基于策略分段的防火墙一致性测试

Policy – Based Segmentation Firewall Conformance Testing

邹吉丰 魏 蓉 蒋 凡 （中国科学技术大学计算机科学与技术系 合肥 230026）

摘要：防火墙是设置在被保护网络和外部网络之间的一道屏障。网络通信需要防火墙根据特定安全策略下进行监控和过滤。防火墙是否有效起到防护作用需要通过防火墙实现和安全策略配置的一致性测试。本文提出了一种改进的防火墙一致性测试方法，引入策略分段的思想，对非原子策略规则有选择的产生测试例，缩减了网络地址空间，将指数级测试例数目 O(2101rpmn2)降低为多项式级 O(Mpmn2)，减少了产生的测试例的数量，有效提高了测试效率。

关键词：防火墙测试 一致性测试 策略分段

1 引言

防火墙是设置在被保护网络和外部网络之间的一道屏障。防火墙能否起到防护作用，最根本、最有效的证明方法是对其进行测试。如何对一款防火墙进行正确测试，一直是信息安全领域非常关心的问题。

已有的更具一般性的防火墙测试方法有三种：渗透测试、防火墙实现测试、防火墙规则测试。渗透测试是为了证明网络防御按照预期计划正常运行的一种机制。通过使用脆弱性测试工具攻击防火墙以检验防火墙的安全缺陷。渗透测试通常由系统管理员或者第三方执行。防火墙实现测试方法旨在评估防火墙规则是否与防火墙执行行为相符。1 这一方法主要被防火墙销售商使用以增加产品可靠性。防火墙规则测试的目标是检验安全策略是否正确的被实现。其基本思想是将安全策略转换成防火墙规则，然后比较产生的防火墙规则和实际的防火墙规则是否相匹配，或者将防火墙规则转换成安全策略，然后比较生成的文档和给定的安全策略是否相符合，是则证明防火墙规则准确的实现了安全策略。总体来说，目前防火墙测试的相关研究基本集中在用预先定义的测试例或基于网络信息设计渗透测试。

本文对文献[1]提出的防火墙测试方法进行了改进，使得该方法的具有更高的效率。

2 防火墙测试

2.1 防火墙规则表示

根据制定的安全策略，防火墙中保存着一个访问控制列表，用以决定当数据包到来时应该采取相应的动作。该访问控制列表由许多列表项组成，每个列表项称为一条规则。每条规则又由协议类型、源 IP 地址、源端口、目的 IP 地址、目的端口组成的过滤域（又称为网络域）和动作域组成。动作域通常只有两种选择：接受，即允许数据包通过防火墙；拒绝，即不允许数据包通过。

当数据包到达防火墙时，首先查找访问控制列表的第一条规则，如果数据包的包头部分和规则的过滤域部分相匹配，则防火墙根据该规则的动作域采取相应的动作，如果数据包的包头部分不和规则的过滤域相匹配，则查找下一条规则，这个过程一直持续下去，直到找到一个规则的过滤域匹配该数据包，此时再根据这条规则的动作域采取相应的动作。

综上所述，防火墙规则的过滤域可表示为[3]：

< protocol, source IP, source port, destin IP, destin port >。

可以使用 Boolean expression 表示规则过滤域，且使用 Binary Decision Diagrams （BDD's）作为表示 Boolean expressions 的标准方法[4]。防火墙规则过滤域的二进制表示并置成一个二进制位串，使用函数 $\Phi_i = AS(R_i)$ 将防火墙规则过滤域 R_i 映射为布尔表达式

Φ_1。这样的表达式中的位可以是一个值(0 或 1),或者不予考虑。二进制位串的每一位分配给一个布尔变量,每个规则或者是某个变量或者是该变量的补码,这由规则的相应位的值确定。由于防火墙规则过滤域通常具有聚集性,所以使用 Boolean expression 表示规则,可以有效地将规则过滤域的二进制位串缩减。例如,规则:

