计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.190500038

基于遗传算法的网络安全配置自动生成框架

白 玮[1] 潘志松[1] 夏士明[1] 成昂轩[2]
1 陆军工程大学指挥控制工程学院 南京 210014
2 93117 部队 南京 210018
(baiwei_lgdx@126.com)

摘 要 合理配置网络安全设备以对信息系统实施必要的访问控制,是网络安全管理的一项重要任务。随着网络规模的不断扩大,各种用户权限之间会形成复杂的依赖关系,传统基于人工的方式配置网络访问控制策略,主要是依据业务系统的实际需求,按照最小权限的原则进行分配,这种分配方式忽略了权限之间的依赖关系,容易产生过授权的现象,从而为网络带来安全隐患。为解决该问题,提出了一个基于遗传算法的安全配置自动生成框架。首先,以网络规划信息和配置信息为基础,确定用户可能的权限,提取网络基础语义,构建相应的网络安全风险评估模型,实现不同安全配置的安全评估;然后,对网络中所有可能的安全配置进行合理编码,确定遗传算子和算法参数,生成初始种群;最后,通过遗传算法,自动选取较优个体来生成子代个体。该框架能够通过自动比较不同的安全配置下的网络安全风险,以及在可能的配置空间内自动搜索安全配置的最优解,来实现网络安全设备访问控制策略的自动生成。构造一个拥有20个设备、30个服务的模拟网络环境对该框架进行验证,在该模拟环境下,该框架能够在种群样本数目为150的条件下,不超过10次迭代即可找到较优的安全配置。实验结果充分表明,该框架能够根据网络的安全需求,自动生成合理的网络安全配置。

关键词:网络安全;安全策略;多域配置;遗传算法;用户权限
中图法分类号 TP309

Network Security Configuration Generation Framework Based on Genetic Algorithm Optimization

BAI Wei[1],PAN Zhi-song[1],XIA Shi-ming[1] and CHENG Ang-xuan[2]
1 Command &Control Engineering College,Army Engineering University of PLA,Nanjing 210014,China
2 Unit 93117,PLA,Nanjing 210018,China

Abstract It is an important task in network security management to configure network security equipment reasonably and enforce access controls upon the information systems. With the increase of network size,there will be complex inter-dependent relationships among user privileges. Traditionally,access control lists are always generated manually according to the business requirements under the principle of least privilege,where the inter-dependent relationships are neglected. The network users may be granted with more privileges than they deserve,which may introduce vulnerabilities to network security. In this paper,a security configuration generation framework based on genetic algorithm optimization was proposed. Firstly,the framework extracts the user privilege information and network semantic information based on the network planning information and configurations information. And a network security risk assessment model is used to assess the network risk under different security configuration. Then,all possible access control configurations are encoded as genes. And initial population are generated based on the pre-determined genetic operators and super parameters. Finally,a better individual is generated according to the genetic algorithm. The framework cannot only compare the network security risks under different security configurations,but also search for the optimal solution of security configuration within the possible configuration space,thus realizing the automatic generation of network security device access control strategy. The framework is validated by constructing a simulated network environment with 20 devices and 30 services. In this simulation environment,the framework can find a better security configuration with no more than 10 generations of iteration under the condition of 150 population samples. Experimental data show that the framework can automatically generate reasonable network security configuration according to network security requirements.

Keywords Network security,Security strategy,Multi-domain configuration,Genetic algorithm,User privilege

到稿日期:2019-05-07 返修日期:2019-08-18 本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。
基金项目:国家重点研发计划(2017YFB0802800)
This work was supported by the National Key Research Development Program of China(2017YFB0802800).
通信作者:潘志松(hotpzs@hotmail.com)

1　引言

在网络安全防护体系构建过程中，通常需要在物理域、网络域以及信息域内同时实施多种访问控制策略，协同实现网络整体安全策略，而合理配置各种网络安全设备，实施访问控制策略是日常网络安全管理的一项重要任务。在这个过程中，由于网络中存在不同的访问控制策略，其主客体并不完全一致，为保证多个访问控制策略能够协同运行，需要主体之间具有严格的对应关系，但在实际网络运行过程中，主体之间的对应关系十分复杂，如果按照高层网络安全策略来确定网络安全设备配置，实施访问控制，则有可能发生用户过授权的情况，即用户获得了比其应得权限更多的权限。

以某企业网络为例，它同时在物理域和网络域实施访问控制。在物理域内，访问控制的主体为用户，客体为空间，所控制的权限是进入空间的权限；而在网络域内，访问控制的主体为终端，客体为网络服务，所控制的权限是服务访问权。为用户授予某个网络服务的访问权的过程分为两个步骤：1）在物理域内，允许其访问其办公室；2）在网络域内，允许其办公终端访问该网络服务。这两个访问控制策略联合正常生效，隐含了该用户在其办公室内只能访问自己办公终端的限制，如果该用户办公室内有其他终端，而两台终端的网络访问权限不一致，则该用户可能会绕过访问控制策略，通过另一台终端访问未被授权的网络服务。

