

基于收发平衡判定的 TCP 流量回放方法

褚伟波¹⁾ 蔡忠闽¹⁾ 管晓宏^{1),2)} 陈明旭¹⁾

¹⁾(西安交通大学智能网络与网络安全教育部重点实验室、机械制造系统工程国家重点实验室 西安 710049)

²⁾(清华大学自动化系信息科学与技术国家实验室 北京 100084)

摘 要 基于真实网络流量的互动式回放测试是当前针对防火墙、IPS 等串接型安全设备进行测评的最新方法. 文中在分析现有基于状态判定的 TCP 流量互动式回放方法基础之上, 引入收发平衡机制, 提出了一种基于收发平衡和状态判定相结合的新的 TCP 流量回放方法. 通过在发送 TCP 数据包前优先进行收发平衡判定将数据包发送出去, 提出的方法能够有效减少 TCP 流量在发送过程中的状态判定开销, 提高回放性能. 对引入收发平衡机制前后的 TCP 流量回放方法的差异进行了分析比较. 从单个 TCP 会话特性、并发会话流量特性、网络传输延迟与丢包等角度分析验证了影响引入收发平衡机制后的算法有效性的因素. 实际流量实验表明, 文中所提方法在回放 TCP 流量时性能有显著提升, 适用于在更大规模的流量环境下对防火墙、IPS 等串接型网络安全设备进行测评.

关键词 网络安全设备测评; 流量回放; 互动式 TCP 流量回放; 状态判定; 收发平衡

中图法分类号 TP393

DOI号: 10.3724/SP.J.1016.2009.00835

A New Method for Interactive TCP Traffic Replay Based on Balance-Checking Between Transmitted and Received Packets

CHU Wei-Bo¹⁾ CAI Zhong-Min¹⁾ GUAN Xiao-Hong^{1),2)} CHEN Ming-Xu¹⁾

¹⁾(MOE Key Laboratory for Intelligent Networks and Network Security, State Key Laboratory for Manufacturing Systems, Xi'an Jiaotong University, Xi'an 710049)

²⁾(Tsinghua National Laboratory for Information Science and Technology, Department of Automation, Tsinghua University, Beijing 100084)

Abstract Interactive network traffic replay is the newest method for testing and evaluation of network devices such as Firewalls, IPSes, routers, switches, etc. Currently state-checking method is used for interactive TCP traffic replay. This paper proposes a new method for interactive TCP traffic replay which is based on the balance status between transmitted and received packets. By checking the balance conditions before sending out TCP packets, the method can significantly reduce the cost of state-checking and enhance the replay performance. The authors made a comparison on the differences of replay methods when introducing the balance mechanism. The efficiency of the method is also investigated and evaluated from aspects of a single TCP session, multi-session traffic, packet losses and latency. Experimental results show that the method outperforms the original state-checking method when replaying actual TCP traffics.

Keywords testing and evaluation of network security devices; network traffic replay; interactive TCP traffic replay; state based method; balance checking

收稿日期: 2008-12-08; 最终修改稿收到日期: 2009-01-19. 本课题得到国家自然科学基金(60574087)、国家“八六三”高技术研究发展计划项目基金(2007AA01Z464, 2007AA01Z475, 2007AA01Z480, 2008AA01Z415)、教育部博士点基金(20070698107)、陕西省自然科学基金(2006F46)、西安市科技计划(zx06026)资助. 褚伟波, 男, 1982年生, 博士研究生, 主要研究方向为网络流量回放和安全设备测试. E-mail: wbchu@sei.xjtu.edu.cn. 蔡忠闽(通信作者), 男, 1975年生, 博士, 教授, 主要研究方向为计算机网络安全、数据挖掘和模式识别. E-mail: zmcai@sei.xjtu.edu.cn. 管晓宏, 男, 1955年生, 博士, 教授, 博士生导师, 主要研究领域为计算机信息安全、系统优化与调度. 陈明旭, 男, 1985年生, 硕士研究生, 主要研究方向为网络场景仿真与再现.

1 引 言

当前对网络防火墙、入侵阻断系统(IPS)等串接型安全设备所用的测评方法主要有手工仿真测试和真实环境接入运行两种. 手工仿真测试是指用 SmartBits^①、协议分析仪^②、包仿真器或自行建造的流量仿真发生器^[1-5]产生测试需要的仿真流量对网络设备进行测评. 手工仿真测试需要的专用设备价格昂贵,且由于实际网络环境复杂性远非手工仿真环境所能比拟,手工仿真测试的结果并不能完全反映出安全设备在真实环境中的性能. 通过仿真流量测试的设备还需要接入真实网络环境运行来考察设备在实际环境下的安全性能. 由于待测设备的不稳定会对实际网络运营造成影响,网络运营商一般都不愿意参与产品测试,而且真实网络中测试环境不

可控,测试场景难以再现,发现问题之后的定位也非常困难.

针对流量仿真测试和实际环境接入测试两种方法的不足,研究人员提出了互动式流量回放测评方法^[6]. 这种方法事先从真实网络中将流量数据捕获记录,通过专用的回放系统回放到测试网络中来再现流量采集点处的网络情形,并对接入的安全设备进行测试. 相比一般的流量回放技术^{③④[7]},互动式的流量回放技术在产生流量的过程中考虑了设备的作用. 采用互动式回放方法的最大优点在于能够模拟流量采集点两侧的行为,实现流量与安全设备之间的动态交互过程. 同时,采用该种方法的测试过程可控,测试场景可再现,对具体问题的定位也非常方便. 因此,互动式的流量回放方法特别适合于 IPS、防火墙等串接型安全设备的测评.

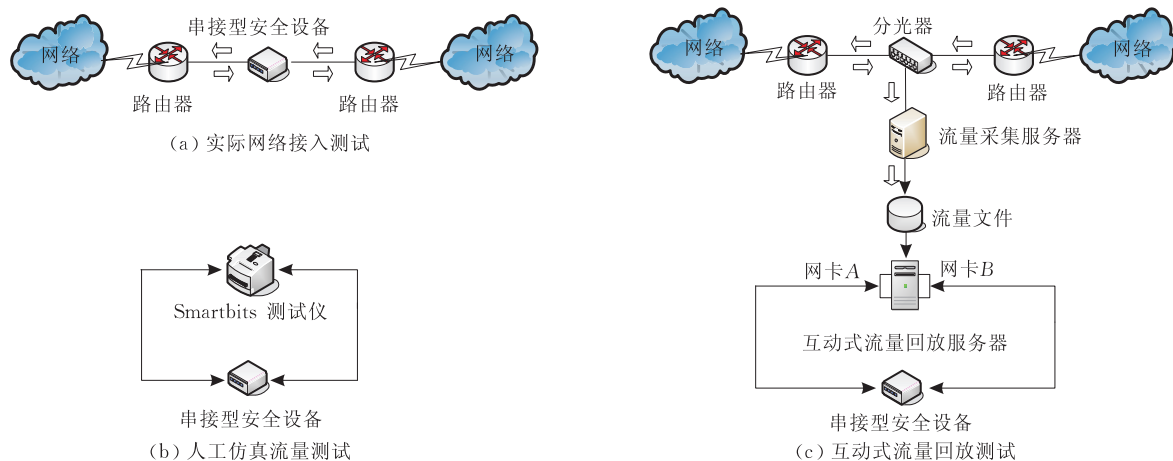


图 1 串接型网络安全设备测试的 3 种方法

当前的互动式流量回放技术主要处理 TCP 协议流量,采用基于状态判定的回放方法^[6]. 流量回放系统依次从 TCP 流量文件中读入并回放 TCP 数据包. 在回放过程中,系统模拟实现 TCP/IP 协议栈,为回放的 TCP 会话双方维护一组状态. 数据包严格遵照 TCP 协议规则进行发送. 待回放的数据包只有通过状态判定时才被系统发送出去. 数据包在通过被测设备并到达系统的另外一个测试端时被接收,系统根据接收到的数据包信息及时更新对应会话的状态. 伴随着数据包的发送和接收,会话状态被不断更新,从而推动整个流量回放过程. 回放系统通过回放后的流量信息来对被测设备进行评估.

