

一种分布式防火墙规则有效性检测算法

汤昂昂, 陈永波, 姬东鸿

(武汉大学 计算机学院, 湖北 武汉 430072)

摘 要: 给出了结构化查询语句的定义, 将查询的过程形式化描述, 提出了一种基于 SFDDs 间逻辑运算实现分布式防火墙规则有效性检测算法。该算法在保持原始规则完整性、一致性、紧凑性的基础上, 消除独立防火墙规则间异常, 保持原始规则语义上的一致。仿真结果表明, 基于 SFDDs 间逻辑运算实现分布式防火墙规则有效性检测算法可以快速有效对规则进行检测。

关键词: 分布式防火墙; 结构化查询; SFDD; 规则有效性; 网络安全

中图分类号: TP391.1

文献标识码: A

文章编号: 1000-7180(2015)02-0005-05

A Distributed Firewall Rules Validity Detection Algorithm

TANG Ang-ang, CHEN Yong-bo, JI Dong-hong

(School of Computer, Wuhan University, Wuhan 430072, China)

Abstract: The paper defines a Structured Query Sentence to describe the query process and proposes a distributed firewall rules validity detection algorithm based on semi-isomorphic firewall decision diagrams (SFDD) logical operation. The algorithm keeps consistency, completeness, and compactness of the original rules, eliminates the rule anomalies in intra-firewall. Our simulation results demonstrate that the algorithm achieves a significant improvement in validity detection of rules.

Key words: distributed firewall; structure query; SFDD; validity detection; network Security

1 引言

防火墙在保护网络安全方面发挥着关键性作用, 在防火墙中含有大量规则, 规则间可能存在异常, 因此理解并分析防火墙的功能变得复杂与困难。对于网络管理员而言, 查询通常作为一种了解与验证防火墙功能的有效方式。查询过程实际是对防火墙规则有效性进行检测。例如, 外网中的一台 PC 受到攻击, 内网的哪些主机接受已被攻击计算机的数据包。解决上述问题需要对上述问题进行形式化的描述, 实现规则有效性检测与提高检测效率。

在防火墙规则分析方面, 文献[1]中提出了一种防火墙规则有效性检测系统 Fang, 但是 Fang 是一种基于线性的防火墙规则^[2]有效性检测系统, 限制了检测效率的提高。提出了规则检测的思路, 但是并

没有给出具体的实现方法。文献[3]中提出一种专家系统来分析防火墙的规则, 但是建立专家系统用于分析一个防火墙中的规则是不切实际的。文献[4]中提出一种防火墙规则验证算法, 但该算法并不适用于分布式防火墙。在文献[5]中, Guttman 提出一种 Lisp-like 语言描述防火墙策略。文献[6]中提出一种 UML 语言描述防火墙策略语言, 这些只适用于描述完整规则。对于防火墙规则有效性分析方面, 需要简化查询的描述并提高检测的效率。

针对上述问题, 本文给出了一种结构化查询语言, 将规则有效性检测过程进行形式化描述, 通过将防火墙规则转化为半同构防火墙决策图 (SFDD), 根据 SFDD 间的逻辑操作实现对防火墙规则的有效性进行查询与验证。

收稿日期: 2014-03-20; 修回日期: 2014-05-13

基金项目: 国家自然科学基金重点项目 (61133012)

2 结构化查询语句定义与定理

防火墙规则包括过滤域和动作域,其中过滤域由 $F_1, F_2, \dots, F_d$ 组成. $D(F_i)$ 表示过滤域 F_i 的取值范围, $D(F_i)$ 是一个有限、非空、非负整数集合. 例如, IP 数据包的源地址的范围表示为 $[0, 2^{32} - 1]$. 满足一个具有 d 维过滤域 $F_1, \dots, F_d$ 的数据包可用 d 维元组 $(p_1, \dots, p_d)$ 表示, 其中 p_i 是 $D(F_i)$ 中的一个元素. 用 $\sum$ 表示数据包所有过滤域. $|\sum|$ 表示过滤域集合的大小, 假设过滤域包含 d 维元组, 则 $|\sum| = |D(F_1)| \times \dots \times |D(F_d)|$. 这里防火墙规则过滤域形式化表示为 $F_1 \in S_1 \wedge \dots \wedge F_d \in S_d$, 其中 S_i 是 $D(F_i)$ 的非空子集. 当且仅当 $p_1 \in S_1 \wedge \dots \wedge p_d \in S_d$, 数据包 $(p_1, \dots, p_d)$ 匹配过滤域 $F_1 \in S_1 \wedge \dots \wedge F_d \in S_d$.

