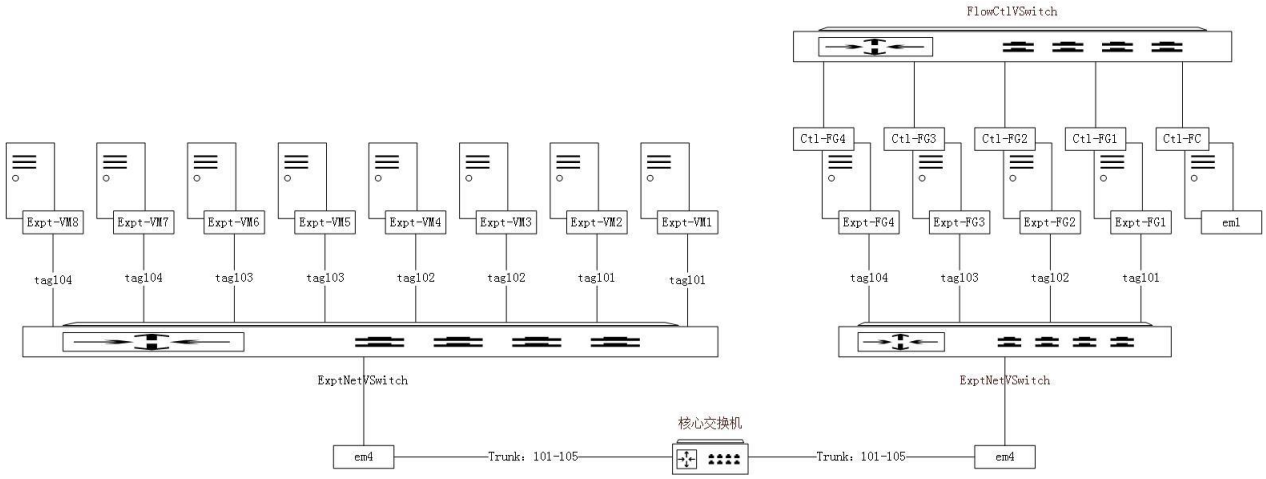

图2 背景流量子系统物理网络配置图

每台Flow Controller VM及其所属的Flow Generator VM均配有一张虚拟网卡作为控制口，通过虚拟交换机OpenVSwitch相互连接形成控制子网，对外隔离，实现Flow Controller VM和Flow Generator VM控制流交互；每台Flow Generator VM配有第二张虚拟网卡作为业务口，通过OVS与实际场景相连，在OVS上被分配Vlan号，从而归属指定子网，与其他场景隔离；每台Flow Controller VM配有第二张虚拟网卡作为管理口，与上级管理模块相连，接受上层发送的流量任务，并分配至对应Flow Generator VM。

3.2 流量生成

从测试角度看，真实网络环境下的网络流量拥有极高的真实性和复杂性，远好于测试仪表生成的模拟流量。但高度的复杂性也成为直接录制重放真实流量在网络测试床中应用的巨大障碍——多变的测试要求很难每次都找到相符的现实网络去录制流量。对录制好的真实流量进行分析重组，使其满足对应场景的使用需求，就是流量生成模块的主要功能。

本系统流量生成模块的实现方式如图3。流量生成模块从上层获得流量任务，根据任务描述生成对应的流量时序表，并根据时序表从预先抓取的真实流量pcap包中分析选取对应的会话流量进行重组，生成新的pcap包，交付流量投放模块。