采用基于状态判定的回放方法产生的流量虽然严格符合 TCP 协议,能够完全实现流量与设备之间的真实互动. 然而回放过程中需要模拟实现 TCP/IP

协议栈,回放过程开销较大,这使得方法的实时处理能力受到限制,不适合模拟高速网络环境下的大规模并发 TCP 通信行为. 另外,即便在无中间设备接入的情况下,严格按照 TCP 协议规则产生的流量可能会与被采集回放的流量不一致,因此并不能完全真实再现采集点处的流量情形.

本文在分析现有基于状态判定的 TCP 流量回放方法基础之上,提出了一种基于收发平衡和状态判定相结合的新的 TCP 流量互动式回放方法. 通过在发送 TCP 数据包之前优先进行收发平衡条件的

- ① Spirent SmartBits. <http://www.spirent.com/analysis/technology.cfm?media=7&ws=325&ss=110>. 2008
- ② Sniffer® + nGenius® Products. <http://www.netscout.com/products>. December 2008
- ③ The Tomahawk Test Tool homepage. <http://tomahawk.sourceforge.net/>. December 2008
- ④ The TCPREPLAY & FLOWREPLAY homepage. <http://tcpreplay.synfin.net/trac/>. December 2008

判定,可以将满足收发平衡条件的绝大多数数据包发送出去,有效减少 TCP 流量在发送过程中的状态判定开销,提高回放性能。

本文第 2 节对基于互动式流量回放的网络安全设备测评方法进行介绍;第 3 节介绍 TCP 流量回放过程中的收发平衡现象,对引入收发平衡机制前后的回放方法进行分析对比;第 4 节给出完整的基于收发平衡和状态判定相结合的 TCP 流量回放算法;第 5 节从单个 TCP 会话、并发会话、回放过程中的传输延迟和丢包等角度对影响算法效率的因素进行分析;第 6 节通过实验验证影响算法效率的因素,并考察算法在回放实际流量时的性能提升效果;第 7 节对全文工作进行小结。

2 基于互动式流量回放的网络安全设备测评方法

当前对网络安全设备的测评主要分为性能测试和安全性测评。性能测试主要考察设备在网络流量环境下的吞吐率、丢包率、背靠背延迟等各种指标,通常利用 SmartBits 等流量发生仪器产生人工设定的流量来冲击设备进行测试;对设备的安全性测评主要考察设备对于异常流量进行阻断控制的各项安全性指标,包括阻断成功率、阻断效率、误阻率、漏阻率、阻断响应速度等,一般采用人工构造攻击脚本、攻击场景和攻击流量等方法进行测试。

对网络安全设备常用的安全性指标的定义和计算方法如下。

定义 1. 称防火墙或者入侵阻断系统等网络安全设备对于包含异常流量及入侵行为的会话实施成功阻断的数目占所有异常会话发生数目的比率为阻断成功率。该指标的计算公式如下：

$$P_{\text{block}} = S_{\text{abnormal_block}} / S_{\text{abnormal}} \times 100\%,$$
其中, $S_{\text{abnormal_block}}$ 为被成功阻断的异常会话的数目, S_{abnormal} 为异常会话发生的总数。

定义 2. 称防火墙或者入侵阻断系统等网络安全设备在某类异常流量冲击下实施阻断控制时所阻断的异常流量大小占该类异常流量总数的百分比为安全设备对该类异常或攻击的阻断效率。该指标的计算公式如下：

$$P_{\text{block_position}} = (1 - F_{\text{abnormal_transmit}} / F_{\text{abnormal}}) \times 100\%,$$
其中, $F_{\text{abnormal_transmit}}$ 为成功传输的异常流量大小, F_{abnormal} 为测试流量中包含的该类异常流量总数。

定义 3. 称防火墙或者入侵阻断系统等安全设备在异常流量冲击下本该完成阻断但并未实际完

成的异常会话数目占全部异常会话发生次数的比率为安全设备的漏阻率。漏阻率指标的计算公式如下：

$$P_{\text{miss}} = S_{\text{abnormal_miss}} / S_{\text{abnormal}} \times 100\%,$$

其中, $S_{\text{abnormal_miss}}$ 为实际遗漏的异常会话阻断数目, S_{abnormal} 为总的异常会话发生次数。称在异常流量冲击下所有被阻断的正常会话数目占全部正常会话发生次数的比率为安全设备的误阻率。误阻率指标的计算公式如下：

$$P_{\text{false_block}} = S_{\text{normal_block}} / S_{\text{normal}} \times 100\%,$$

其中, $S_{\text{normal_block}}$ 为实际被阻断的正常会话数目, S_{normal} 为总的正常会话数目。

现有方法对安全设备的性能测试和安全性测评基本是分离的,测试过程中 $S_{\text{abnormal_block}}$ 、 $S_{\text{abnormal_miss}}$ 等指标的测量很少考虑背景流量的影响,且测试过程采用了人工仿真环境,与设备实际运行的目标环境相差较大,因此,现有方法的测评结果并不完备。

基于互动式流量回放的网络安全设备测评方法则考虑在有真实背景流量的情况下,对设备安全性进行测试,同时考察设备的性能,从而对设备的安全性提出一个更为准确的界定。

如图 2 所示,基于互动式流量回放的网络安全设备测评方法在应用之前,一般需要先对从网络采集的原始流量进行分析处理,标定出正常流量和异

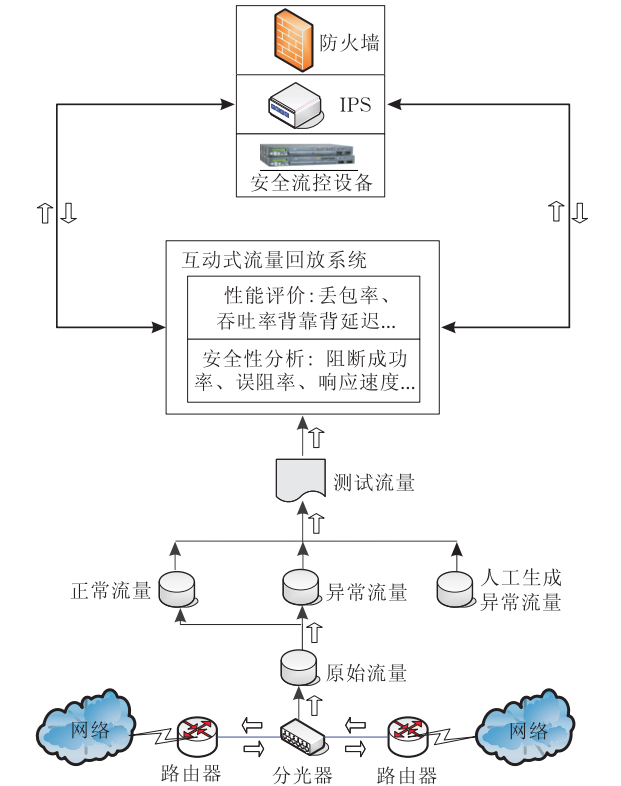


图 2 基于互动式流量回放的网络安全设备测评方法框架

常流量. 测试之前, 根据安全设备的测评目标选取相应的背景流量和测试案例. 背景流量通常从正常流量库中选取, 主要用于模拟再现目标网络环境. 而测试案例则从攻击等异常流量库中选取或是采用人工生成的异常案例. 在具体实施测试的过程中, 回放系统通过在基于真实流量的背景流量中混杂测试案例流量来冲击设备, 达到考察设备在实际背景流量下的安全性能的目的.