2.1 结构化查询语句定义

定义一种结构化的查询语句 (Structure Query Sentence), 给出如何具体使用结构化查询语句对规则的有效性进行检测. 结构化查询语句的语法定义如下:

```
select  $F_i$  from  $f$ 
where  $(F_1 \in S_1) \wedge \dots \wedge (F_d \in S_d) \wedge (action = \langle act \rangle)$ 
```

其中, F_i 表示规则中的过滤域 $F_1, \dots, F_d$, f 代表防火墙规则集, S_j 表示过滤域 F_j 的域值范围是 $D(F_j)$ 的非空子集. act 表示规则的动作域. 在文中规则的过滤域仅考虑接受 (a) 和拒绝 (d) 这两种情况.

如图 1 所示, 内部网络与外部网络通过分布式防火墙连接, 这里的分布式防火墙仅包括上游与下游防火墙. 假设数据包具有以下 5 元组, 其分别为源接口 (I), 源 IP 地址 (S), 目的 IP (D), 目的端口 (N), 协议类型 (P). 下面是使用结构化查询语句具体描述查询相关的问题. 对于内部网络 Host1 通过防火墙端口 67 收到外部网络哪台主机发送的 TCP 数据包.

查询语句 Q 表示如下:

```
select  $I$  from  $f$ 
where  $(N \in \{67\}) \wedge (P \in \{tcp\}) \wedge (D \in \{Host1\}) \wedge (I \in \{all\}) \wedge (S \in all) \wedge (action = accept)$ 
```

2.2 结构化查询定理

防火墙查询语法的形式化描述如下:

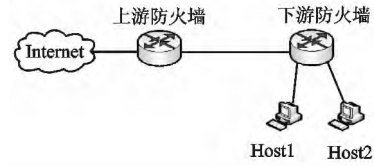


图 1 网络结构图

```
select  $F_i$  from  $f$ 
where  $(F_1 \in S_1) \wedge \dots \wedge (F_d \in S_d) \wedge (action = \langle act \rangle)$ 
```

Q 作为防火墙查询结果集, 满足以下条件:

$Q.result = \bigcup_{j=1}^n Q.r_j$ 其中,

$$Q.r_j = \begin{cases} S_i \cap S'_i, (S_1 \cap S'_1 \neq \emptyset) \wedge \dots \wedge (S_d \cap S'_d \neq \emptyset) \wedge (\langle act \rangle = \langle act' \rangle) \\ \emptyset, \text{其他} \end{cases}$$

定理的证明过程如下所示.

第一, 证明对于任意一个规则 r_j 满足 $Q.r_j \subseteq Q.result$. 其中 r_j 等价表示 $(F_1 \in S'_1) \wedge \dots \wedge (F_d \in S'_d) \rightarrow \langle act' \rangle$. 如果不满足条件 $(F_1 \in S'_1 \neq \emptyset) \wedge \dots \wedge (F_d \in S'_d \neq \emptyset) \wedge (\langle act \rangle = \langle act' \rangle)$, $Q.r_j = \emptyset$. 如果满足条件 $(F_1 \in S'_1 \neq \emptyset) \wedge \dots \wedge (F_d \in S'_d \neq \emptyset) \wedge (\langle act \rangle = \langle act' \rangle)$, $Q.r_j = S_i \cap S'_i$. 对于一个规则 r_j , 当一个数据包 $(p_1, \dots, p_d)$ 满足 $(p_1 \in S_1 \cap S'_1) \wedge \dots \wedge (p_d \in S_d \cap S'_d)$, 且 $\langle act \rangle = \langle act' \rangle$, 其中 $f.(p_1, \dots, p_d) = \langle act' \rangle$. $Q.r_j = S_1 \cap S'_1 = \{p_i \mid (p_i, \dots, p_d) \in \sum, (p_i \in S_1) \wedge \dots \wedge (p_d \in S_d) \wedge (f.(p_1, \dots, p_d)) = \langle act \rangle\}$, 则可知 $Q.r_j \subseteq Q.result$.

