

# 一种优化的实时网络安全风险量化方法

李伟明 雷 杰 董 静 李之棠

(华中科技大学计算机学院 武汉 430074)

**摘 要** 准确地评估网络安全风险是提高网络安全性的关键. 基于隐马尔卡夫模型的实时网络安全风险量化方法, 以入侵检测系统的告警作为输入, 能够实时量化网络风险值, 有效评估网络受到的威胁, 但仍然存在配置复杂、评估容易出现误差等问题. 该文提出了优化的方法, 利用参数矩阵自动生成代替手工设置, 提高了准确性, 简化了配置复杂度. 首先将 IDS 告警和主机的漏洞、状态结合起来, 定义攻击的威胁度来更好地体现攻击的风险, 并对攻击进行分类, 简化隐马尔卡夫模型的输入. 其次, 提出了利用遗传算法来自动求解隐马尔卡夫模型中的矩阵, 定义风险描述规则作为求解的优化目标, 解决隐马尔卡夫模型难以配置的问题. 风险描述规则为描述网络安全风险提供了形式化的方法, 利用这种规则建立的规则库可以作为风险评估方法的通用测试标准. 最后, 通过比较实验和 DARPA 2000 数据实际测试, 证明文中方法能够很好地反映网络风险, 量化网络面临的威胁.

**关键词** 网络安全; 风险评估; 隐马尔卡夫模型

**中图法分类号** TP393 **DOI号**: 10.3724/SP.J.1016.2009.00793

## An Optimized Method for Real Time Network Security Quantification

LI Wei-Ming LEI Jie DONG Jing LI Zhi-Tang

(School of Computer Science, Huazhong University of Science and Technology, Wuhan 430074)

**Abstract** Exactly assessing the security risk of a network is the key to improving the security level of a network. The Hidden Markov Model based real time network security risk quantification method can get the risk value and evaluate the threat dynamically and timely, whose input is Intrusion Detection System alerts. But it's complex to configure and it tends to acquire errors. These faults are resolved in an optimized method presented in this paper. The optimized method improves the accuracy and simplifies the configuration with automatically calculate matrixes in HMM. First, it combines IDS alert, host information and asset value to define the threat of an attack. The threat is more accurate than the alert and is applied to classify attacks. Second, the new method uses the genetic algorithm to generate the HMM status transformation matrix and observation matrix automatically, and it defines risk description rules as the genetic algorithm optimization target. The risk description rule provides a formal method to characterize the network security risk, and the rule base can be used as the test criterion for other risk assessment methods. At last, the comparative experiment and DARPA 2000 data experiment obtain good results and prove that this method is practical to measure the risk of network security.

**Keywords** network security; risk assessment; hidden Markov model

收稿日期: 2008-12-08; 最终修改稿收到日期: 2009-01-16. 本课题得到国家自然科学基金(60573120)资助. 李伟明, 男, 1975年生, 博士, 讲师, 主要研究方向为网络安全. E-mail: lwm@hust.edu.cn. 雷 杰, 男, 1983年生, 博士研究生, 主要研究方向为网络安全. 董 静, 女, 1984年生, 硕士研究生, 主要研究方向为网络安全. 李之棠, 男, 1951年生, 博士, 教授, 主要研究领域为系统结构、网络安全.

## 1 引 言

随着网络技术的广泛应用,网络规模不断扩大和开放,网络也会受到各种安全威胁的影响,如外部攻击者入侵、DDoS、蠕虫、病毒、内部攻击等.网络管理员需要考虑对网络安全影响的多种因素,因此很难有效地发现网络风险,并采取应对措施.如果能够对网络面临的风险进行实时量化的评估,动态发现网络中存在的主要问题并提出改进方向,就能迅速帮助管理员找到解决风险的方法,所以实时网络安全风险量化方法具有重要的理论和实用价值.