查找冲突和冗余的网络安全配置，进而自动生成网络安全配置，是网络安全风险分析的一个重要方向。Hari 等首次提出了防火墙策略冲突问题[1]；Hamed 等提出基于排序二进制决策图（Ordered Binary Decision Diagrams）的防火墙策略冲突检测方法[2]；之后，国内外学者陆续提出了基于规则分割、策略树以及防火墙决策图等方式的防火墙冗余策略和冲突策略的检测方法[3-8]，但是这些方法主要对防火墙策略的五元组信息进行提取和分析，缺乏在网络空间整体视角下对防火墙规则影响的全面分析。针对网络安全策略冲突检测问题，Lupu 等将策略冲突分类为模态冲突和元策略冲突，并提出了相应的冲突检测方法与消解原则[9]；Macfarlane 等讨论了高层安全策略和底层防火墙实现之间的关系，对防火墙策略的管理方法和系统进行了总结[10]；Garcia 等提出了针对有状态防火墙的错误配置检测方法[11]；Hachana 等提出了一种基于多防火墙的当前配置信息，得到高级访问控制语义的方法[12]；Muthukumaran 提出了一个跨安全域的安全互操作模型[13]；Jarraya 等提出了一种在云环境下，对虚拟机迁移后的防火墙配置进行验证的方法[14]；Basile 等通过对防火墙可达性进行建模，达到发现安全问题和错误配置的效果[15]，但这些工作主要集中在网络域，没有对物理域、网络域以及信息域的安全策略进行统一分析。现有的多域安全策略统一分析方法，主要是通过形式化的语言来对目标网络进行描述，并通过逻辑的方式来判断网络是否能够达到不安全的状态。Probst 等提出了一种统一描述物理域和信息域的模型[16]；Kotenko 等提出了一种基于原子动作的前置条件和后置条件，来统一描述社会工程学攻击和物理接触攻击的模型[17]；Dimkov 统一描述了物理域、网络域和社会域信息，重点关注了物体的移动带来的安全问题[18]；在之前的工作中，我们提出了一个综合物理域、网络域以及信息域配置信息，对网络空间安全风险进行综合评估的配置风险评估框架 MDC-Checker，该框架对安全配置的评估结果可以作为衡量网络空间配置优劣的方法[19]。

本文以用户权限为网络空间安全风险度量指标，设计了一套基于遗传算法的网络安全配置自动生成框架，该框架能够通过提取网络空间多域配置的语义信息，基于权限依赖关系逻辑推理，来得到用户在当前配置下的实际权限，并在此基础上，重点研究网络安全配置自动生成框架和基于遗传算法的安全配置自动生成算法，实现了网络安全配置从人工生成向自动生成的转变。

2　基于遗传算法的网络安全配置自动生成框架

基于遗传算法的网络安全配置自动生成框架的主要作用是根据网络规划设计方案，在给定网络基本拓扑和最小化网络安全风险的条件下，实现网络安全设备的自动配置。该框架的整体结构如图 1 所示。

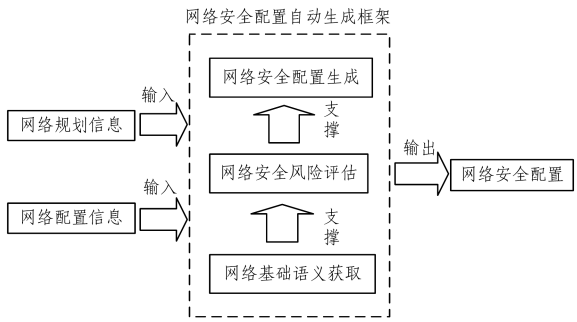

图 1　网络安全配置自动生成框架

Fig. 1　Network security configuration generation framework

该框架主要包括 3 个功能模块：网络基础语义获取模块、网络安全风险评估模块以及网络安全配置生成模块。其中，网络基础语义获取模块负责从网络的多域配置中提取相应的语义信息，得到相应的网络权限和权限依赖关系等信息，为定量度量网络安全风险提供必要信息；网络安全风险评估模块主要是依赖网络基本信息，根据网络可能的安全配置度量在该安全配置下网络的安全风险；安全配置自动生成模块主要根据最小化网络安全风险的原则，在可能的安全配置空间内寻找出能够最小化网络安全风险的配置，该配置即为当前网络的最优安全配置。其中，网络基础语义获取模块处于最底层，为网络安全风险评估模块提供基础数据；网络安全风险评估模块处于中层，为网络安全配置生成模块提供不同安全配置下的网络安全风险评估结果；网络安全配置生成模块处于最高层，调用下层模块来完成安全配置生成任务。

2.1　网络基础语义获取

网络多域配置语义分析模块主要负责从网络的多域配置中提取相应的语义信息，最终从网络中获得的基础语义主要包含用户权限和权限依赖关系两种。其中，用户权限在目标网络空间内，用户可能获得的所有权限的集合为 P，对于每一个用户权限 $p\in P$，存在一个用户权限到权限类型的映射 $l:P\rightarrow L$，其中，L 是用户权限类型集合，主要包含空间进入权、设备