第二, 证明 $Q.result \subseteq \bigcup_{j=1}^n Q.r_j$. 假设 $Q.result$ 中存在一个数据包 p_i , 则至少存在一个数据包 $(p_1, \dots, p_d)$ 满足 $(p_1 \in S_1) \wedge \dots \wedge (p_d \in S_d) \wedge (f.(p_1, \dots, p_d)) = \langle act \rangle$. 数据包 $(p_1, \dots, p_d)$ 匹配防火墙 f 中的规则 r 即 $(F_1 \in S'_1) \wedge \dots \wedge (F_d \in S'_d) \rightarrow \langle act' \rangle$. 在一致性防火墙 f 中的规则和 $(p_1, \dots, p_d)$ 具有相同的动作域, 当 $f.(p_1, \dots, p_d) = \langle act' \rangle$ 时, $\langle act \rangle = \langle act' \rangle$.

因为每一 p_i 是 $S_1 \cap S'_1$ 中的一部分, 当 $(F_1 \in S'_1 \neq \emptyset) \wedge \dots \wedge (F_d \in S'_d \neq \emptyset) \wedge (\langle act \rangle = \langle act' \rangle)$ 时, $Q.r_j = S_1 \cap S'_1$. 因此 $p_1 \in Q.r_j$, $Q.result \subseteq \bigcup_{j=1}^n Q.r_j$.

3 规则有效性检测算法

分布式防火墙规则有效性检测算法的思想: 首先将防火墙规则转化为 FDD, FDD 保持了原始规则

在语义上的一致性、紧凑性与完整性。其次,通过对 FDDs 进行半同构化处理以及半同构体间的逻辑运算获得分布式防火墙有效性检测模型。最后,基于半同构体实现分布式防火墙规则的有效性检测。

3.1 SFDD 模型

分布式防火墙规则有效性检测过程中,将 FDD 所有的节点转化为半同构节点,完成半同构体的构建。以下为相关定义。

定义 1 当且仅当 FDD 满足每一个节点入度值小于等于 1 并且每一条边的标识为单一的连续区间时,FDD 为简单化 FDD。

定义 2 当 FDD f 和 f' 中对应非终端节点 v 和 v' ,且 v 和 v' 节点的边一一对应,具有相同的标识,称 v 和 v' 是半同构节点。

定义 3 当 FDD f 和 f' 中所有对应非终端节点具有相同的出度和入度值,且对应的边具有相同的标识,称 f 和 f' 是半同构体。

FDD 半同构化过程如下所示。

将规则序列 $\langle r_1, \dots, r_n \rangle$ 转化为对等体 FDD f ,FDD 半同构化之前要完成 FDD 简单化处理,FDD 的半同构化等价于全部非终端节点的半同构化。对于节点 v_a 有 m 条边 $\{e_1, e_2, \dots, e_m\}$, v_b 有 n 条边 $\{e'_1, e'_2, \dots, e'_n\}$, v_a 和 v_b 出边的 $I(e)$ 为单一连续区间。假设边 e_1 的 $I(e_1) = [a, b_1]$,边 e'_1 的 $I(e'_1) = [a, b'_1]$ 。 $I(e_1)$ 和 $I(e'_1)$ 比较结果分以下两种情况:(1) 当 $b_1 = b'_1$, $I(e_1) = I(e'_1)$ 。下一步比较边 e_2 和 e'_2 的 $I(e_2)$ 和 $I(e'_2)$ 值。(2) 当 $b_1 \neq b'_1$,假设 $b_1 < b'_1$,将边 e_1 拆分成两条边 e 和 e' , e 和 e' 分别指向 e'_1 的子图, $I(e) = [a, b_1]$, $I(e') = [b_1 + 1, b'_1]$ 。下一步比较 $I(e_2)$ 和 $I(e')$ 。这里边的标识左区间是相同的,将这个比较过程继续下去,直到节点最右侧的边为止。当最右侧边的值也相同时,完成节点的半同构化。重复上述过程,除终端节点外,实现所有节点半同构化。为了分析的简单,防护墙规则只考虑二维,0 表示规则的动作域(d),1 表示动作域(a),图 2 与图 3 所示为图 1 中上游防火墙 f_u 与下游防火墙 f_d 的规则,图 4 与图 5 是上下游防火墙规则产生的 FDD,图 6 与图 7 分别是上游防火墙与下游防火墙规则产生的半同构体 SFDD。