为了量化地评估网络安全,目前有各种方法和研究方向.对网络安全进行量化的指标很多研究集中于代码级别的 Bug 数或者被利用的安全漏洞报告数量,而 Howard<sup>[1]</sup>提出通过比较系统的 attack surface 来测量攻击的相对安全级别,所谓 attack surface 是所有对外界暴露的系统活动以及其访问的资源.通过对 attack surface 中每一个类型定义一个代价函数,计算其总的代价,然后通过测量系统 attack surface 的变化就可以知道系统安全的变化了.在量化标准建立的基础上,基于概率的安全方法可以用来量化安全,使用依赖性建模和分析理论对系统的可靠性、可生存性和安全性进行评估.对于较小的系统可以用状态图的方法来计算稳定状态的概率分布以及假设系统部件失效是独立的并服从特定分布来计算联合概率.对于大的系统,复杂 Petri 网是一种通用的建模和分析方法.基于概率的方法中,Ortalo<sup>[2]</sup>基于 COPS 提供的数据,采用依赖图对系统漏洞建模,模拟系统失效和安全分支之间的关系,用马尔卡夫模型计算攻击者导致安全失效的平均代价值来量化系统安全,给出系统的安全演化过程;攻击图的方法<sup>[3]</sup>主要内容是产生和分析系统的攻击图,通过搜集一系列系统属性来代表系统状态,然后作为攻击图的节点.例如主机中特定的漏洞,主机之间的连接,外部访问者对网络的访问权限.然后图的每一个连接代表攻击者可以进行的攻击.可以通过自动化工具产生攻击图,然后计算出攻击的成功概率来表示量化的系统安全度量.这些都是静态评估方法,主要通过系统的静态因素来评估风险,没有实时性,对网络正在遭受的攻击缺乏检测,从而导致网络安全问题应对处于被动和延迟的局面.

实时量化风险评估具有直观和动态的优点,但是相对也复杂得多,很多方法尚处于探索阶段.

Gehani<sup>[4]</sup>提出了基于主机的实时风险评估. Jonsson 和 Olovsson<sup>[5-6]</sup>利用 IDS(Intrusion Detection System)中的一些实验数据对攻击者行为进行了分析,通过考察系统的实时输入和系统的反应来对系统的可靠性进行一定的评估.而 Arnes<sup>[7]</sup>认为网络风险是主机风险的组合,而主机可以处于不同的安全状态,每种安全状态的概率决定了其安全风险.而状态之间的转换由隐马尔卡夫(HMM)状态转换矩阵和观察矩阵决定,其中状态矩阵表示主机安全状态的转化,观察矩阵表示 IDS 系统告警和状态之间的关系. Haslum<sup>[8]</sup>提出可以不使用离散 HMM 而是使用连续 HMM 来对状态转换进行计算.国内,陈秀真博士<sup>[9]</sup>也通过 IDS 数据实现网络安全威胁态势的定量评估,即利用 IDS 日志,结合服务、主机自身的重要性,按照网络系统组织结构,提出一种服务、主机、局域网系统 3 个层次安全威胁态势定量评估模型及相应的量化计算方法.可能存在的问题是对攻击之间的关系没有考虑,实际上攻击之间可能存在条件关系,多步攻击连续发生和多个攻击单独出现对主机造成的影响是不同的;主机处于不同状态,攻击造成的影响也是不同的.陆余良教授<sup>[10]</sup>提出将目标主机基本信息、目标主机可能存在的漏洞、各漏洞的可利用性影响因素利用 Dempster-Shafer 证据理论融合起来,得到主机的总体安全量化值.李涛教授等<sup>[11]</sup>提出了一种基于免疫的网络安全风险检测模型,主要利用人工免疫理论,通过检测网络虚拟抗体浓度对网络系统面临攻击时的实时风险评估.张永铮博士<sup>[12]</sup>为了提高网络风险评估的准确性,研究了网络中主机风险的联系,使用了网络节点关联性(NNC)的分析方法.利用 NNC 可以将若干孤立的弱点联系起来,更加准确分析网络总体安全风险.

本文主要是对基于 HMM 的实时量化网络安全风险评估方法的改进和优化,基本结构如下:第 2 节描述基于 HMM 的网络安全风险量化方法,指出其不足并提出本文的解决方法,同时也列举了本文的创新点;第 3 节描述 IDS 告警的分类方法,通过引入威胁度的概念,将 IDS 告警优先级别、主机漏洞严重度和资产重要性结合起来对 IDS 告警进行分类;第 4 节描述利用遗传算法求解 HMM 中 Trans 矩阵和 Obs 矩阵.重点是定义风险描述规则库来得到优化解,风险描述规则可以作为一种通用的网络风险描述方法,支持不同的风险量化方法;第 5 节是实验,通过适应性测试、比较测试, DARPA 2000 数据集 3 个实验来证明本方法的有效性.