使用权、设备支配权、端口使用权、端口支配权、服务可达权、服务支配权、文件支配权和信息知晓权等 9 种权限，每种权限所代表的含义如表 1 所列。

表 1 用户权限类型

Table 1 User privilege types

权限类型	权限含义
空间进入权	可进入某空间的权限
设备使用权	可正常使用某设备的权限
设备支配权	可对某设备配置进行更改的权限
端口使用权	可使用某设备端口访问网络的权限
端口支配权	可对某设备端口配置进行更改的权限
服务可达权	数据报文可到达某网络服务的权限
服务支配权	可正常使用某网络服务的权限
文件支配权	可对某数字文件进行读写的权限
信息知晓权	知晓某信息的权限

依据表 1 所列的权限类型，可根据网络空间的基本配置，对网络中所有可能的权限进行列举和统计，从而形成用户权限集合 P。如，对于涉及到的每一个空间(如房屋、楼宇、校园等)，均可在权限集合中增加相应的空间进入权；对于每一个涉及到的网络设备，均可增加相应的设备使用权和设备支配权等。

在获取空间中所有的可能权限后，还需要获取权限之间可能的依赖关系，权限之间的关系可以使用一阶逻辑进行描述，集合 R 表示所有的权限依赖关系。如，$r: PA \rightarrow PB$ 表示如果用户获得了权限 PA，则其可以获得权限 PB；同样，$r: PA \wedge PB \rightarrow PC$ 表示用户同时获得了权限 PA 和 PB，才能够获得权限 PC。权限依赖关系的建立分为 3 个步骤：首先，建立权限类型之间的依赖关系集合 KR，因为权限类型相对较少，其建立的工作量相对较小；然后，根据权限集合 P，对于每一条权限类型依赖关系 $kr \in KR$，逐一查找所有可能满足其推理条件的权限，并计算出其可以推理出的权限，从而形成不同的权限依赖关系，进而形成集合 R；最后，根据网络空间实际情况对所有推理规则进行验证，除去在实际中不存在的依赖关系，从而形成最终的权限依赖关系集合。

分析可能的权限转移规则可以发现，网络安全配置能够影响的权限依赖关系仅有服务可达权对端口使用权的依赖，即在防火墙上增加不同的访问控制列表，允许或拒绝来自某个端口的某个流量到达某个网络服务，从而影响能够使用该端口的人员访问该服务。也就是说，不同网络安全配置之间的差异性主要表现为，在该配置下用户服务可达权对端口使用权的依赖关系的不同。

2.2 网络安全风险评估

在配置优化的过程中，涉及 3 种用户权限，分别是用户应得权限、用户初始权限和用户实际权限。用户应得权限指，在网络规划和设计时明确用户应当得到的权限，它一般由网络管理人员对网络规划设计文件进行分析得到；用户初始权限是根据网络空间物理域和信息域安全策略，明确分配给人员的权限，这些权限可以通过分析相关安全配置得到；用户实际权限是用户根据网络初始权限获取到的用户权限，这些权限需要根据网络权限之间的依赖关系，由用户初始权限推断得到。

用户应得权限、用户初始权限和用户实际权限可以分别用矩阵 $\boldsymbol{PD}, \boldsymbol{PI}, \boldsymbol{PA} \in R^{|U| \times |P|}$ 来表示，其中，U 是网络中用户的集合，P 是网络中所有权限的集合。$\boldsymbol{PD}(i,j)=1$ 表示用户应该拥有该权限，$\boldsymbol{PD}(i,j)=0$ 表示用户不应该拥有该权限；$\boldsymbol{PI}(i,j)=1$ 表示用户在初始状态拥有该权限，$\boldsymbol{PI}(i,j)=0$ 表示用户在初始状态不拥有该权限，一般来说，在初始权限矩阵中，用户只拥有空间进入权和信息知晓权，而其他权限均是这两个权限的衍生权限；用户实际权限矩阵表明，在当年网络配置的情况下用户最终拥有的权限，其中，$\boldsymbol{PA}(i,j)=1$ 表示用户实际拥有该权限，$\boldsymbol{PA}(i,j)=0$ 表示用户最终不拥有该权限。

根据用户初始权限矩阵计算用户实际权限矩阵的过程，实际上是依据权限依赖规则进行推理的过程，具体内容请见文献[19]。在推理时，分别对矩阵 $\boldsymbol{PI}$ 的每一行进行推理，$\boldsymbol{PI}$ 的每一行对应着某一个用户的初始权限，按照预先建立的权限依赖关系，逐一对当前用户的初始权限进行推理，从而得到其实际权限，也就是其可能获得的最大权限。用户实际权限矩阵计算的主要流程如图 2 所示。