利用互动式流量回放技术可以方便地得到并发会话数、丢包率、吞吐率、传输延迟等各种性能指标和正常、异常会话的阻断和完成信息, 并利用这些信息进行阻断成功率、阻断效率、误阻率、漏阻率、阻断响应速度等指标的计算和安全性评价以及具体问题的分析定位. 基于互动式流量回放的方法已经开始成为网络安全设备测评的重要方法.

3 TCP 流量回放过程中的收发平衡机制

3.1 TCP 流量互动式回放过程的收发平衡

当前 TCP 协议流量的互动式回放采用基于状态判定的方法. 在 TCP 流量回放过程中, 系统会按照数据包的发送方向和接收方向, 用至少两块网卡来模拟实际通信过程中的双向收发过程: 回放系统

为每一个 TCP 会话维护一组通信状态, 并对待发送数据包的相关协议字段进行分析, 依据当前的会话状态, 参照 TCP 协议规范决定是否发送数据包; 回放系统还需要对接收到的数据包进行分析, 基于数据包包含的协议状态信息, 根据 TCP 协议规范更新相关 TCP 会话的协议状态.

例如在回放图 3(a) 所示的 TCP 会话的过程中, 回放系统用测试端 A 和 B 分别模拟会话通信端 a 和 b . 系统分析会话的第 1 个数据包, 发现是 SYN 包, 便根据 TCP 协议规则, 由测试端 A 将 1 号包发送给测试端 B, 同时测试端 A 的会话状态变为 SYN_SENT 状态. 系统接着分析会话的第 2 个数据包, 确定为 SYN-ACK 包, 应由测试端 B 发送给测试端 A, 但如果测试端 B 当前还没有接收到 1 号数据包, 测试端 B 的会话状态仍为无连接状态, 根据 TCP 协议规则, 这时 2 号包是不能发送的, 只有当测试端 B 接收到 1 号数据包时, B 的状态变为 SYN_RECV 状态, 2 号包才可以发送. 依次类推, 3 号包在 A 接收到 2 号包之后发送, 4 号包在 B 接收到 3 号包之后发送. 在连接建立后进入 ESTABLISHED 状态时的数据传输过程如图 3(b), 数据发送方接收到确认报文后推动发送窗口向前移动来发送后续数据报文. 相应地, 接收方在接收到数据报文之后发送确认报文.

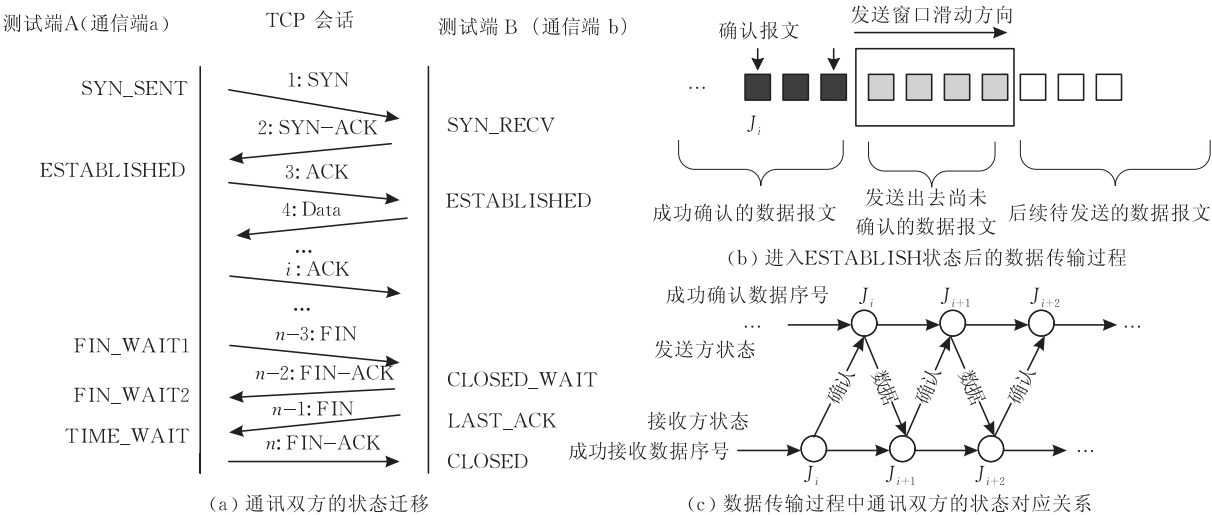


图 3 TCP 会话回放过程

这样回放完整的 TCP 会话时, 我们发现, t 时刻, 测试端 A 发送 i 号数据包时, 若此前 B 发送的数据包都已经被 A 接收到了, 即 B 发送数据包的总数 $N_{\text{Bsend}}[t]$ 与 A 接收到数据包总数 $N_{\text{Arecv}}[t]$ 相等, 则 i 号数据包可以直接发送, 而无须对当前的会话状态和 i 号包的协议字段做具体的判断, 我们称

这一现象为 TCP 流量回放过程中的收发平衡现象.

在流量回放过程中产生收发平衡现象的原因在于正常的 TCP 通信过程中, 通信双方数据包发送都是遵照 TCP 协议规范进行的, 在采集得到的实际流量中数据包顺序就已经包含了协议状态的演变过程. TCP 协议采用累积确认机制, 通信双方均以累

积发送和接收到的数据序列号来对数据传输过程进行同步控制. 所以双方通信过程中的数据传输状态也是同步的, 如图 3(c) 所示. 通信两端均需要接收到相应的数据包来推动协议状态的变更, 任意一方在进入后续状态时必然已经依序遍历过前面所有的状态, 也即接收到前面所有的对立端发送过来的推动状态变更的数据包. 又因为数据包的发送也需在协议状态满足之后进行, 所以完整的 TCP 通信过程中就包含了这种数据包的发送和接收关系, 在按序播放时则体现出收发平衡现象.

利用收发平衡现象, 可以在会话流量包含完整 TCP 通信过程的前提条件下, 简单地根据对立方向发送过来数据包的接收情况来决定当前数据包的发送.

定义 4. 对于一个回放的 TCP 会话流量, 若系统在发送当前 TCP 数据包时, 位于该数据包之前且属于该数据包相反方向的已经发送和接收到的数据包个数相等, 则称在该数据包相反方向的**收发平衡条件**满足. 这里已经发送和接收的数据包不包括在回放时由于发送端超时而重传的数据包以及回放测试过程中中间设备产生的数据包.

3.2 TCP 流量回放方法比较

利用收发平衡机制可以实现 TCP 流量的交互式回放, 具体方法如下: 系统依次从 TCP 流量文件中读入并回放 TCP 数据包. 在回放过程中, 为会话双方分别维护两个数据包收发计数器, 用于记录当前回放过程中会话两个方向发送和接收到的数据包个数. 当一个方向的数据包需要发送时, 只需判定所属会话在待发送数据包相反方向的收发平衡条件. 待发送数据包只有满足收发平衡条件时才被系统发送出去. 不满足条件的数据包被放入会话等待队列中等候发送.