## 2 基于 HMM 的网络安全风险量化方法

HMM 方法首先定义每台网络中的主机具有  $N$  个状态,用  $S=\{s_1,\cdots,s_N\}$  表示,那么该主机的状态序列为  $X=\{x_1,\cdots,x_T\}$ ,  $x_t\in S$ . 通常认为主机可以处于 4 种状态: Good, Probed, Attacked, Compromised 分别用 G, P, A, C 表示,那么  $S=\{G,P,A,C\}$ . 如果能够较为准确地计算主机处于何种状态,那么就可以定量分析主机的风险. 如果假设主机所能够观察到的攻击有  $M$  种,用  $A=\{a_1,\cdots,a_M\}$  表示,那么攻击序列为  $Y=y_1,\cdots,y_T$ ,其中  $y_t\in A$ . 为了计算状态,HMM 还包含一个三元组  $\lambda=(Trans, Obs, Init)$ ,其中 **Trans** 表示状态转换矩阵,即主机状态之间转换的概率,  $Trans_{ij}$  表示在  $t$  时刻状态为  $S_i$ ,那么到  $t+1$  时刻状态为  $S_j$  的概率,即  $Trans_{ij}=P(x_{t+1}=s_j|x_t=s_i)$ ,  $1\leq i,j\leq N$ . 观察矩阵 **Obs** 表示当处于某一个特定状态的时候,观察到某种攻击的概率,即  $Obs_{nm}$  表示在时刻  $t$ ,主机处于  $S_n$  状态观察到  $a_m$  的概率,即  $Obs_{nm}=P(y_t=a_m|x_t=s_n)$ ,  $1\leq n\leq N$ ,  $1\leq m\leq M$ . 初始状态 **Init** 是一个向量,表示计算开始主机处于各个状态的概率  $Init=(r_1,\cdots,r_N)$ . 通过 **Trans** 和 **Obs** 可以计算出当前主机处于各种状态的概率  $p_t=(r_1,\cdots,r_N)$ .

在  $t$  时刻状态分布表示为  $r_t=\{r_t(i)\}$ ,  $1\leq i\leq N$ ,状态的分布概率公式为

$$r_t(i)=P(x_t=s_i|y_t) \tag{1}$$

再引入一个代价向量 **C**,代表一台主机在每个状态的风险值,那么可以将主机状态的定性分析转化为定量分析. 如  $C=\{1,10,20,100\}$ ,表示该主机在每种状态下的风险,可以利用公式计算主机当前风险值为

$$R=\sum_{i=1}^N r_i c_i \tag{2}$$

如果一台主机的风险值在 1~10 之间表示很可能被探测到了,在 10~20 之间表示已经遭受到了攻击,如果超过 20 表示攻击已经比较严重了. 假设一个网络中有  $L$  台主机,那么很容易得到整个网络的风险值:

$$R_{net}=\sum_{i=1}^L R_i \tag{3}$$

这样以 IDS 告警作为输入,利用 HMM 来量化网络安全风险,具有以下 3 个优点:(1)易于量化,可以给主机的每个状态确定一个风险代价,再综合

每个状态的概率,得到当前主机一个明确的风险值,并且由这个风险值的高低可以直观地看到安全事件的严重程度,并观察到细微的变化.(2)由于输入是动态的,所以输出也是动态的,随着系统受到不同的攻击,可以实时地反应系统的风险.而对于被边界防火墙或者 IPS 阻隔的攻击,不会对内部网络造成影响,所以即使内部网络存在某些漏洞也不会产生风险,这样更加准确地体现了风险的含义.(3)参数可配置,通过对不同网络采用不同的初始状态矩阵、**Trans** 矩阵、**Obs** 矩阵和风险代价向量会得到不同的风险评估结果,对于不同的网络环境有很好的适应性.最后,隐马尔可夫模型的计算量是比较小的,一般内部网络主机数量有限,因此整个计算过程消耗的时间非常短,据本文统计对一个 C 类地址网络进行一次风险评估,计算时间在 10ms 以下,所以当网络受到攻击,其风险值可以实时更新.

但是利用 HMM 进行安全风险量化的方法也存在两个明显问题,首先是如何控制 **Obs** 矩阵的规模,因为对于 IDS 告警来说攻击方式多种多样,Snort 基本告警就有 8000 多个,直接将 IDS 告警和 **Obs** 矩阵关联,那么 **Obs** 矩阵的规模将非常庞大,运算效率会非常低.所以必须找到一种合适的方法,将告警归类,才能把 HMM 的 **Obs** 矩阵缩小到一个可以快速计算的规模.其次,如何确定 **Trans** 矩阵和 **Obs** 矩阵的具体数值.要准确地配置这两个矩阵,难度非常大.一般的方法是网络管理员手工设置,但是手工设置随意性较大,准确性较差,效果的好坏往往依赖于网络管理员水平高低,不适合大规模推广应用. Holsopplea 对此就提出了疑问,并以此为理由没有采用基于 HMM 的方法<sup>[13]</sup>.为解决这些问题,本文引入了攻击威胁度来对攻击进行分类,并将攻击和网络环境、用户配置结合起来.另外通过遗传算法来自动对 HMM 中的矩阵进行自动求解.在对遗传算法设定优化目标时,本文提出了一种通用的网络安全风险描述规则,用来形式化地描述攻击和网络安全风险之间的联系,并建了一个 29 条的风险描述规则库.这 3 点都是对原有方法的优化和创新,其中风险描述规则可以作为一个通用的网络安全风险量化测试标准,用来评价各种不同的方法的有效性,有待进一步的深入研究.