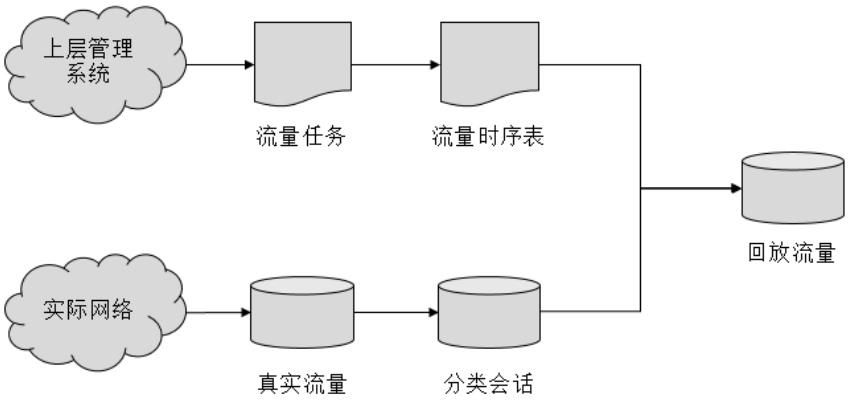

图3 流量生成模块流程图

首先，流量任务以json文件的形式下发到流量生成模块，主要描述任务的起止时间与场景的节点属性。根据节点类型的不同，以预设的规则为每个节点分配流量会话类型时序。将场景中所有节点的会话类型时序汇总，生成场景的流量时序表。

接着，流量生成模块对预先抓取的真实流量进行会话分析，根据各个协议的特征拆分流量，将流量数据包分组形成一组组端到端的流量会话。目前实现的会话流量分析重组主要基于TCP协议[7, 8]，部分特征如表1所示。首先根据TCP协议三次握手四次挥手特征分割TCP连接形成会话[9, 10]，然后根据各协议的端口号和特征关键字将会话分类，作为候补流量。

表1 流量会话分析依据

协议类型	端口号	特征关键字
HTTP[11]	80	请求包：GET, POST, PUT, HEAD, DELETE, TRACE, OPTIONS, CONNECT和PATCH 应答包：http版本、状态码
FTP[12]	20、21	FTP 命令：USER、PASS、SIZE、REST、CWD、RETR、PASV、PORT、QUIT FTP 响应码：三位数字响应码
SSH[13]	22	版本协商：SSH-ssh协议版本-详细版本\r\n

最后，流量生成模块根据已经生成的流量时序表，在对应的候补流量中随机选择会话进行填充，组成投放流量包，交付至流量投放模块。

4. 系统测试

实验测试环境依托于实验室内网络测试床搭建的网络场景。有四个测试场景在背景流量系统下并行运行。作为样例，其中一个网络场景包括7台网关、37台交换机、各类服务节点与用户节点总计37个终端设备，为其分配1台Flow Controller VM和4台Flow Generator VM。

该测试场景某时刻整体的流量统计如图4。测试中，Flow Controller VM可以正确接收上层下发的流量任务，并拆解分发给4台Flow Generator VM。4台Flow Generator VM共计重组生成2GB的流量pcap文件，在该场景中可以重放约100分钟，并自动进行循环播放。