R = <any,67.*.*.*,any,121.*.*.*,any>

对应的布尔表达式(Φ)中只有源 IP 的位 X_4 到 X_{35} 和目的 IP 的位 X_{53} 到 X_{83} 是需考虑的变量,其余位可不予考虑。因此:

$\Phi = (X_4' \wedge X_5' \wedge X_6' \wedge X_7' \wedge X_8' \wedge X_9' \wedge X_10' \wedge X_11) \wedge (X_{53}' \wedge X_{54} \wedge X_{10} \wedge X_{55} \wedge X_{56} \wedge X_{57} \wedge X_{58}' \wedge X_{59}' \wedge X_{60})$

这样,100 个变量就缩减到 16 个变量。

2.2 防火墙一致性测试

文献[1]提出了一种基于规范的防火墙测试方法,该方法以机构的网络安全策略作为规范,测试防火墙是否符合网络安全策略。鉴于防火墙的复杂性,文章做出了一些简化假设:假设所有的防火墙都是基于状态包过滤的,且不测试定时或序号带来的问题。

该防火墙测试方法测试防火墙与给定的安全策略的一致性,提出了一种网络安全策略形式说明的语言,并通过各种抽象测试例产生方法的组合提出一种新的抽象测试例生成方法。该方法能够找到防火墙配置和防火墙实现中的错误。

该防火墙测试方法是一个黑盒测试。其测试步骤为:首先,形式化具体的网络安全策略得到形式化策略;然后,根据形式化策略产生具体的测试例。该测试例由一系列的网络数据包(测试数据)和每个包是否到达各自目的地的期望描述构成;最后,通过直接在实际的网络上执行产生的测试例,找到防火墙配置和防火墙实现中的错误。

本文主要关注测试例的生成。文献[1]中测试例的生成由三部分构成:一是从形式化策略中产生测试元组;二是产生抽象测试例;三是结合测试元组和抽象测试例产生具体的测试例。

测试元组是根据形式化策略(下文简称策略)产生的。策略描述在不同区域间哪种类型的通信量被允许,因此根据策略产生的测试元组表示为:(sIP,dIP,proto,exp),一个测试元组描述形策略是否允许从源 IP 到目的 IP 的使用协议 proto 的连接。

抽象测试例旨在测试防火墙协议的实现。其生成方法为:对待测协议建模,从协议模型通过 UIO 方法产生 UIO 序列,该序列即是抽象测试例。假设 m 表示状态转换数,n 表示自动机的状态数,则在最坏情况下测试序列的长度为 $O(mn^2)$[5]。

最终的测试例是通过测试元组实例化抽象测试例得到的。

该方法假设策略包括 r 条规则,且最多有 p 个协议,则该方法所需要产生的测试元组数目为 $O(rp)$。每个协议只需执行一次抽象测试例的生成。每条策略只需执行一次测试元组的生成和测试元组实例化抽象测试例得到最终的测试例。因此,最坏情况下,每个测试元组实例化 $O(mn^2)$ 个抽象测试例产生的具体测试例有 $O(rpmn^2)$ 个。

该方法生成的测试例能够同时找到防火墙规范和防火墙实现的错误。但是,最坏情况下产生的最终测试例有 $O(rpmn^2)$ 个的分析是建立在给定的策略规则都是原子的前提下。比如规则(tcp, 202.38.64.22, 135,202.38.73.9,445, deny)。对于非原子规则,比如规则(any, *.*.*.*,any, *.*.*.*,any, deny),最坏情况下要考虑的网络地址空间为 2^{101},使用该方法产生的测试例达到 $O(2^{101}rpmn^2)$ 个。显然这样的网络地址空间太大,产生的测试例数量太多,较难在实际的测试中应用。

对于给定的规则为非原子的情况,通常有三种方法产生测试例。一是穷举法,即产生所有可能的测试例。这种方法简单,但会产生大量的数据包,也相当耗时。另一种方法是随机法,即随机选择测试例集。这种方法只产生少量的数据包,但是未能测试到决定性路径的可能性较高。第三种方法是选择法,即智能的有选择的生成数据包。显然,选择法是穷举法和随机法的一个折中。