基于收发平衡机制的 TCP 流量交互式回放方法实现简单, 只需要维护两个数据包收发计数器, 数

据包在发送时也仅需进行收发平衡条件判定, 回放系统对于数据包的处理开销较小, 适用于大规模 TCP 流量的回放.

基于收发平衡机制的交互式流量回放方法在回放单个 TCP 会话时, 一个方向的数据包需要等全部接收到对立方向发送的数据包时才能够被发送出去, 所以, 严格基于收发平衡条件回放的会话数据包顺序与网络捕获到的 TCP 会话数据包次序相同. 采用这种方法在单个会话的数据包顺序层面上可以完全再现流量采集点处的情形.

然而, 基于收发平衡机制的交互式流量回放方法在会话回放过程中任意时刻只有一个方向的数据包在传输, 这与 TCP 会话双向传输的特点不符.

为了克服采用收发平衡机制带来的会话在任意时刻只能单向传输数据包的缺点, 提出一种基于收发平衡与状态判定相结合的 TCP 流量交互式回放方法. 这种方法实现如下: 系统依次从 TCP 流量文件中读入并回放 TCP 数据包. 在回放过程中, 系统同时为回放的 TCP 会话双方维护一组状态和数据包收发计数器. 数据包在发送时首先进行收发平衡条件判定. 若收发平衡条件满足, 则将数据包发送出去; 若收发平衡条件不满足, 再采用基于状态判定的方法来发送数据包.

由于将收发平衡机制和状态判定两种方法用于数据包的发送, 基于收发平衡和状态判定相结合的 TCP 流量回放方法产生的数据包顺序也不一定严格与被回放会话的数据包次序相同.

相比较基于状态判定的方法, 引入收发平衡机制后的 TCP 流量回放方法可以通过优先进行收发平衡判定来发送数据包, 减少回放系统在数据包发送这一环节上的开销, 提高回放效率. 同时, 对于单个 TCP 会话, 回放过程保留 TCP 双向传输的特点.

基于状态判定、基于收发平衡以及两者相结合的 TCP 流量回放方法的比较如表 1.

表 1 3 种不同 TCP 流量交互式回放方法的比较

流量回放方法	属性				
	数据包发送依据	是否维护会话状态	单会话回放时单向/双向传输	是否严格遵守 TCP 协议规范	是否和被回放会话的数据包次序相同
基于状态判定	待发送数据包和回放会话的状态信息	是	双向	严格遵守	不一定相同
基于收发平衡	待发数据包相反方向的收发平衡条件	否	单向	不一定严格遵守	相同
基于收发平衡和状态判定相结合	先根据收发平衡条件, 在收发平衡无法发送时再采用状态判定的方法	是	双向	不一定严格遵守	不一定相同

3.3 数据包发送依赖关系上的一致性

虽然引入收发平衡机制前后的流量回放方法在产生数据包顺序上有所差异,但由于被回放的流量信息是实际 TCP 通信过程产生的,因此,在按序回放时无论采用何种回放方法均需要再现 TCP 通信双方状态的演变过程. TCP 通信状态的变更由数据包的接收和发送推动,回放过程应严格再现这些数据包的发送和接收关系.

定义 5. 任意一个 TCP 数据包 p_i 的不同方向发送依赖关系可以定义成为与其相关的方向相反的一组数据包 $R(p_i)$, 使得数据包 p_i 在发送时必须先接收到 $R(p_i)$. 对于确认报文, 它的相反方向发送依赖关系就是该确认报文对应的对立端发送过来的数据报文. 对于数据报文, 它的相反方向发送依赖关系就是对立端发送过来的确认报文^[8]. 对于 $\forall p_j \in R(p_i)$, 有关系 $\langle p_j, p_i \rangle$ 成立, 表示 p_j 与 p_i 方向相反, 且 p_i 需要接收到 p_j 之后才能够发送. TCP 会话的不同方向数据包发送依赖关系可用这样一系列按序出现的二元组来表示.

命题 1. 从网络捕获到的 TCP 数据包序列与实际 TCP 会话在不同方向数据包发送依赖关系上保持一致.

证明. 假设从网络捕获到的 TCP 流量包含完整的 TCP 通信过程.

对于任何一个属于 TCP 会话的不同方向数据包发送依赖关系的二元组 $\langle p_i, p_j \rangle$, 设 t_i 和 t_j 分别是 p_i 和 p_j 的发送时间, t_{trans} 为 p_i 由发送端传输到另一端需要的时间, 显然 $t_j \geq t_i + t_{\text{trans}}$. 记 $t'_i = t_i + \Delta t_i$ 为数据包 p_i 被捕获到的时间, $t'_j = t_j + \Delta t_j$ 为数据包 p_j 被捕获到的时间. 由于 $t'_i = t_i + \Delta t_i \leq t_i + t_{\text{trans}} \leq t_j \leq t_j + \Delta t_j = t'_j$, 所以 p_i 数据包比 p_j 数据包先被捕获到, 因此 $\langle p_i, p_j \rangle$ 仍满足. 由 $\langle p_i, p_j \rangle$ 的任意性可知, 网络捕获到的数据包序列与实际 TCP 会话在不同方向数据包发送依赖关系保持一致. 证毕.

推论 1. 由于基于收发平衡机制发送的 TCP 会话数据包次序与采集得到的次序完全相同, 所以结合命题 1 可得, 基于收发平衡机制发送的 TCP 数据包序列与实际 TCP 会话在不同方向数据包发送依赖关系上保持一致.

命题 2. 采用基于收发平衡和状态判定相结合的方法发送的 TCP 会话数据包序列与实际 TCP 会话在不同方向数据包发送关系上保持一致.

证明. 当采用基于收发平衡和状态判定相结合的方法时, 对于任意一个属于 TCP 会话不同方向

数据包发送依赖关系的二元组 $\langle p_i, p_j \rangle$, 若数据包 p_j 通过收发平衡发送, 则由推论 1 可知, $\langle p_i, p_j \rangle$ 满足. 若数据包 p_j 通过状态判定发送, 则自然满足 $\langle p_i, p_j \rangle$. 所以, 无论通过何种途径发送, $\langle p_i, p_j \rangle$ 仍旧满足. 由 $\langle p_i, p_j \rangle$ 的任意性可知, 采用基于收发平衡和状态判定相结合的方法发送的 TCP 会话数据包序列与实际 TCP 会话在不同方向数据包发送依赖关系上保持一致. 证毕.

4 基于收发平衡和状态判定相结合的 TCP 流量回放算法

通过上面的分析证明可知, 基于收发平衡和状态判定相结合的 TCP 流量回放方法可以在提高效率的同时, 确保产生流量与实际 TCP 会话在不同方向数据包发送依赖关系上的一致性. 因此, 可以作为实际的 TCP 流量交互式回放技术使用.

基于收发平衡和状态判定相结合的 TCP 流量回放算法完整描述如下.

算法 1. 基于收发平衡和状态判定相结合的 TCP 流量回放算法.

TCP 数据包的发送过程:

1. 从输入流量记录中读入一个 TCP 数据包 P .
2. 提取数据包 P 的(源 IP, 目的 IP)、(源端口, 目的端口), 组成四元组, 根据 Hash 算法定位所属会话 S_P .
3. 判断数据包 P 的收发平衡条件是否满足. 如果满足, 转至步 5; 否则, 执行下一步.
4. 提取数据包 P 的相关字段信息和会话的状态信息, 根据状态判定数据包 P 是否可以发送. 如果能够发送, 则转至步 5; 否则将该数据包置于会话 S_P 的等待数据包队列中等待发送.
5. 根据数据包 P 的方向, 调用对应网卡将 P 发送出去, 并把 P 放入接收对比队列; 如果 P 为数据报文, 则将 P 放入系统重传队列; 更新 S_P 的状态, 递增数据包 P 方向的当前已发送的数据包个数.
6. 继续从输入流量记录中读入下一个 TCP 数据包 P , 执行发送过程. 直到流量记录中的数据包回放完毕.