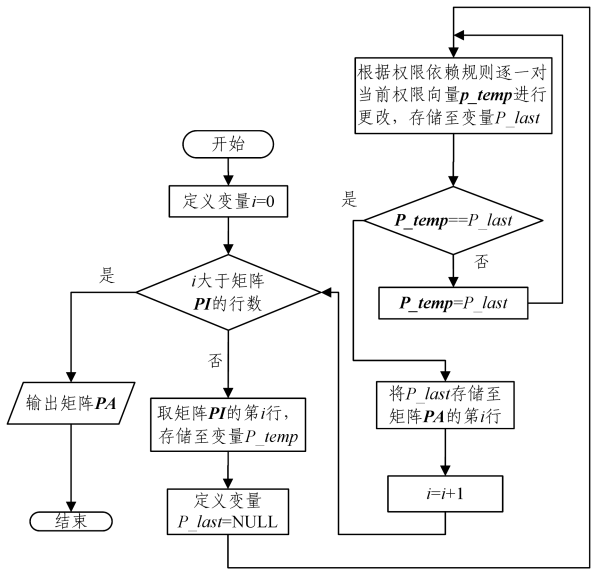

图 2 用户实际权限矩阵 $\boldsymbol{PA}$ 计算流程

Fig. 2 Calculation of user actual privilege matrix $\boldsymbol{PA}$

对当前配置下的网络安全风险进行评估，主要根据用户应得权限和用户实际权限的差值进行度量。如果用向量 $\boldsymbol{w}^{\mathrm{T}}=(w_1, w_2, w_3, \cdots, w_{|P|})$ 来表示用户权限的权重，则在安全配置 c 下的网络安全风险 $\sigma(c)$ 为：

$$\sigma(c)=\frac{\| abs(\boldsymbol{PA}^{(c)}-\boldsymbol{PD}^{(c)}) \times w \|_1}{|U| \times \| w \|_1} \tag{1}$$

其中，$\boldsymbol{PA}^{(c)}$ 和 $\boldsymbol{PD}^{(c)}$ 分别为安全配置 c 下的用户实际权限矩阵和用户应得权限矩阵，函数 $abs(\boldsymbol{M})$ 计算矩阵 $\boldsymbol{M}$ 中每一个元素的绝对值，$\| \boldsymbol{M} \|_1$ 表示矩阵的 L1 范数。

2.3 网络安全配置生成

安全配置自动生成模块主要利用用户实际权限推断模块提供的用户实际权限计算功能，来计算和比较不同网络安全设备配置所对应的适应度函数结果，再利用遗传算法(Genetic Algorithm)自动生成最优的网络安全配置。

遗传算法是一种受生物进化启发的学习方法，是模拟达尔文生物进化论的自然选择和遗传学机理的生物进化过程的计算模型，是一种通过模拟自然进化过程搜索最优解的方法，

其提供了一种求解复杂系统问题的通用框架，对问题有很强的鲁棒性，被广泛应用于许多科学领域。基于遗传算法的安全配置自动生成算法的主要步骤如图 3 所示，其主要可分为网络安全策略编码、种群初始化、生成新种群和输出最优个体等 4 个阶段。

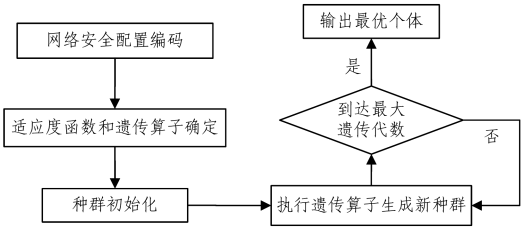

图 3　基于遗传算法的安全配置生成

Fig. 3　Security configuration generation based on genetic algorithm

3　基于遗传算法的安全配置自动生成算法

3.1　网络安全配置编码

在安全配置自动生成过程中，由于优化目标是希望找到最优的安全配置，其首先需要对安全配置进行编码，建立相应的“基因”和“染色体”。网络安全策略主要是指网络上存在的访问控制列表，一条访问控制规则可以用四元组(p_f, p_t, n, v)来表示，其中，p_f 和 p_t 是同属于一台设备的两个端口，n 表示数据流源地址，v 表示目标网络服务，该四元组表示在从端口 p_f 到端口 p_t 的链路上，允许源地址为 n、目的服务为 v 的数据流通过。对于需要优化安全配置的目标链路$\langle p_f, p_t \rangle$，若其所有可能通过该链路的网络数据流中源地址的数量为 n_{ft}，目的服务数量为 v_{ft}，则该链路上所有可能的访问控制规则的数量为 $n_{ft}v_{ft}$，若存在多条需要优化的链路，则网络上所有可能配置的访问控制列表的数量为：

$$W = \sum_{\langle p_f, p_t \rangle \in K} n_{ft} v_{ft} \tag{2}$$

其中，K 为需要优化的链路集合。如果用基因值 0 来表示某访问控制列表未被配置，1 表示该访问控制列表被配置，则网络上所有的安全策略（访问控制列表）配置的状态可以构成一个长度为 W 的数值串，该数值串可以作为描述当前网络安全配置的染色体，也可作为优化种群中的一个个体，其中每一个数值即是构成该染色体的基因。