## 3 威胁度算法

本算法综合了漏洞、资产、环境因素等各个方面来评估一个告警所表示的攻击的威胁程度.通过计

算一个 IDS 的告警对网络造成的影响,将 IDS 告警进行分类.预设的威胁度为 10 级,即所有 IDS 告警将根据影响程度归入 10 个类别,这样可以将 *Obs* 矩阵的大小控制为  $4 \times 10$ . 首先可以定义以下的概念,如表 1 所示.

表 1 攻击威胁度定义			
名称	中文名	含义	范围
<i>Severity</i>	严重度	攻击针对的漏洞的严重程度	0~9
<i>Asset</i>	资产值	攻击针对的资产的关键程度	0~9
<i>Priority</i>	优先级	攻击目标或攻击类型的优先级别	0~9
<i>Reliability</i>	可信度	攻击成功执行的可能性	0~9
<i>Threat</i>	威胁度	以上几个值的综合	0~9

*Severity* 值反映攻击所针对的漏洞的严重程度.漏洞的严重度由漏洞披露机构在提出一个新的漏洞时确定. *Asset* 值反映攻击针对资产的关键度,例如针对普通 PC 和关键服务器的两个同类型攻击具有不同的威胁度,因为目标资产的关键程度不同.该值由管理员设定,可以为一个子网设定 *Asset*,也可以为某个关键的服务设定 *Asset*. *Priority* 值反映在管理员的视角中为某类攻击的优先级.该值由管理员通过 Policy 来指定. *Reliability* 值反映一个攻击成功执行的可能性,这个值需要结合攻击所在的环境因素来计算.大部分严重度较高的攻击都需要目标主机的配置,具有相应的漏洞,若是这些条件不满足(比如针对的端口没有打开,或是服务没有运行)则攻击成功的概率较低,反之,若是漏洞确实存在则攻击成功的概率较高.我们使用漏洞扫描程序(Nessus)和主机信息扫描程序(Nmap)得到实时的受保护网络的主机配置信息,同时系统维护一个知识库用来保存每种 IDS 告警所对应的漏洞信息(相应攻击若成功执行需要满足什么样的配置条件),包括应用程序、漏洞号、操作系统、端口、可能的后果(主机配置的改变).将这些因素结合起来最终得到一个攻击的威胁度 *Threat*.

- 具体算法如下:
1. 分析 IDS 的一条告警.
  2. 从配置文件中读出攻击目标主机或者服务器的 *Asset* 值.
  3. 综合用户定义的 *Priority* 值和 IDS 告警中的 *Priority*,两者相加转化为 *Severity* 值.
  4. 计算 *Reliability* 值.
    - 4.1. 取出相告警的目的 IP 地址 *dstIP*,目的端口 *dstPort*,攻击特征值 *SID*.
    - 4.2. 通过 *SID* 查找 *Snort-os.xml* 配置文件,得到该攻击对应的 OS 值.由 *dstIP* 查找 *nmap* 扫描出来的主机操