TCP 数据包的接收更新过程:

1. 从系统的两块回放网卡接收缓冲区中接收一个 TCP 数据包 P .
2. 提取数据包 P 的(源 IP, 目的 IP)、(源端口, 目的端口), 组成四元组, 根据哈希算法定位所属会话 S_P .
3. 提取数据包 P 的相关字段信息, 根据 TCP 协议规范更新 S_P 的状态. 若数据包 P 在接收对比队列中出现, 则递增会话数据包 P 方向的当前已接收到的数据包个数; 若 P 为数据确认报文, 则将 P 确认的数据报文从系统重传队列中

删除。

4. 调用数据包的发送过程,尝试发送属于 S_P 的等待数据包队列中的数据包。

5. 重传系统重传队列中超时的已被系统发送出去但尚未接收到确认的数据报文。

6. 继续从系统的两块回放网卡接收缓冲区中接收 TCP 数据包,执行接收更新过程,直到没有从任何一个网卡中接收到数据包为止^①。

记 P 为任意 TCP 数据包, S_P 为 P 所属的会话, a 与 b 为会话的两个通信端, Cs_{ab} 与 Cs_{ba} 分别记录了回放系统已经发送的由 a 到 b 方向和由 b 到 a 方向属于会话 S_P 的数据包个数; Cr_{ab} 与 Cr_{ba} 分别记录了系统已经接收到的由 a 到 b 方向和由 b 到 a 方向属于会话 S_P 的数据包个数。

算法中数据包 P 能够发送的收发平衡条件是指以下两个条件同时成立:

(1) 属于会话 S_P 的数据包中位于 P 之前的所有数据包均已发送;

(2) 如果 P 为 a 到 b 方向的数据包, 必须有 $Cs_{ba} = Cr_{ba}$ 成立; 如果 P 为 b 到 a 方向的数据包, 必须有 $Cs_{ab} = Cr_{ab}$ 成立。

算法中将发送出去的数据报文放入系统重传队列是为了在发送端超时的情况下将已发送的数据报文进行重传。而设置接收对比队列的目的是为了识别接收到的数据包是否属于回放的原始采集流量 (IPS 等设备可能发送阻断数据包) 以及区分是否为系统超时重传的数据包, 从而确保收发平衡机制的正常运行。

在 TCP 流量回放过程中引入收发平衡机制后的另外一个优点就在于系统能够回放由于数据包缺失而状态不全的不完整会话。当采用基于状态的回放方法时, 会话数据包的缺失导致状态不全, 会话流量的回放将被阻塞。而引入收发平衡机制后可以通过收发平衡判定将状态判定无法发送的数据包发送出去, 推动回放过程, 从而回放整个会话。

5 算法性能分析

5.1 算法效率评价指标

引入收发平衡机制后, TCP 数据包可通过收发平衡发送来提高流量回放效率。在回放过程中满足收发平衡条件发送的数据包越多, 算法的效率提升就越高。因此, 可认为通过收发平衡判定发送的数据包比例为衡量算法效率的指标, 相应地, 那些影响收发平衡条件的因素就是影响算法效率的因素。

5.2 单会话流量特性影响

在单个会话中, 相反方向的数据包交替出现, 状态判定满足时收发平衡条件也满足的数据包的分布特性直接影响到算法的效率, 该分布特性由会话数据包之间的确认关系来决定。

具体地, 如果会话流量中任意连续出现的两个相反方向数据包是属于会话不同方向发送依赖关系的二元组, 且后面数据包是前面数据包的累积确认, 那么此相邻两个数据包中, 位于后面的数据包在发送时必然需要全部接收到位于前面的对立端发送过来的数据包, 此时所有数据包均在满足状态判定的同时满足收发平衡条件。

相反, 如果会话一端连续发送一批数据包, 而接收端给出的确认却不是该批数据包的累积确认, 则确认报文可以在接收到相应的数据包时便可以通过状态判定发送出来, 而无需全部接收到发送端传输过来的数据包, 如图 4 所示。TCP 会话流量中该种类型的数据包越多, 算法的效率就越差。

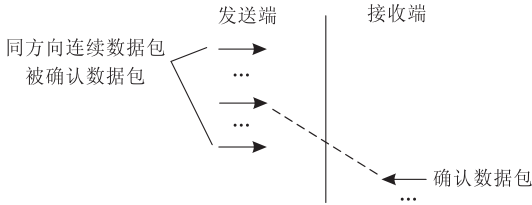


图 4 单个 TCP 会话的流量情形: 收发平衡
不满足时数据包发送的情况

5.3 并发会话流量特性影响

收发平衡条件要求待发送数据包在发送前就已经接收到同会话对立方向发送的所有数据包。设数据包从发送出去到被系统接收到所经历的时间为 T_1 , 这期间系统处理流量文件中的每个数据包所需时间为 T_2 。若在流量文件中, 当前待发送数据包与同会话对立方向发送过来的最后一个数据包之间的距离 d (如图 5) 满足 $d \geq \left\lceil \frac{T_1}{T_2} \right\rceil$ (条件 1), 则当前同会

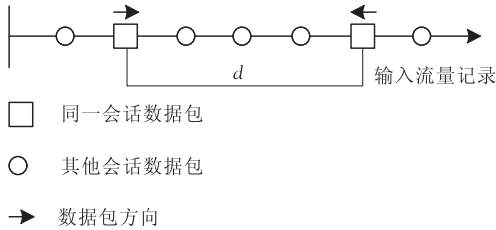


图 5 输入流量中的同一会话反方向
相邻数据包之间的距离示意图

① 采用 libpcap 作为系统的接收引擎, 修改了 Linux 下 libpcap 的源代码使其支持非阻塞的数据包接收。

话对立方方向发送过来所有数据包会在待发送数据包发送之前被系统全部接收到,待发送数据包可直接由收发平衡判定发送出去.在流量中满足条件 1 的数据包越多,算法的效率也就越高.

记 T_{read} 、 T_{send} 、 T_{recv} 分别为系统从流量文件中读入、发送和接收一个数据包所需的平均时间, T_{trans} 为数据包从回放系统一个测试端到达另外一个测试端需要的平均传输延时. 在无传输丢包的情况下, 数据包从发送出去到被系统接收到所经历的时间为 $T_1 = (T_{\text{send}} + T_{\text{trans}} + T_{\text{recv}})$. 而在此时间内, 系统处理每个数据包所需的平均时间为 $T_2 = (T_{\text{read}} + T_{\text{send}} + T_{\text{recv}})$.

这时, 条件 1 可记为 $d \geq \left[\frac{T_{\text{send}} + T_{\text{trans}} + T_{\text{recv}}}{T_{\text{read}} + T_{\text{send}} + T_{\text{recv}}} \right]$.

会话并发对于算法效率的影响主要在于其他会话数据包增大了同一会话相邻的反方向数据包之间的距离, 使得回放流量中满足条件 1 的数据包增多, 更多的数据包在发送时便已经满足收发平衡条件, 从而提高算法的效率.