本文将策略分段[2]的思想引入防火墙一致性测试中,对非原子策略规则使用选择法产生测试例,这就大大缩减了网络地址空间,同时也减少了产生的测试例的数量,使得文献[1]的防火墙一致性测试方法具有更高的效率。

3　基于策略分段的防火墙一致性测试

3.1　策略分段的思想

定义 1：网络通信地址空间是这样的空间，它的元素都是在网络环境中识别通信流的不同元组，尤其是从防火墙观点来看。它通常由传送协议、源地址、源端口、目的地址、目的端口组成。

定义 2：一个分段是一个或多个规则总网络通信地址空间的一个子集，分段中的元素完全符合同一策略规则集。

策略分段的思想是：分段和分类防火墙策略的地址空间。这种分段是基于关键测试因素的，比如规则交互、命令、复杂性及网络信息。

本文基于规则的相互作用和网络信息划分这些规范的策略。分段行为使得属于同一分段的数据包就策略的观点来看等同的。同时，分段保证了所有的分段互不相交。这样，测试所有的决定性路径等价于测试所有的这些分段。

3.2　策略分段算法

为了确定一个分段，需要包含以下要素：

Address space(AS)：表示一个分段的地址空间的布尔表达式。

Included rules(R_{in})：分段内的规则的有序链表。

Excluded rules(R_{out})：分段外的规则的有序链表。

Effective rules(R_{eff})：对分段做出贡献的规则的有序链表。

Owner rule(OR)：分段的属主，即 R_{in} 中的第一条规则。

Filtering action(ACT)：该分段采取的过滤动作，与分段属主的规则相同。

文献[2]给出一个分段算法，可以根据策略规则自动划分分段。该算法的核心思想是将策略规则地址空间划分，将其划分成互不相交的子集，每一个子集即一个分段。

下面具体介绍一下该算法：

算法输入：策略规则集合、防火墙默认过滤动作和初始域(即需要考虑的整个地址空间)。

算法输出：划分完毕的互不相交的分段，其中每一个分段要包含上文提到的六个要素。

算法步骤：

第一步：根据策略规则进行初始化，将初始分段添加入分段链表。

第二步：对于所有规则进行循环处理(第一层循环)，将每一条规则与分段链表中的每一个分段进行交运算得到分段内集合(IncSeg)和差运算得到分段外集合(ExcSeg)(第二层循环)。此时存在三种情况：1)得到 IncSeg 和 ExcSeg 集合均不为空，此时需要向分段链表中添加两个新的分段，即到 IncSeg 和 ExcSeg 所对应的分段；2)分段与规则没有交互，即 IncSeg 为空，此时将分段外集合对应的分段加入分段链表；3)分段的地址空间是规则的地址空间的子集，将 ExcSeg 对应的分段加入分段链表。最后从分段链表中删除当前分段，处理下一分段。

第三步：返回划分好的分段链表。

3.3　基于策略的地址空间分段实例

对于给定策略的规则集{R1,R2,R3}，具体规则见表 31。分段行为如图 1 所示，经策略规则地址空间划分，得到对应的分段见表 2。

表 1

R1	tcp	202.38.*.*	any	202.38.73.*	any	accept
R2	tcp	202.38.64.*	any	202.38.*.*	any	accept
R3	any	*.*.*.*	any	*.*.*.*	any	deny

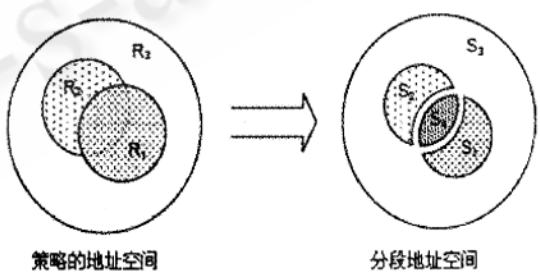

图 1

表 2

Segment	AS	Rin	Reff	OR	ACT
S1	¢1	{R1,R3}	{R1,R2,R3}	R1	accept
S2	¢2	{R2,R3}	{R1,R2,R3}	R2	accept
S3	¢3	{R3}	{R1,R2,R3}	R3	deny
S4	¢4	{R1,R2,,R3}	{R1,R2,R3}	R1	accept