- 作系统信息 *host-os.xml*,如果两者不匹配,可靠性很低,  $Reliability=1$ ,结束计算 *Reliability*.
- 4.3. 如果 OS 匹配,那么根据 *dstPort* 查找 Nessus 扫描出来的主机打开端口信息 *host-port.xml*,如果不匹配,可靠性较低,  $Reliability=3$ ,结束计算 *Reliability*.
  - 4.4. 根据 *dstPort* 查看 Nessus 扫描该端口是否存在漏洞  $valunrable==1$ ,如果不存在,攻击成功的可能性较低,  $Reliability=6$ ,结束计算 *Reliability*.
  - 4.5. 根据 *dstPort* 查找 Nessus 扫描结果,如果 Nessus-ID 和 *SID* 不匹配,  $Reliability=8$ ,结束计算 *Reliability*.
  - 4.6. 如果 NessusID 和 *SID* 相匹配,  $Reliability=9$ .
  5. 计算最后的 *Threat* 即攻击威胁度.
    - 5.1. 如果  $Reliability \leq 3$  那么  $Threat=0.1 \times Asset + 0.1 \times Severity + 0.8 \times Reliability$ .
    - 5.2. 否则  $Threat = 0.1 \times Asset + 0.3 \times Severity + 0.6 \times Reliability$ .
- 算法第 5 步中参数的权值是通过两个标准来确定的,首先要求 *Threat* 值尽量准确反映攻击对系统的威胁;其次希望各个攻击的 *Threat* 在 0~9 的空间分布比较均匀,以便区分不同的攻击对网络造成的影响.对于第一个标准,为了衡量 *Threat* 值是否准确,定义如下两个变量:
- 威胁度算法可能会错误地将一些不具有威胁的告警归入到高威胁告警和紧急事件中,这样的告警称为“评估误报”(Assessing False Positives,AFP).我们用评估误报率%AFP 来衡量这种错误,计算公式见式(1),其中  $N(AFP)$  表示高威胁告警和紧急事件中误报的数量, $N$  表示威胁度评估完成之后的告警数量.“评估漏报”(Assessing False Negatives,AFN)指属于高威胁告警和紧急事件但是没有被归入其中的告警.同评估误报类似,我们用评估误报率%AFN 来衡量这种错误,计算公式如式(2).
- $$\%AFP = N(AFP) / N \tag{4}$$
- $$\%AFN = N(AFN) / N \tag{5}$$
- 本文搜集了 70 种对网络造成不同影响的攻击,对 *Asset*, *Severity*, *Reliability* 参数假设 4 种情况:3 个参数的权值相等;不考虑 *Asset* 其他参数的权值相等;不考虑 *Severity* 其他参数的权值相等;不考虑 *Reliability* 其他参数的权值相等,然后分别计算其 *Threat* 值,判断%AFP 和%AFN,得到如图 1 的结论.
- 从图 1 看出, *Reliability* 是一个比较关键的参数,对结果影响最大.再综合考虑 *Threat* 对不同攻击分布均匀的标准,就得到现在的方案.其中, *Reliability* 如果大于 3 的话,表示攻击和目标主机

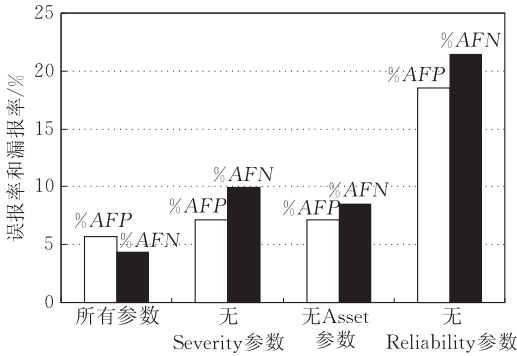


图 1 威胁度参数测试

信息比较符合,这时候进一步判断攻击是否成功,需要参考 *Severity* 值,因此通过降低 *Reliability* 的权重来增加 *Severity* 权重来提高可靠性,降低误报,保证具有高 *Threat* 值攻击的高可信度. 当 *Reliability* 小于等于 3 的时候,通常表示攻击和目标主机的信息不是非常符合,这时候攻击本身表示的 *Severity* 意义就不大,因此,应该降低其权重,增加 *Reliability* 权重,主要依靠 *Reliability* 进行判断. 采用这个方案后 %AFP 和 %AFN 都可以降低到 1.5% 左右.

4 求解 *Trans* 矩阵和 *Obs* 矩阵

本文采用了遗传算法来自动得到 *Trans* 和 *Obs* 矩阵,为了和手工配置的矩阵相区别,称这样得到的结果为优化的 *Trans* 和 *Obs* 矩阵. 遗传算法是一种模拟生物在自然环境中的遗传和进化过程的一种自适应全局优化概率搜索算法,已在组合优化、模式识别、神经网络、经济预测等领域得到广泛应用. 遗传算法是一种启发式算法,它的基本思想来源于遗传进化,主要是借助于生物进化机制与遗传学原理,按照自然选择和适者生存的原则,利用简单的编码技术和繁殖机制,模拟自然界生物群体优胜劣汰的进化过程,实现对复杂问题的求解. 遗传算法的重要部分包括编码表示、设置适应度函数、杂交和变异运算策略,为了将遗传算法和 HMM 方法相结合,需要对这些部分进行设置.

4.1 编码表示

遗传算法的搜索空间由染色体组成,染色体一般由二进制串组成,因此需要将 *Trans* 矩阵和 *Obs* 矩阵映射到染色体空间. 值得注意的地方有两个,为了符合 HMM 矩阵的定义, *Trans* 矩阵和 *Obs* 矩阵的每行之和应该为 1,即概率的和为 1;其次 *Trans* 矩阵和 *Obs* 矩阵中每个概率为浮点数,转化为二进

制的时候,必须在精度和长度之间取得一个折衷.