3.2　适应度函数和遗传算子确定

在利用遗传算法进行网络安全策略优化时，需要确定相应的适应度函数和遗传算子。所谓的适应度函数，是评价个体优劣的一个标准。本算法中，如果第 i 个样本对应的安全配置为 c，则定义其适应度函数 $f(i)=1-\sigma(c)$，其中 σ 的定义如 2.2 节中的式(1)所示。基于遗传算法的安全配置自动生成算法的根本目的是，通过算法查找使得 $f(i)$ 值最大的样本，也就是网络安全风险最小的安全配置组合。

在遗传算法中，一般涉及 3 个算子：选择算子、交叉算子和变异算子。

选择算子指定从原种群中选择父体的方式，它根据个体适应度对种群中的个体进行优胜劣汰的操作，使得适应度较高的个体有较大的概率被遗传到下一代群体中，常用的选择算子有比例选择方法、无回放随机选择方法以及排序选择方法等。本文框架使用比例选择方法，也称为轮盘赌选择方法，即在选择父代时，第 i 个父代被选择的概率为 $\rho(i)=\frac{f(i)}{\sum f(i)}$，也就是说，适应度高的个体有更高的概率被选择。

交叉算子指定父代如何产生子代，常用的交叉算子有单点交叉、两点交叉和均匀交叉，本文算法采用均匀交叉的方式。具体地，对于两个父代个体 P_1 和 P_2，按照概率 ρ_e 随机产生一个交叉模板向量，其每一个分量为 1 或者 0，当交叉模板向量的第 i 位为 1 时，表示生成后代个体的第 i 位继承自个体 P_1；当其第 i 位为 0 时，表示生成后代的第 i 位继承自个体 P_2。

变异算子模拟了生物进化功能，对新产生的后代基因进行随机变异，从而增加算子的最优解搜索能力，常用的变异算子有位变异、均匀变异和高斯变异等，本文方法使用位变异，即对于个体的每一个基因，按照概率 ρ_m 随机指定其为变异点，如果该基因被指定为变异点，则将该位置对应的值取反，否则保留原值。

3.3　种群初始化

利用遗传算法进行安全策略优化的基本思想是，通过一个种群的不断进化来得到使目标函数最优的个体的过程，即最优解。因此，在算法进行迭代优化前，首先要产生一个初始种群，即产生 N 个初始个体。

按照 3.1 节中提及的网络安全配置编码可以看到，每一个安全配置均可以被表示成一个长度为 W 的二进制数值串，反之，每一个长度为 W 的二进制数值串，也均能够映射到一个网络安全配置上。因此，在种群初始化时，只需要随机产生 M 个独立的个体，即可满足相关要求。在初始化个体时，引入参数 z，$0 \leqslant z \leqslant 1$，其表示一个染色体中基因为 1 的比例，即在网络安全设备上配置为允许通过的访问控制列表的比例，z 值越大，设备允许通过的数据流的种类就越多。

3.4　生成新种群

在确定了初始种群后，即可以执行遗传算子生成新种群，其主要方式包括：

(1)根据构建的适应度函数，计算初始化种群中所有个体的适应度，其中个体 i 的适应度表示为 $f(i)$；

(2)执行选择操作，为种群中的每一个个体 i 赋予一个被抽中的概率 $\rho(i)=\frac{f(i)}{\sum f(i)}$，并按照这个概率选取两个父代个体；

(3)执行交叉操作，以概率 ρ_c 随机判断两个父体是否进行交叉，如果不需要交叉，则直接将两个父代个体加入子代，否则以概率 ρ_e 产生一个模板向量，并按照这个模板向量，产生两个新后代；

(4)执行变异操作，以概率 ρ_m 对新产生的后代进行随机位取反操作，并将其加入到新种群中；

(5)重复步骤 2－步骤 4，直至生成 N 个个体。

3.5　输出最优个体

判断当前种群生成代数，如果不大于预设代数 G，则重复进行 3.4 节的操作，否则计算当前种群内所有个体的适应度函数，输出适应度函数最大的个体，其所对应的安全策略即为找到的最优安全策略。

综上所述，基于遗传算法的网络安全配置自动生成算法如算法 1 所示。

算法 1 基于遗传算法的网络安全配置自动生成算法

输入：个体编码长度 w，染色体内基因为 1 的比例 z，种群规模 N，终止进化的代数 G，适应度计算函数 f(*)，样本交叉概率 ρ_c，基因交叉概率 ρ_e，基因变异概率 ρ_m