3.4 方法分析

经策略规则地址空间划分得到分段后，需对每个分段 S_i 定义一个分段权重 ρ_i，该分段权重反映该分段的测试强度，其主要由分段内的规则的内在属性决定。从测试的角度来看，该权重是错误发生在该分段的机率的一个度量。文献[2]中给出了测试强度的形式化表示：

$$\rho_i = \omega_1 \frac{|S.R_{in}| + |S.R_{eff}|}{|R|} + \omega_1 \cdot \omega t(S.OR) + \omega_3 \sum_{r \in R_{in} \cup R_{eff}} \omega t(r) + \omega_4 \frac{||S.AS_{used}||}{||S.AS||}$$

其中，$(\omega_1 \cdots \omega_4)$ 为调节因子，由被测防火墙过滤算法决定；$\omega t(r)$ 是规则的权重，由规则的复杂性，规则间的相关性决定。

定义 θ_i 为 S_i 的测试密度，θ_i 形式化表示为：$\theta_i = \frac{\rho_i}{\sum_{s_i \in S} \rho_i}$；并定义总的测试规模 M，M 由测试人员根据以下两个条件选择：(1) 每个分段至少有一个测试例，故 $M \geq |S|$；(2) 每个分段 S_i 的测试例数目为 $M * \theta_i$，其中 $i, M * \theta_i \geq 1$。最后可以得到最终产生的测试例数目 N：

$$N = (\sum_{s_i \in S} M * \theta_i) * pmn^2 = M * pmn^2$$

由此可以看出，总的测试例数目取决于分段个数。通常情况下，防火墙策略规则具有很强的聚集性，规则间彼此交互很少，因此，分段数在可接受范围内。基于策略分段的防火墙一致性测试方法将指数级测试例数目降低为多项式级，提高了测试效率。

4 结论

策略分段技术基于规则的相互作用和网络信息划分规范的策略。分段行为使得属于同一分段的数据包就策略的观点来看等同的。同时，分段保证了所有的分段互不相交。这样，测试所有的决定性路径等价于测试所有的这些分段。策略分段技术保证了所有的决定性路径都能被测试到。同时，该技术避免了穷举测试和纯随机测试的不足。

本文在文献[1]的防火墙一致性测试中，引入策略分段的思想，并利用这一思想对非原子策略规则有选择的产生测试例，这就大大缩减了网络地址空间，并且减少了产生的测试例的数量，使得该防火墙测试方法具有更高的效率。

但是，文献[1]的防火墙一致性测试基于所有的防火墙都是基于状态包过滤的假设，且不测试定时或序号带来的问题。我们下一步的工作试图消除这些假设。另外，该方法未考虑 NAT 问题，这也是我们进一步研究的一个方向。

参考文献

1 Diana Senn, David Basin. Germano Caronni. Firewall Conformance Testing. In: Testing of Communicating Systems. 17th IFIP TC6/WG 6.1 International Conference. Montreal. Canada. 2005.

2 Adel El – Atawy, Khaled Ibrahim, Hazem Hamed et al.. Policy Segmentation for Intelligent Firewall Testing. In: First Workshop on Secure Network Protocols (NPSec. In conjunction with ICNP 2005). Boston. MA. USA. 2005.

3 Ehab S. Al – Shaer, Hazem H. Hamed. Modeling and Management of Firewall Policies. In: IEEE Transactions on Network and System Management. 2004. Volume 1 – 1.

4 R. Bryant. Graph – Based Algorithms for Boolean Function Manipulation. In: IEEE Transactions on Computers. 1986. Volume 35. Issue 8. Pages: 677 – 691.

5 Krishan Sabnani, Anton Dahbura. A protocol test generation procedure. In: Computer Networks and ISDN Systems. 1988. vol. 15. Pages: 285 – 297.