*Trans* 和 *Obs* 矩阵编码的算法如下:

- 1. *Trans* 矩阵的大小为  $4 \times 4 = 16$ , *Obs* 矩阵的大小为  $4 \times 10 = 40$ .
- 2. 为防止误差,对 *Trans* 和 *Obs* 的每行矩阵进行归一化,即为和为 1.
- 3. *Trans* 和 *Obs* 矩阵中每个元素都为 0 到 1 的小数,除以 0.015625,得到的商用一个 6bit 长度的二进制串表示.
- 4. 再次检查每行矩阵是否归一化.
- 5. 将 *Trans* 和 *Obs* 矩阵链接起来,得到一个 336bit 的二进制串,即为染色体.

4.2 适应度计算

如何判断一个染色体的适应度是本方法中最困难的地方,因为精确描述网络的风险值难度很大,存在很多模糊的地方,例如管理员可能知道某种攻击会对网络产生很大风险,但是如果具体给出在 0~100 间的数值就很困难,也许 85 和 95 都是可行的,而且该数值也会随着网络环境不同而变化. 另外要求管理员对数量众多的每个攻击都给出风险值也是一个工作量非常大的工作. 为了避免这种情况,本文采用了定义风险描述规则的办法来得到适应度.

一条风险描述规则用来描述一系列攻击对于网络风险的影响,其主要语法如下:

```
riskRule ::= destip attack state risk
destip ::= "destip" ":" ipAddressConst
attack ::= "attack" ":" attackList
attackList ::= attackSig "," attackList
attackSig ::= variable "=" intConst intConst intConst
state ::= "state" ":" termList
risk ::= "risk" ":" termList
```

任何一个风险描述规则由 4 个部分组成:

- (1) 目的地址(destip). 指的是攻击所针对的目的 IP 地址,用于将攻击目标和主机漏洞信息、主机资产信息结合起来;
- (2) 攻击序列(attack). 表示对一个主机的一系列攻击,每个攻击包含 3 个参数: signatureID 表示攻击被 IDS 识别后赋予的标识,端口表示攻击针对的 TCP/UDP 端口,如果是其他协议那么为 0;严重程度表示 IDS 赋予该攻击的严重程度,这个数值完全从 IDS 的角度出发,分类比较简单和不确定. 例如一个攻击描述为

```
$1=1123 80 2,
```

其中, \$1 表示这是第一个攻击, Signature ID 为 1123,攻击的端口为 80,IDS 提供的攻击严重度为 2. 这样一个攻击序列可以如下所示,攻击之间用逗

号隔开：

\$1=1123 80 2, \$2=456 9999 3, \$3=432 7777 1.

(3) 主机状态判断(state). 主机状态判断是对应一个攻击,判断当前主机最可能处于什么状态. 状态变量描述主机当前的状态,数量和 Attack 中攻击的个数相等,例如 \$s1 表示第一状态. 当遇到前面的攻击序列时,主机可能取的状态序列为

$$\$s1==G, \$s2==P, \$s3==A,$$

其中 \$s1 表示主机状态序号,==表示判断是否相等,G 表示 Good 状态,合起来表示当一台主机受到 \$1=1123 攻击的时候,其状态应该在 Good 状态,这样本状态序列中的 3 个攻击分别对应 G,P,A 状态. 当然可以通过逻辑运算符描述将多个判断结合起来,例如, \$s1==A || \$s1==C.

(4) 风险评价(risk). 通过一系列风险表达式来描述风险值之间的相对关系,由以下语法单元组成: 风险变量、风险表达式和风险评价序列. 风险变量是根据攻击序列和安全状态计算出来的,个数和攻击个数相同,为 \$r1, \$r2, \$r3 等,预定义的变量表示可以直接引用变量, \$MAX 表示计算出来的一系列风险值中最大值, \$MIN 表示计算出来的风险值中的最小值, \$AVG 表示均值. 风险表达式通过操作符将风险变量组合起来,例如 \$r1>\$r2 或 \$r1<\$r3. 在 Risk 和 State 中都可以使用的操作符如表 2 所示.

表 2 风险描述规则的操作符

操作符	含义
>	>大于
<	<小于
>>	>>远大于,即一个风险变量的值,大于另外一个的 2 倍
<<	<<远小于,即一个风险变量的值,不到另外一个的一半
==	==等于
>=	>=大于等于
<=	<=小于等于
&&	与,逻辑操作符将多个表达式组合起来
	或,逻辑操作符将多个表达式组合起来
+	对风险变量的值进行算术+运算
-	对风险变量的值进行算术-运算

风险评价由一系列的风险评价表达式组成,例如对于 3 个连续的攻击,可能的风险评价为 \$r1-3<\$r2, \$r2==\$MAX, \$r3<<\$r2. 表示第 1 个攻击产生的风险值小于第 2 个,第 2 个攻击产生的风险值是最大的,第 3 个风险值远小于第 2 个.