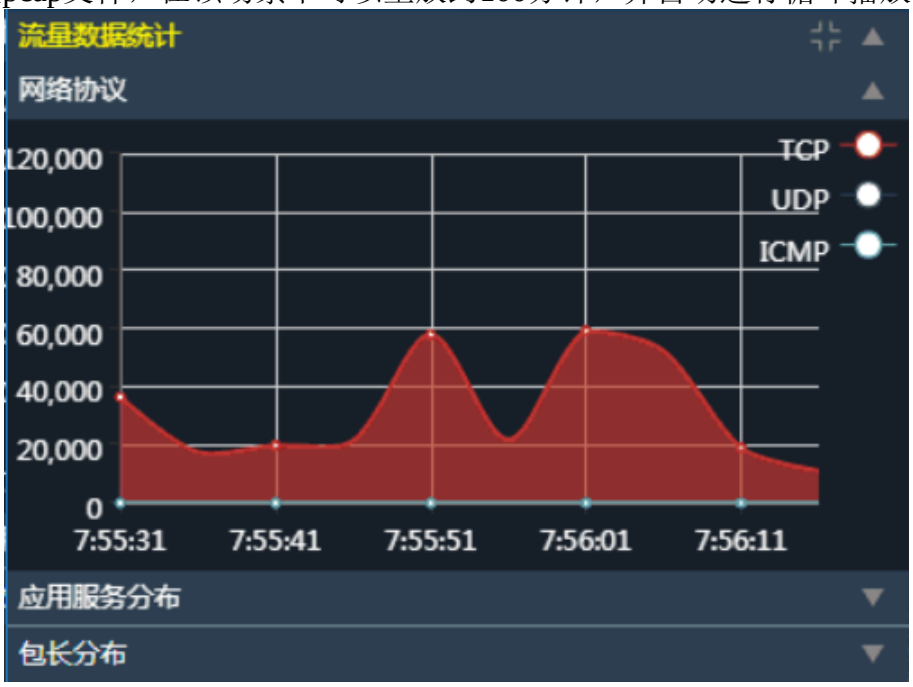

图4 场景中某时刻流量统计

5．结束语

本文提出了一种背景流量管理系统方案，实现了网络测试床环境下网络场景的背景流量生成工作，向上接收管理系统发送的背景流量任务，解决了多场景同时运行时背景流量分别重放并保持场景与场景之间、子网与子网之间网络隔离的问题，并通过分析真实流量重组会话，根据场景节点属性进行会话填充来提高了场景中背景流量的真实性。从测试结果来看，本系统达到了预期的效果，是可行有效的。

致谢

本论文获得中国科学院网络测评技术重点实验室和网络安全防护技术北京市重点实验室资助。获得了国家重点研发计划2016YFB0801004、2016QY04W0905、2016QY071405课题资助。

References

[1] Ranka, Jinendra. National Cyber Range. DEFENSE ADVANCED RESEARCH PROJECTS AGENCY ARLINGTON VA STRATEGIC TECHNOLOGY OFFICE (STO), 2011.

[2] Davis, Jon, and Shane Magrath. A survey of cyber ranges and testbeds. No. DSTO-GD-0771. DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION EDINBURGH (AUSTRALIA) CYBER AND ELECTRONIC WARFARE DIV, 2013.Adleman L M, An Abstract theory of computer viruses, 8th Annual International Cryptology Conference, Santa Barbara, California, USA, 1988

[3] A. P. Hansen, "Cyber Flag: A Realistic Cyberspace Training Construct," DTIC Document 2008.

[4] M. Varshney, K. Pickett, and R. Bagrodia, "A Live-Virtual-Constructive (LVC) framework for cyber operations test, evaluation and training," MILCOM 2011, pp. 1387-1392.

[5] L. Pridmore, P. Lardieri, and R. Hollister, "National Cyber Range (NCR) automated test tools: Implications and application to network-centric support tools," Autotestcon, IEEE, pp. 1-4.

[6] Jing Zhang, Research and Application of Network Protocol Analysis Technology, Central South University, 2012

[7] Jon Postel, Internet Protocol RFC 791 [EB/OL], http://www.faqs.org/rfcs/rfc791.html, 1981

[8] Jon Postel, User Datagram Protocol RFC 768 [EB/OL], http://www.ietf.org/rfcs/rfc768.txt, 1980

[9] Jon Postel, Transmission Control Protocol RFC 793 [EB/OL], http://www.ietf.org/rfcs/rfc793.txt, 1981

[10] Wolf T, You S, Ramaswamy R, Transparent TCP acceleration, Computer Communications, 2009, 32(4): 691-702.

[11] T.Berners-Lee, R.Fielding, H.Frystyk. Hypertext Transfer Protocol-HTTP/1.0. RFC 1945 [EB/OL]. http://www.ietf.org/rfc /rfc1945.txt, 1996.

[12] Jon Postel, J.Reynolds. File Transfer Protocol. RFC 959 [EB/OL], http://www.ietf.org/rfc/rfc959.txt, 1985.

[13] P.Mockapetris. Domain Name Service, RFC 1034 [EB/OL], http://www.ietf.org/rfc/rfc1034.txt, 1987.

[14] Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76.

[15] Nunes, Bruno Astuto A., et al. "A survey of software-defined networking: Past, present, and future of programmable networks." IEEE Communications Surveys & Tutorials 16.3 (2014): 1617-1634.