TCP 协议在设计时采用了数据包的累积确认机制, 保证了基于收发平衡的算法在处理 TCP 协议流量时能够获得较好的效率. 即便 TCP 会话的流量特性较差, 输入流量往往又是多个 TCP 会话的并发, 进一步保证了算法在回放 TCP 流量时的有效性.

5.4 传输延时和丢包影响

网络传输过程中的延时和丢包等外部因素会导致数据包接收的延迟. 当数据包在回放时发生丢包, TCP 协议确保数据包发生重传. 数据包往往需要经历多于一次的传输才能到达另外一端而被系统接收. 数据包从发送出去到被系统接收到所经历的时间 T_1 被自然延长, 这使得流量中满足条件 1 的数据包数量下降, 致使算法的效率下降. 同样, 当数据包在回放过程中的传输延时 T_{trans} 被增大, 算法的效率会因为时间 T_1 的延长而下降.

在下一节实验部分, 将会具体验证上述 3 类因素对于算法效率的影响.

6 实 验

6.1 实验环境设置

流量回放系统的设计大体可以分成两类. 一类是诸如 TCPOpera^[6]、Monkey^[7] 的设计, 把整个 TCP 会话流量的回放实施在两个节点上, 每个节点负责一个方向流量的回放. 另外一类设计诸如 Tomahawk^③、TCPREPLAY^④, 整个 TCP 会话流量

的回放实施在单个节点上, 该节点一般配置两个测试接口, 每个测试接口负责一个方向流量的回放. 与多节点的回放系统相比, 单节点的回放系统可以观测整个 TCP 会话两个方向的信息. 由于需要维护会话两个方向的数据包收发状态, 本文所提出的引入收发平衡机制的回放方法需要实施在单节点的回放系统上.

利用实验来考察基于收发平衡和状态判定相结合的 TCP 流量回放算法. 实验环境如图 6. 回放服务器上的两块回放网卡与千兆交换机、Linux 双穴主机相连组成测试回路. Linux 双穴主机起数据包的转发作用, 并被用于设置不同的丢包率. 流量回放服务器配置 2×2.0 GHz Intel Xeon 处理器, 3GB RAM. 流量回放服务器和 Linux 双穴主机均配备 Intel e1000 千兆网卡, 操作系统为 Redhat 9.0 (2.4.20 的内核). 整个测试回路是一个纯千兆的环境.

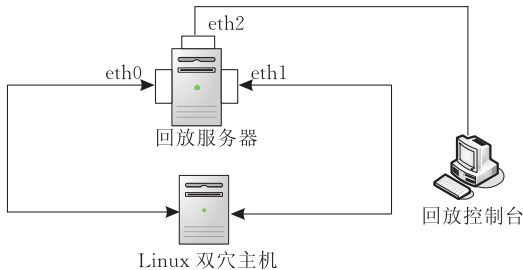


图 6 流量回放实验环境

6.2 单个会话回放实验

采用人工构造的流量来考查单个 TCP 会话的特性对于引入收发平衡机制后的回放方法的影响. 产生的 TCP 会话流量中, 一端向另外一端传输数据报文, 另一端给出确认报文. 每次在传输连续的若干数据报文之后, 接收端给出连续的等数目确认报文. 确认报文与数据报文交替出现, 每一个确认报文确认前面对应的数据报文. 确认报文中的 ACK 字段确保只有接收到对应的数据报文才能发送. 在确认报文中设置 WIN 字段大小为一个数据包, 使得数据报文的发送需要接收到确认报文中的 WIN 窗口字段通告才能够发送. 这样构造流量的目的是为了使得确认报文和数据报文都需要接收到各自对应的数据包才能发送. 图 7 是一个同方向连续数据包个数为 3 的构造流量示意图, 其中数据包之间的状态相互依赖关系已经标出.

从单会话 TCP 流量的构造方法可以知道, 在一批连续的同方向数据包中, 除了最后一个以外, 其余的数据包均可以在接收到与各自状态依赖的数据包之后便发送出去, 而无需满足收发平衡条件. 当同方

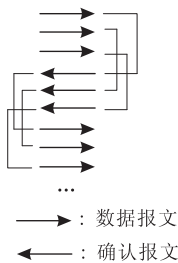


图 7 同方向连续数据包个数为 3 的构造 TCP 会话示意图

向连续数据包个数为 1 时,数据包的回放过程近似成为一个乒乓过程,数据报文和确认报文均可通过收发平衡条件发送出去.图 8 是实验中收发平衡发送的数据包比例与构造流量中同方向连续包个数之间的关系,可见同方向连续包个数越多,收发平衡发送的数据包比例就越低.图 9 是实验中引入收发平衡机制制前后回放性能的对比,可以看出,引入收发平衡机制之后单个 TCP 会话流量回放效率提高.实际

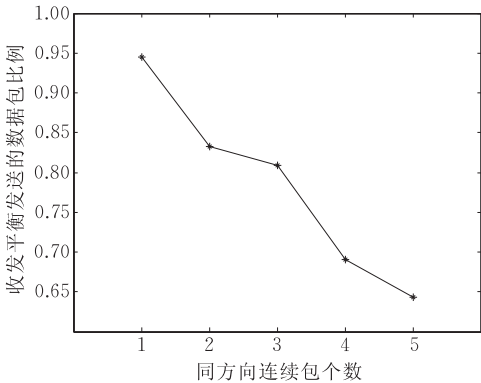


图 8 收发平衡发送的数据包比例与构造流量中同方向连续数据包个数之间的关系

192.168.1.250	192.168.1.40	FTP-DATA	FTP Data: 1176 bytes
192.168.1.250	192.168.1.40	FTP-DATA	FTP Data: 1460 bytes
192.168.1.40	192.168.1.250	TCP	5006 > ftp-data [ACK] Seq=1 Ack=9639349 win=65
192.168.1.250	192.168.1.40	FTP-DATA	FTP Data: 1460 bytes
192.168.1.250	192.168.1.40	FTP-DATA	FTP Data: 1176 bytes
192.168.1.250	192.168.1.40	FTP-DATA	FTP Data: 1460 bytes
192.168.1.40	192.168.1.250	TCP	5006 > ftp-data [ACK] Seq=1 Ack=9643445 win=65
192.168.1.250	192.168.1.40	FTP-DATA	FTP Data: 1460 bytes
192.168.1.250	192.168.1.40	FTP-DATA	FTP Data: 1176 bytes
192.168.1.250	192.168.1.40	FTP-DATA	FTP Data: 1460 bytes
192.168.1.250	192.168.1.40	FTP-DATA	FTP Data: 1460 bytes
192.168.1.40	192.168.1.250	TCP	5006 > ftp-data [ACK] Seq=1 Ack=9649001 win=65

图 10 会话 1 的流量状况

192.168.197.51	192.168.197.52	TCP	1135 > 1060 [ACK] Seq=1 Ack=16385 win=57911 Len=0
192.168.197.52	192.168.197.51	TCP	1060 > 1135 [ACK] Seq=16385 Ack=1 win=65535 Len=1460
192.168.197.52	192.168.197.51	TCP	1060 > 1135 [ACK] Seq=17845 Ack=1 win=65535 Len=1460
192.168.197.52	192.168.197.51	TCP	1060 > 1135 [ACK] Seq=19305 Ack=1 win=65535 Len=1460
192.168.197.52	192.168.197.51	TCP	1060 > 1135 [ACK] Seq=20765 Ack=1 win=65535 Len=1460
192.168.197.52	192.168.197.51	TCP	1060 > 1135 [ACK] Seq=22225 Ack=1 win=65535 Len=1460
192.168.197.52	192.168.197.51	TCP	1060 > 1135 [PSH, ACK] Seq=23685 Ack=1 win=65535 Len=892
192.168.197.52	192.168.197.51	TCP	1060 > 1135 [ACK] Seq=24577 Ack=1 win=65535 Len=1460
192.168.197.52	192.168.197.51	TCP	1060 > 1135 [ACK] Seq=26037 Ack=1 win=65535 Len=1460
192.168.197.52	192.168.197.51	TCP	1060 > 1135 [ACK] Seq=27497 Ack=1 win=65535 Len=1460
192.168.197.51	192.168.197.52	TCP	1135 > 1060 [ACK] Seq=1 Ack=19305 win=54991 Len=0