风险描述规则和染色适应度之间可以建立直接的数值关系. 对于一条风险描述规则,通过代入

*Trans* 和 *Obs* 矩阵用 HMM 方法计算该规则中的判断条件是否成立. 本文采用的是将 state 和 risk 部分中的每一个判断条件作为计分依据,如果结果符合,计分为 1,如果不符合,计分为 0. 那么一条规则有  $n$  个判断条件,符合的条件为  $g$  个,那么相应染色体适应度为

$$fitness=\frac{g}{n}$$
 (6)

进一步,可以将多个风险描述规则放到一起,组成一个风险描述规则库,然后共同检验一个染色体的适应度,看该染色体是否同时符合更多的攻击场景. 假设有  $w$  条规则,这时总的适应描述度定义为式(7),通过定义风险描述规则,就可以计算出一个染色体的优秀程度,判断其基因遗传到下一代染色体中的概率.

$$allFit=\sum_{i=1}^w fitness_i$$
 (7)

本文的研究和实验中,风险描述规则从一个蜜网(honeynet)<sup>[14]</sup>的 IDS 告警中得到,由于蜜网环境容易受到攻击,因此初始状态向量 *Init* 取 (0.7, 0.1, 0.1, 0.1),代价向量 *C* 为 (1,10,20,50),如果在其他环境中,可以根据网络环境进行适当调整. 从蜜网的 Snort 日志中取出有代表性的一个片断,然后转化成为风险描述规则. 在图 2 中,截取了对蜜网中 NetBIOS 服务器进行攻击的片断,攻击者利用 IPC\$进行扫描并尝试缓冲区溢出,对于这样一个比较典型的攻击场景,转化为一个风险描述规则,该规则认为攻击对于服务器风险会依次增加,第 5 个攻击对服务器的威胁达到最大,所以得到的风险相对值为 \$r1<\$r5, \$r2<\$r5, \$r3<\$r5, \$r4<\$r5. 同时由于安全状态的不确定性,所以定义了比较宽松的条件,例如当遭到第 5 个攻击后,可能处于 Probed 和 Attacked 状态,即 \$s5==P || \$s5==A.

通过遗传算法,我们得到满足这个规则和其他规则的一个优化的 *Trans* 和 *Obs* 矩阵,在图 3 中描述了这两个矩阵. 在图 3 中也列出了使用 *Trans* 和 *Obs* 矩阵计算这个规则的中间结果,得到 5 个攻击带给服务器的风险值为 \$r1=23.11087, \$r2=23.59597, \$r3=23.830004, \$r4=23.80352, \$r5=28.21422,平均风险值 \$AVG 为 24.510916,这主要是由于蜜网中的主机打开了 NetBIOS 服务,告警的可信度很高. 同时风险值也符合该规则中定义的逐步上升的定义,而服务器的状态也处于 P,P,P, P,P,A 状态.