$$\begin{array}{l} F_1 \in [1,50] \wedge F_2 \in [1,60] \rightarrow a \\ F_1 \in [1,100] \wedge F_2 \in [1,100] \rightarrow d \end{array}$$

图 2 上游防火墙 f_u

$$\begin{array}{l} F_1 \in [1,30] \wedge F_2 \in [1,20] \rightarrow a \\ F_1 \in [1,30] \wedge F_2 \in [1,100] \rightarrow d \\ F_1 \in [1,100] \wedge F_2 \in [1,40] \rightarrow a \\ F_1 \in [1,100] \wedge F_2 \in [1,100] \rightarrow d \end{array}$$

图 3 上游防火墙 f_d

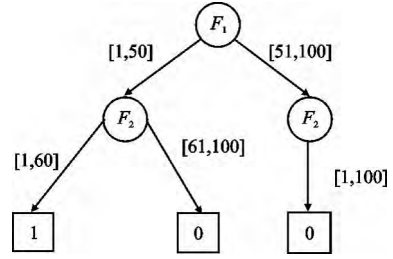


图 4 FDD f_u

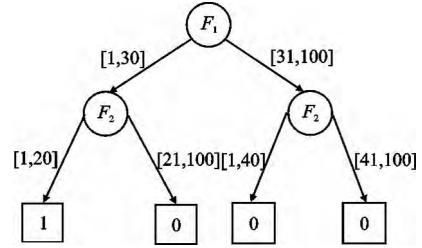


图 5 FDD f_d

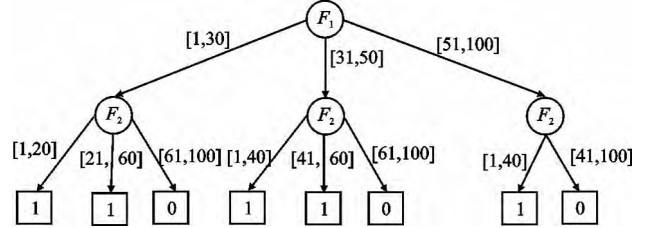


图 6 SFDD f

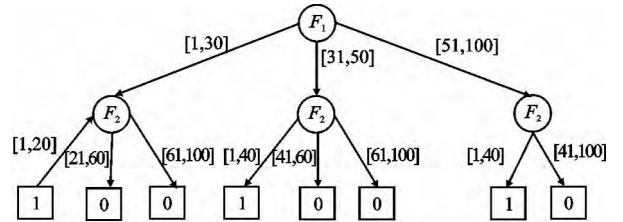


图 7 SFDD f'

3.2 规则有效性检测模型

(1) 半同构体 SFDD 间的逻辑操作

SFDDs 间逻辑操作为 $f \wedge f' \Leftrightarrow \sum_{i=1}^n (C_r^i \wedge C_r'^i)$

等价 f 和 f' 中所有对应决策路径中终端节点值之间逻辑操作。其中 $C_r^i, C_r'^i$ 表示叶子结点值。当 $C_r^i = C_r'^i$, C_r^i 和 $C_r'^i$ 的值保持不变。当 $C_r^i \neq C_r'^i$, $C_r^i = C_r'^i =$

0, 通过半同构体 $f \wedge f'$ 操作后, 与原始规则的语义保持一致. 图 8 所示为是 SFDDs 间逻辑运算.