图 11 会话 2 的流量状况

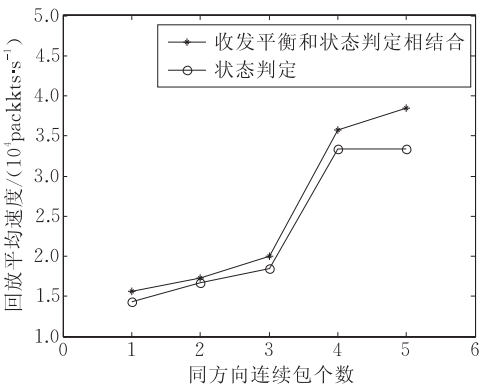


图 9 按数据包统计的引入收发平衡机制前后系统在回放构造 TCP 会话时的性能对比

的 TCP 会话中同方向连续包个数一般小于 5,因此可以预见算法在回放实际单个 TCP 会话时的性能提升.

6.3 并发会话回放实验

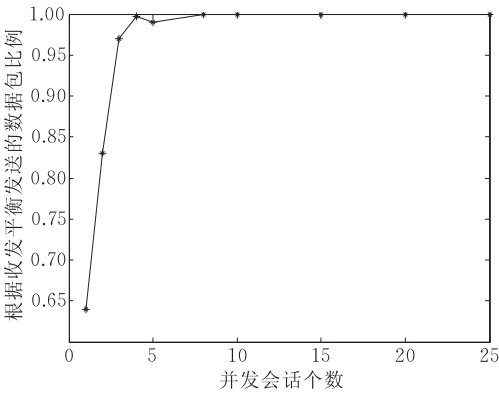
在考察会话的对于并发算法影响的实验中,引入了两个实际的 TCP 会话.其中一个会话的流量(记为会话 1)来自实验室内网的 FTP 文件传输.该会话数据包传输的网络环境通畅,基本上是发送端连续发送若干个数据包之后,接收端发送一个累积确认,如图 10 所示.

另外一个会话的流量(记为会话 2)来自两个主机的网络邻居文件传输.该会话两个对等端的处理能力相差较大,流量中大量出现确认数据包与被确认数据包之间间隔多个与被确认数据包相同方向的数据包的现象,如图 11 所示(确认包和被确认包已用底色标出).

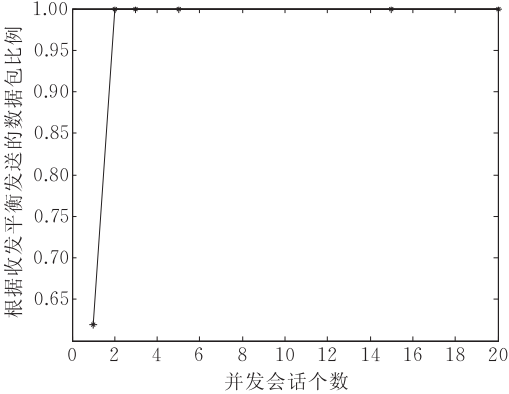
对上述两个会话的回放实验发现,会话 1 的流量中共包括 101358 个数据包,其中根据收发平衡判定发送的数据包个数为 93183 个,占总数据包数的 92%.而会话 2 的流量中共包括 17582 个数据包,其中根据收发平衡判定发送的数据包个数为 10915 个,占总数据包数的 62%.可见,会话 1 的流量特性要远远优于会话 2.

分别以会话 1 和会话 2 为基础,人工构造了并

发会话个数不同的一系列流量进行实验.实验结果如图 12 和图 13.从图 12 的结果可以看出,会话的并发会明显提升收发平衡判定发送的数据包比例.诸如会话 2 这样流量特性较差的单个会话,算法的效率也会随着并发会话个数的增大而显著增大.另外,如图 13 所示,在两个会话的实验过程中,均发现引入收发平衡判定之后算法性能显著提升.

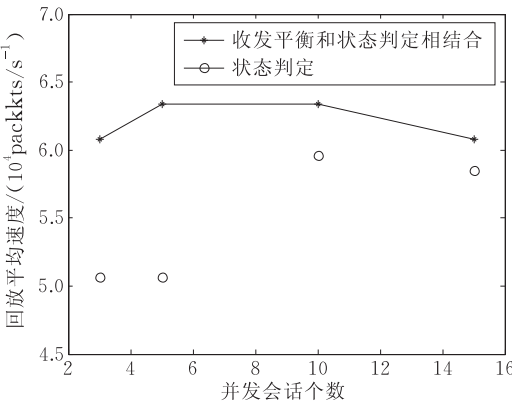


(a) 以会话1为基础的并发流量

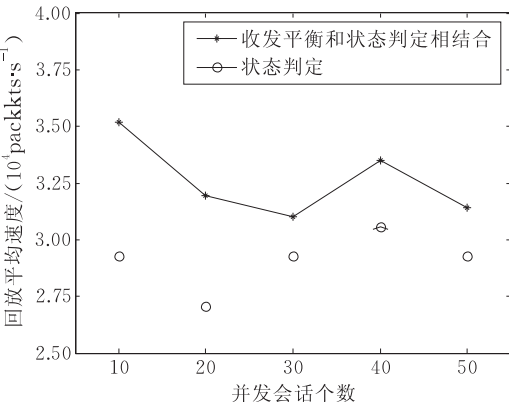


(b) 以会话2为基础的并发流量

图 12 并发会话实验中根据收发平衡判定发送的数据包比例随并发会话个数的变化曲线



(a) 以会话1为基础的并发流量



(b) 以会话2为基础的并发流量

图 13 按数据包统计的并发会话实验中引入收发平衡机制前后算法的性能对比曲线

表 2 流量回放系统的基本参数

pkt_size/Byte	$T_{read}/\mu s$	$T_{send}/\mu s$	$T_{recv}/\mu s$	$T_{trans}/\mu s$
806	13	6.5	2	41

表 2 是流量回放系统处理一个平均数据包大小为 806 字节的实际流量时的性能参数.由上一节的分析可知,当同一会话相邻数据包之间的距离(间隔其他会话数据包个数) $d \geq \left[\frac{T_{send} + T_{trans} + T_{recv}}{T_{read} + T_{send} + T_{recv}} \right] = 3$ 时,流量中几乎所有的数据包均可以被收发平衡方式发送出去.图 12 的实验结果与理论分析一致.

6.4 传输丢包和延时实验

以会话 1 为基础,并发会话数为 15 的流量进行

实验考察传输丢包和延时对于算法效率的影响.实验过程中依次在 Linux 双穴主机上设置丢包率为 0、1%和 3%.表 3 是在不同丢包率下的实验结果.实验验证了传输丢包会降低算法的效率,且丢包率越高,收发平衡发送的数据包比例越小.