输出：最优安全配置 c

1. 随机生成 N 个长度为 w 的二进制数值串作为初始种群 P，每个数值串内值为 1 的个数为 z×w
初始化迭代次数
2. generation=0
3. while(generation<=G)
4. begin
5. 计算种群 P 中每一个个体的适应度 f(i)，以及其被挑选的概率 $\rho(i)=\frac{f(i)}{\sum f(i)}$
6. 置新种群 P′=∅
7. while(|P′|<N)
8. begin
9. 执行选择操作：根据概率选择两个父代个体 s 和 t。
10. 执行交叉操作：首先选择随机数 r∈[0,1]，如果 r>ρ_c，则得到新个体 s′=s 和 t′=t；否则，对 s 和 t 的每一个基因位置，选择 r′∈[0,1]，如果 r′>ρ_e，则该位基因不变，否则对该位的基因执行交叉操作，最终得到两个新个体 s′和 t′。
11. 执行变异操作：对于 s′和 t′的每一个基因位置，分别选择随机数 r″∈[0,1]，如果 r″>ρ_m，则对该位置的基因执行取反操作，从而得到两个新个体 s″和 t″，并将其加入种群 P′
12. end
13. 执行更新操作：P=P′
14. generation=generation+1
15. end
16. 计算种群中每一个个体的适应度 f(i)，挑选适应度最大的个体 i，计算其对应的安全配置 c
17. return c

4 实验与结果

4.1 实验环境

为了验证算法的有效性，文本对某公司的内部网络进行模拟仿真，如图 4 所示。

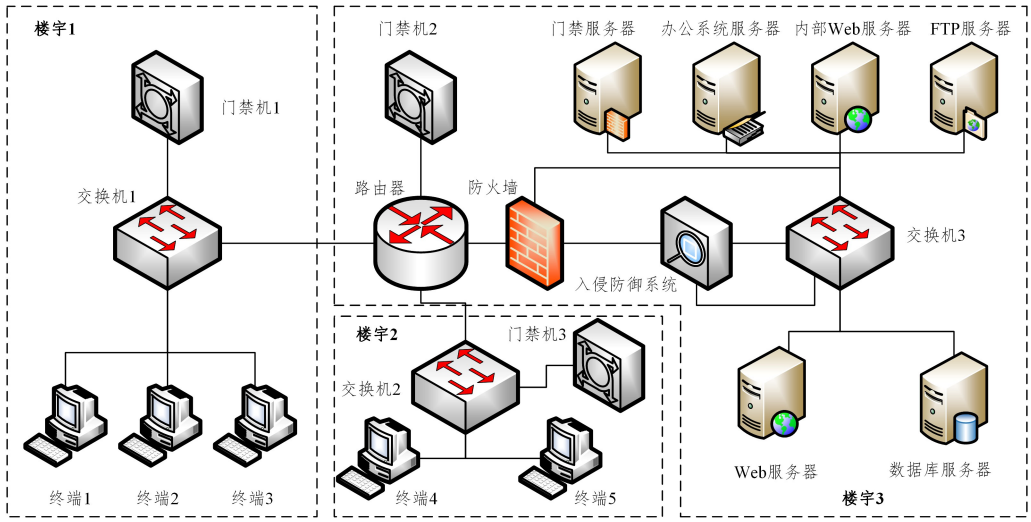

图 4 模拟实验网络

Fig. 4 Simulated network for experiment

该环境由 20 台设备组成，其中包括 1 台路由器、1 台防火墙、1 台入侵防御系统、3 台交换机、6 台服务器、3 台门禁机和 5 台终端，分布在 3 个楼宇和 8 个房间内。其中，终端 1、终端 2 和终端 3 放置在楼宇 1 的房间 1-1 中，交换机 1 放置在房间 1-2 中，门禁机 1 放置在楼宇 1 的大堂(房间 1-3)中；终端 4 和终端 5 放置在房间 2-1 中，交换机 2 放置在房间 2-2 中，门禁机 2 放置在楼宇 2 的大堂(房间 2-3)中；路由器、防火墙、入侵防御系统和所有服务器均放置在楼宇 3 的 3-1 房间中，门禁机 3 放置在楼宇 3 的大堂(房间 3-2)中。

网络中共有 30 个服务，其中，Web 服务器和内部 Web 服务器分别在 80 端口上提供 Web 服务，OA 服务器在 80 端口上提供 OA 服务，不同的用户使用不同的用户名和密码进行登陆(不同用户的服务在建模时被视为不同的服务)；FTP 服务器在 21 端口上提供 FTP 服务，用于所有的网络管理员共享相应的信息；数据库服务器在 1433 端口上提供相应的数据库服务，为 web 服务和 OA 服务提供底层支持；门禁服务器在 8080 端口上提供相应的认证服务，用于所有门禁机认证用户。除了这些服务，每个设备均提供相应的管理服务，所有终端和服务器均开启远程桌面服务；所有服务器和路由器提供 SSH 服务；防火墙、IPS 和门禁系统提供基于 Web 的管理服务。

网络中包含 5 个用户，分别为 Alice，Bob，Charles，David 和 Eric，日常分别使用终端 1 到终端 5，其中 Alice，Bob，Charles 是普通用户，不负责管理工作，David 主要负责网络设备的管理，Eric 主要负责安全设备和服务器设备的管理。依照这个原则，管理员预先设置了相应的物理域以及信息域防护策略。

在此基础上，利用本文方法自动生成防火墙的安全策略，需要配置的安全设备主要是防火墙 Firewall，其生成的访问控制列表主要部署在链路〈firewall_e0，firewall_e1〉上，即允许部分终端访问相应的网络服务。