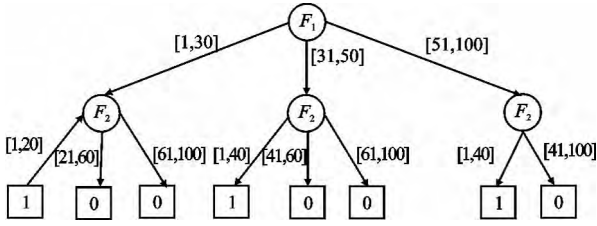


图 8 SFDDs 间逻辑运算

3.3 相关算法

采用类 C 语言描述, 给出了 SFDDs 逻辑运算算法、规则有效性检测算法的伪代码。

(1) SFDDs 逻辑运算算法

输入: 防火墙规则决策图 FDD f_a, f_b

输出: 逻辑运算后的结果

LogicalAnd(f_a, f_b, v_a, v_b)

Semi-isomorphic(f_a, f_b, v_a, v_b)

for each($v_1 e_1 \dots v_k e_k v_{k+1}$) in f' and f''

if($C_r \neq C_r'$) then

$C_r = C_r \& C_r'$;

end

// 以下实现 FDDs 半同构化

Semi-isomorphic(f_a, f_b, v_a, v_b)

$Q = \emptyset$;

$Q = (v_a, v_b)$;

while($Q \neq \emptyset$) {

PopQ(v_a, v_b);

$S = \text{Shape}(f_a, f_b, v_a, v_b)$; $Q += S$;

}

// 以下完成节点的半同构化

Shape(f_a, f_b, v_a, v_b);

if($v_a \cup v_b$ is terminal) return $\emptyset$;

else if ($v_a \cup v_b$ is nonterminal)

$E(v_a) = \{e_{a,1}, \dots, e_{a,m}\} I(e_{a,1}) < \dots < I(e_{a,m})$;

$E(v_b) = \{e_{b,1}, \dots, e_{b,m}\} I(e_{b,1}) < \dots < I(e_{b,m})$;

for($i = 1, j = 1; i \leq m \mid j \leq n$); {

$I(e_{a,i}) = [A, B], I(e_{b,j}) = [A, C]$

if($B = C$) then { $i = i + 1; j = j + 1$ };

else if($B < C$) then {

$I(e) = [A, B]$;

copy($e_{b,j}, t$);

$I(e_{b,j}) = [B + 1, C]$;

$i = i + 1$;

else {

$I(e) = [A, C]$;

$M(e) = M(e_{a,i})$

copy($e_{a,i}, t$);

$I(e_{a,i}) = [C + 1, B]$;

$j = j + 1$;

}

$E(v_a) = \{e_{a,1}, \dots, e_{a,k}\} I(e_{a,1}) < \dots < I(e_{a,k})$ and $k \geq 1$

$E(v_b) = \{e_{b,1}, \dots, e_{b,k}\}$;

$I(e_{b,1}) < \dots < (e_{b,k})$ and $k \geq$

$S = \emptyset$; for($i = 1; i \leq i$); ++)

$S += (e_{a,i}, t, e_{b,i}, t)$;

return S;

}

(2) 规则有效性检测算法

输入 Q: select F_i from f where

$(F_1 \in S_1) \wedge \dots \wedge (F_d \in S_d) \wedge (action = < act >)$

输出: 查询结果 Q.result

$Q.result = \emptyset$

CHECK($f.root, (F_1 \in S_1) \wedge \dots \wedge (F_d \in S_d) \rightarrow (action = < act >)$)

return Q.result

CHECK($v, (F_1 \in S_1) \wedge \dots \wedge (F_d \in S_d) \rightarrow (action = < act >)$)

if(v is a terminal node) and ($F(v) = < dec >$) then

$Q.result = Q.result \cup (S_i \cap S_i)$

if(v is a nonterminal node)

for e in (v) do

if ($e \cap e_j \neq \emptyset$) then

CHECK($e, t, (F_1 \in S_1) \wedge \dots \wedge (F_d \in S_d) \rightarrow (decision = < dec >)$)

4 性能分析与评价

由于缺少公开的防火墙规则数据, 我们通过构造无关的防火墙规则, 来检测线性防火墙规则有效性检测算法与基于 SFDDs 逻辑运算实现防火墙规则有效性检测算法的执行效率。