表 3 不同丢包率下的算法回放效果对比

丢包率/%	收发平衡方式 发送数据包个数	状态判定方式 发送数据包个数	算法 效率/%
0	847844	1	99.99
1	706018	141827	83.2
3	554136	293709	65.36

实验中分别将回放服务器和 Linux 双穴主机的

网卡限速至 100Mbps 和 10Mbps,来考察传输延时对算法的影响,表 4 是在不同带宽环境下的实验结果.实验验证传输延时增大时算法效率下降.

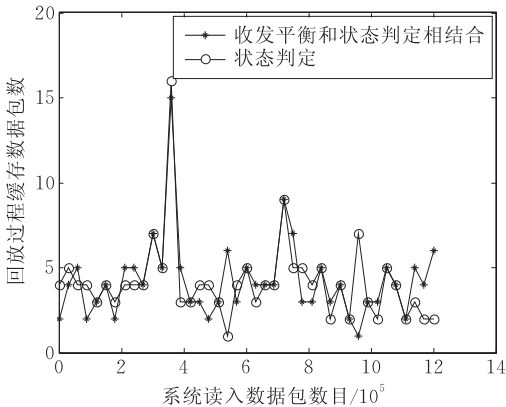
表 4 不同带宽环境下的算法回放效果对比			
回放网络 带宽/Mbps	收发平衡方式 发送数据包个数	状态判定方式 发送数据包个数	算法 效率/%
1000	847844	1	99.99
100	818369	29476	96.5
10	646057	201788	76.2

6.5 真实流量回放实验

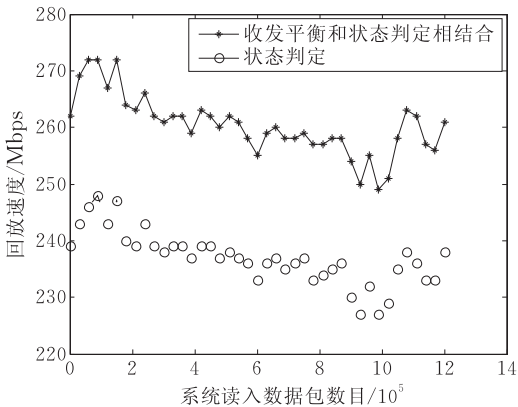
为对比引入收发平衡机制前后算法在回放实际

TCP 流量时的性能和效率,采用从真实网络中捕获的 TCP 流量记录进行实验.该记录中包含 1355385 个 TCP 数据包,共计 7695 个会话.实验中发现共有 1105675 个数据包可由收发平衡判定发送出去,149710 个数据包由状态判定发送出去.由收发平衡判定发送的数据包个数占总数的 85%以上.

引入收发平衡机制前后系统在回放实际 TCP 流量时的对比如图 14 所示.从结果可以看出,引入收发平衡判定机制后系统在回放 TCP 流量时内存需求方面与原来算法并无多大差别,但是回放性能却有显著提升.



(a) 回放过程中系统缓存数据包数对比



(b) 回放过程性能对比

图 14 基于收发平衡和状态判定相结合的回放方法和传统基于状态判定的回放方法在回放真实 TCP 流量时的对比

7 结 论

交互式流量回放技术可以用于在真实网络流量作为背景流量的情况下对设备的安全性进行测试,是一种新型的网络安全设备测评方法.本文在现有针对 TCP 流量采用的基于状态判定的回放方法基础之上,提出了一种基于收发平衡和状态判定相结合的新的 TCP 流量回放方法.通过在发送 TCP 数据包前,优先进行收发平衡判定将数据包发送出去,所提方法可以有效减少 TCP 流量在发送过程中的状态判定开销,提高回放性能.对影响算法效率的因素进行了分析与验证,并通过真实流量实验考察了引入收发平衡机制前后 TCP 流量的回放性能.实验表明,本文所提方法可以有效提升 TCP 流量交互式回放性能,适用于在更大规模的流量环境下对防火墙、IPS 等串接型网络安全设备进行测评.

参 考 文 献

[1] Danzig Peter B, Jamin Sugih. tcplib: A library of TCP inter-

network traffic characteristics. Computer Science Department, University of Southern California; Technical Report USC-CS-91-495, 1991

[2] Hong S, Wong F, Wu S Felix, Lilja B, Yohansson Tony Y, Johnson H, Nelsson A. TCPtransform: Property-oriented TCP traffic transformation//Proceedings of the GI SIG SIDAR Conference on Detection of Intrusions and Malware and Vulnerability Assessment (DIMVA). Vienna, Austria, 2005: 222-240

[3] Nicol David M, Yan Guanhua. Simulation of network traffic at coarse time-scales//Proceedings of the 19th Workshop on Parallel and Distributed Simulation (PADS'05). Monterey, CA, 2005: 141-150

[4] Sommers J, Barford P. Self-configuring network traffic generation//Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement. Taormina, Sicily, Italy, 2004: 68-81

[5] Nicol David M, Yan Guanhua. High-performance simulation of low-resolution network flows. Simulation, 2006, 82(1): 21-42

[6] Hong S, Wu S. Felix; On interactive Internet traffic replay//Proceedings of the Recent Advances in Intrusion Detection. Seattle, Washington, USA, 2005: 247-264

[7] Cheng Y, Hölzle U, Cardwell N, Savage S, Voelker Geoffrey M. Monkey see, monkey do: A Tool for TCP Tracing and Replaying//Proceedings of the USENIX Annual Technical

Conference. General Track. Boston, MA, USA, 2004: 87-98

[8] Wright Gary R, Stevens W Richard. TCP/IP Illustrated, Volume 1: The Protocols. Addison-Wesley, 1994(in Chinese)

(Wright Gary R, Stevens W Richard. TCP/IP 详解, 卷 1: 协议. 范建华等译. 机械工业出版社, 2000)



CHU Wei-Bo, born in 1982, Ph. D. candidate. His research interests include network traffic replay and network security devices testing and evaluation.

CAI Zhong-Min, born in 1975, Ph. D. , professor. His

mainly engaged in the research of network security and intrusion detection.

GUAN Xiao-Hong, born in 1955, Ph. D. , professor, Ph. D. supervisor. His research interests include network security, system optimization and scheduling.

CHEN Ming-Xu, born in 1985, M. S. candidate. His research interests include network simulation and network scene reproduction.

Background

This work is partly supported by the National High Technology Research and Development Program (863 Program) of China (grant Nos. 2007AA01Z464, 2007AA01Z475, 2007AA01Z480, 2008AA01Z415), the National Natural Science Foundation of China (grant No. 60574087), the Doctoral Fund of Ministry of Education of China (grant No. 20070698107), the Science Foundation of Shaanxi Province (grant No. 2006F46), and Xi'an Science and Technology Program (grant No. zx06026).

Testing and evaluation for network devices such as firewalls, IPSes, IDSes is of great importance in network security assurance. Recently interactive network traffic replay is proposed to test in-line devices. Current interactive traffic replay mainly deals with TCP traffic, which is replayed using state-checking method. The replay system extracts state information from every packet and compares it with the state of the corresponding TCP session to determine whether or not

to send out the packet. Since the replay system has to maintain state information for every TCP session and perform state-checking when packets are to be sent out, the replay performance is severely limited. In this paper the authors present a new method for interactive TCP traffic replay, which is based on the balance status between transmitted and received packets. By checking the balance conditions before sending out TCP packets, the method can significantly reduce the cost of state-checking and enhance the replay performance. They made a comparison of the differences of replay methods when introducing the balance mechanism. The efficiency of the method is also investigated and evaluated from perspectives of a single TCP session, multi-session traffic, packet losses and latency. Experiments show that the method outperforms the original state-checking method when replaying real TCP traffics.