4.2　实验过程和结果

首先，对网络上的多域信息进行提取，共得到247个用户权限，权限依赖规则采取文献[19]中描述的14条标准规则；然后，根据业务实际需求，建立相应的用户初始权限矩阵 ***PI*** 和用户应得权限矩阵 ***PD***；接着，对网络中的安全策略进行编码，通过分析多域实体发现，在防火墙左侧可能的源地址为22个，右侧可能的服务为26个(同一服务可以部署在不同的端口上)，则在链路〈firewall_e0，firewall_e1〉上可能存在22×26=572个可能的访问控制策略，因此每个个体用一个长度为572的二进制数字串来表示，用0和1分别表示在边上设置和不设置相应的安全策略；最后，设置不同参数，执行算法1，寻找最优的安全策略。

首先，本文比较了参数 ρ_m 对算法性能的影响，在 $N=150$，$G=30$，$\rho_c=0.7$，$\rho_e=0.7$，$z=0.1$ 的条件下，计算不同的 ρ_m 对最优个体查找性能的影响，每组参数重复实验5次，结果如表2所列。其中，最优个体适应度表示在5次实验中找到的最优个体，找到最优个体的代数表示在实验中找到适应度为0.8523的个体(最优个体)的代数，“—”表示该组参数中有一次或多次未找到最优个体。

表2　参数 ρ_m 对算法性能的影响

Table 2　Algorithm performance on varying ρ_m

ρ_m	最优个体适应度	找到最优个体的代数
0.01	0.8523	5.8
0.02	0.8523	6.6
0.03	0.8523	7.0
0.04	0.8523	6.4
0.05	0.8523	6.8
0.06	0.8523	8.0
0.07	0.8523	8.2
0.08	0.8523	6.4
0.09	0.8523	7.6
0.1	0.8523	9.8
0.2	0.85158	—
0.3	0.84152	—
0.4	0.83306	—
0.5	0.82818	—

然后，比较了不同的参数 ρ_c 对算法性能的影响，其他参数设置为 $N=150$，$G=30$，$\rho_e=0.7$，$\rho_m=0.01$，$z=0.1$，结果如表3所列。

表3　参数 ρ_c 对算法性能的影响

Table 3　Algorithm performance on varying ρ_c

ρ_c	最优个体适应度	找到最优个体的代数
0.1	0.8523	14.7
0.2	0.8523	12.8
0.3	0.8523	10.0
0.4	0.8523	8.4
0.5	0.8523	5.4
0.6	0.8523	6.6
0.7	0.8523	4.4
0.8	0.8523	4.8
0.9	0.8523	5.2
1	0.8523	5.2

接着，比较不同的参数 ρ_e 对算法性能的影响，其他参数为 $N=150$，$G=30$，$\rho_c=0.7$，$\rho_m=0.01$，$z=0.1$，结果如表4所列。

表4　参数 ρ_e 对算法性能的影响

Table 4　Algorithm performance on varying ρ_e

ρ_e	最优个体适应度	找到最优个体的代数
0.1	0.8523	7.0
0.2	0.8523	6.2
0.3	0.8523	5.2
0.4	0.8523	3.4
0.5	0.8523	5.4
0.6	0.8523	4.2
0.7	0.8523	3.6
0.8	0.8523	4.6
0.9	0.8523	8.4
1	0.8511	—

最后，比较了不同的参数 z 对算法性能的影响，其他参数为 $N=150$，$G=30$，$\rho_c=0.7$，$\rho_e=0.4$，$\rho_m=0.01$，结果如表5所列。

表5　参数 z 对算法性能的影响

Table 5　Algorithm performance on varying z

z	最优个体适应度	找到最优个体的代数
0.01	0.8523	3.0
0.02	0.8523	5.0
0.03	0.8523	4.0
0.04	0.8523	5.2
0.05	0.8523	7.0
0.06	0.8523	7.2
0.07	0.8523	8.6
0.08	0.8523	15.4
0.09	0.8523	—
0.1	0.8523	—
0.2	0.8511	—
0.3	0.8511	—
0.4	0.8458	—
0.5	0.8418	—
0.6	0.8392	—
0.7	0.8252	—
0.8	0.8239	—
0.9	0.8213	—

4.3　结果分析

由实验结果可以看出，不同参数的设置对优化结果有着较大的影响。综合表2－表5的数据可知，当 $\rho_m\leqslant 0.05$，$\rho_c\geqslant 0.6$，$0.4\leqslant\rho_e\leqslant 0.8$，$z\leqslant 0.07$ 时，算法能够取得较好的优化效果。