如图 9 所示, 随着防火墙中规则数量的增加, 基于线性与基于 SFDDs 间逻辑运算实现的防火墙规则有效性检测算法的处理时间都在增加。但是基于 SFDDs 间逻辑运算实现的防火墙规则有效性检测算法的执行时间增加幅度相对较小。在规则数量一致的情况下, 基于 SFDDs 间逻辑实现的防火墙规则有效性检测算法比线性防火墙规则有效性检测算法需要更少的时间。综上, 可以看出基于 SFDDs 间逻辑运算实现的防火墙规则有效性检测算法的执行效率更高。

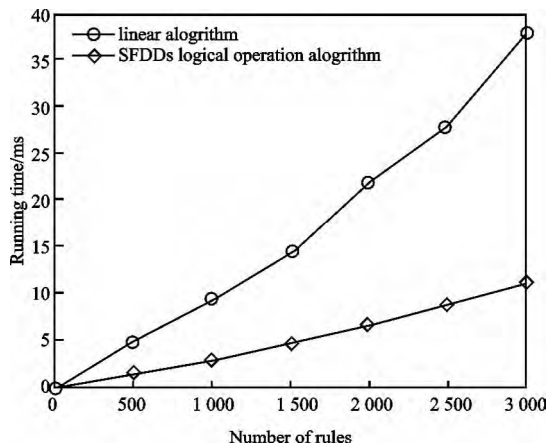


图9 防火墙规则有效性检测

5 结束语

提出了一种基于 SFDDs 间逻辑运算实现分布式防火墙规则有效性检测算法. 实验结果表明, 该算法具有较高的执行效率.

参考文献:

- [1] Mayer A, Wool A, Ziskind E. Offline firewall analysis [J]. Int' l J. Information Security, 2005, 5 (3): 125-144.
- [2] Hazelhurst S, Attar A, Sinnappan R. Algorithms for

Improving the dependability of firewall and filter rule lists[C] // Proc. Int'l Conf. Dependable Systems and Networks (DSN'00). New York, 2000: 576-585.

- [3] Anaya T, Moody C. A knowledge-based system implementation of intrusion detection rules [C] // New Generations (ITNG), 2010 Seventh International Conference on. Las Vegas, 2011: 738-742.
- [4] Liu A X. Firewall policy verification and troubleshooting[J]. Computer networks, 2009, 53(16): 2800-2809.
- [5] Guttman J D. Filtering postures: Local enforcement for global policies[C] // Security and Privacy, Proceedings, 1997 IEEE Symposium on. France: Paris, IEEE, 1997: 120-129.
- [6] Bartal Y, Mayer A J, Nissim K, et al. Firmato: a novel firewall management toolkit[J]. ACM Trans Computer Systems, 2004, 22(4): 381-420.

作者简介:

汤昂昂 男, (1990-), 硕士研究生. 研究方向为防火墙体系构造、可信计算、信息系统安全等. E-mail: tang_angang@qq.com.

陈永波 男, (1990-), 硕士研究生. 研究方向为网络安全、可信计算、信息检索等.

姬东鸿 男, (1967-), 博士, 博士生导师. 研究方向为自然语言处理、智能信息处理、语义网技术、搜索技术、数据挖掘等.

(上接第4页)

- [5] Abousamra A K, Melhem R G, Jones A K. NoC-aware cache design for chip multiprocessors[C] // Proceedings of the 19th International Conference on Parallel Architectures and Compilation Techniques. Austria, Vienna, ACM, 2010: 565-566.

作者简介:

葛健达 男, (1990-), 硕士研究生. 研究方向为数字集成电路设计. E-mail: gejianda0723@163.com.

谢 憬 男, (1981-), 助理研究员. 研究方向为大规模集成电路设计、片上处理器体系结构、先进集成电路设计方法.

唐 正 男, (1989-), 硕士研究生. 研究方向为数字集成电路设计.

王 琴 女, (1975-), 副教授. 研究方向为大规模集成电路设计、先进集成电路设计方法.

毛志刚 男, (1962-), 教授, 博士生导师. 研究方向为大规模集成电路设计、可重构体系结构设计、集成电路与元器件可靠性等.