参数 ρ_m 决定基因变异的比例，从表2可以看出，当 $\rho_m>0.1$ 时，算法的性能急剧下降，证明同时对多个基因位进行交叉不利于保留最优样本，影响算法性能；参数 ρ_c 决定两个父代样本交叉的概率，从表3可以看出，对于选定的父代样本，增加其交叉的比例，将有利于快速找到可能的最优样本；参数 ρ_e 决定两个父代基因交叉的概率，从表4可以看出，找到合适的基因交叉比例有利于提升算法的性能，因为交叉比例过低或过高将同时意味着生成的子代和父代过于相似，不利于发现较优样本；参数 z 决定生成的安全策略中允许通过的数据流的数量，当 $z\geqslant 0.3$ 时，算法的性能快速降低，这与实际网络安全管理的经验相符，因为在实际配置安全策略时，严格进行访问控制，只允许较少部分的端口访问相应的服务。

结束语　合理配置网络安全设备是网络安全管理的重要任务，也是避免潜在安全风险的必然要求。传统基于人工配

置网络安全设备的方式，难以在网络规模不断扩大时合理匹配多个安全设备配置，容易出现配置错误和策略冲突。本文提出了一个网络安全配置自动生成框架，该框架能够在收集网络多域配置语义、管理网络多域信息的基础上，在不同安全配置下，对用户的实际权限进行推断，并利用遗传算法自动生成网络安全设备配置。实验结果证明，该框架能够根据网络安全策略，自动生成合理的网络安全设备配置，有效降低网络潜在的安全风险。

参 考 文 献

[1] HARI A, SURI S, PARULKAR G. Detecting and resolving packet filter conflicts[C]// Proceedings IEEE INFOCOM 2000 Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, 2000: 1203-1212.

[2] HAMED H, AL-SHAER E, MARRERO W. Modeling and verification of IPSec and VPN security policies[C]// 13TH IEEE International Conference on Network Protocols (ICNP'05). 2005: 269-278.

[3] HU H, AHN G J, KULKARNI K. FAME: a firewall anomaly management environment[C]// Proceedings of the 3rd ACM Workshop on Assurable and Usable Security Configuration. 2010: 17-26.

[4] GOBJUKA H, AHMAT K A. Fast and scalable method for resolving anomalies in firewall policies[C]// 2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). 2011: 828-833.

[5] MANSMANN F, GOBEL T, CHESWICK W. Visual analysis of complex firewall configurations[C]// Proceedings of the Ninth International Symposium on Visualization for Cyber Security. 2012: 1-8.

[6] CLARK P G, AGAH A. Modeling Firewalls for Behavior Analysis[J]. Procedia Computer Science, 2015, 62: 159-166.

[7] SAÂDAOUI A, BEN Y B S N, BOUHOULA A. FARE: FDD-based firewall anomalies resolution tool[J]. Journal of Computational Science, 2017, 23: 181-191.

[8] KHOUMSI A, ERRADI M, KROMBI W. A formal basis for the design and analysis of firewall security policies[J]. Journal of King Saud University - Computer and Information Sciences, 2018, 30(1): 51-66.

[9] LUPU E C, SLOMAN M. Conflicts in policy-based distributed systems management[J]. IEEE Transactions on Software Engineering, 1999, 25(6): 852-869.

[10] MACFARLANE R, BUCHANAN W, EKONOMOU E, et al. Formal security policy implementations in network firewalls[J]. Computers & Security, 2012, 31(2): 253-270.

[11] GARCIA A J, CUPPENS F, CUPPENS B N, et al. Management of stateful firewall misconfiguration[J]. Computers & Security, 2013, 39: 64-85.

[12] HACHANA S, CUPPENS B N, CUPPENS F. Mining a high level access control policy in a network with multiple firewalls[J]. Journal of Information Security and Applications, 2015, 20: 61-73.

[13] MUTHUKUMARAN T. Secure Interoperation Model for Different User Authentication System using Multi Level Security (MLS)[J]. International Journal of Advanced Research in Computer and Communication Engineering, 2015, 4(5): 596-600.

[14] JARRAYA Y, EGHTESADI A, SADRI S, et al. Verification of Firewall Reconfiguration for Virtual Machines Migrations in the Cloud[J]. Computer Networks, 2015, 93(P3): 480-491.

[15] BASILE C, CANAVESE D, PITSCHEIDER C, et al. Assessing network authorization policies via reachability analysis[J]. Computers & Electrical Engineering, 2017, 64: 110-131.

[16] PROBST C W, HANSEN R R. An extensible analysable system model[J]. Elsevier Advanced Technology Publications, 2008, 13(4): 235-246.

[17] KOTENKO I, STEPASHKIN M, DOYNIKOVA E. Security Analysis of Information Systems Taking into Account Social Engineering Attacks[C]// the 19th International Euromicro Conference on Parallel, Distributed and Network-Based Processing. 2011: 611-618.

[18] DIMKOV T. Alignment of organizational security policies: theory and practice[D]. Enschede: University of Twente, 2012.

[19] BAI W, PAN Z, GUO S, et al. MDC-Checker: A Novel Network Risk Assessment Framework for Multiple Domain Configurations[J]. Computers & Security, 2019, 86: 388-401.

BAI Wei, born in 1983, Ph.D, lecturer. His main research interests include network security, security policy and security management.

PAN Zhi-song, born in 1973, Ph.D, professor, Ph.D supervisor. His main research interests include artificial intelligence and network security.