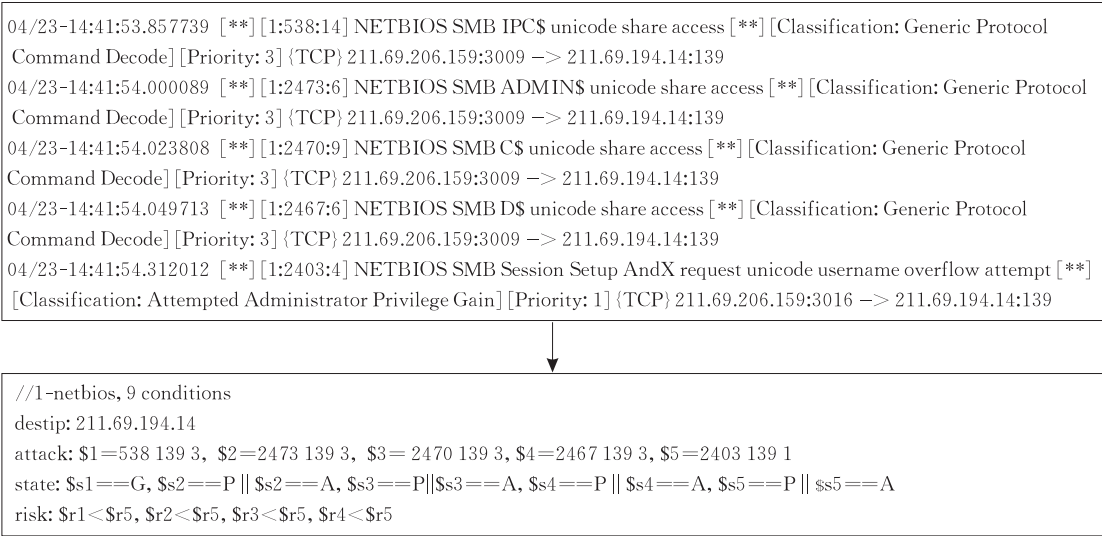


图 2 原始 IDS 告警转换为风险描述规则

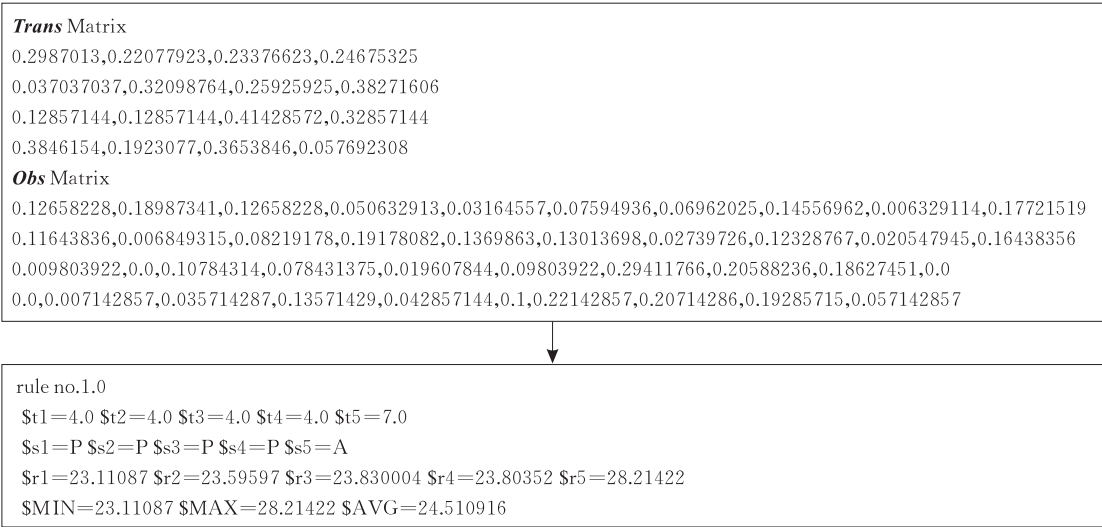


图 3 优化的矩阵和中间计算结果

为了实验和测试的需要,针对蜜网的环境和受到的攻击,本文建立了一个风险描述规则库,一共包含 29 条规则,规则主要分为两大部分:特定攻击场景和普通攻击序列.特定攻击场景是针对一组有前后联系的攻击,例如 NetBIOS 攻击或 RPC 攻击,一般包含扫描、溢出、越权访问等多个步骤,通过比较各个步骤的威胁程度来得到主机遭受攻击时风险值发生的变化;而另外一部分来自于普通的攻击序列,序列中的几个攻击并没有一定的逻辑关系,仅仅针对同一台主机,主要通过分析单个攻击对主机造成的影响建立规则,目的是匹配网络中大多数风险并不严重的情况.具体每条规则的描述如表 3 所示.

表 3 风险描述规则库

规则序号	规则描述	攻击数目
1-netbios	利用 netbios 访问远程共享,并试图缓冲区溢出	5
2-finger	利用 finger 获得信息,并试图通过漏洞越权访问	7
3-web misc	访问 Web 站点特殊文件,看是否存在信息泄漏	3
4-ftp	利用 ftp 匿名登陆网站,并试图访问特殊文件	5
5-netbios SMB-DS	振荡波蠕虫 W32.Sasser,利用 Lsass.exe 漏洞进行传播	5
6-netbios path overflow	利用 netbios 的路径中 overflow 漏洞进行攻击	4
7-netbios unicode access	访问 netbios 共享资源,可看做扫描或者非法访问	6
8-codered worm	红色代码蠕虫试图执行 Web server 上的 root.exe	4
9-RPC info disclosure	对 RPC 服务进行扫描,例如 rstatd,mountd,sadmind 的端口信息	10
10~29	非攻击场景,而是出现频繁的攻击序列	4~8

两类规则的典型例子如图 4,图 4 左侧包含了由攻击场景得到的第 3 和第 4 条规则,而右侧显示

了由普通攻击序列得到的第 15 和第 16 条规则.

<pre>//rules of attack scenario //3-WEB-MISC destip: 211.69.194.13 attack: \$1=1122 80 2, \$2=1147 80 2, \$3=1113 80 2 state: \$s1==G, \$s2==P  \$s2==A, \$s3==P   \$s3==A risk: \$r1&lt;&lt;\$r2, \$r1&lt;&lt;\$r3  //4-ftp destip: 211.69.194.13 attack: \$1=1672 21 2, \$2=553 21 3, \$3=1992 21 3, \$4=336 21 2, \$5=334 21 2 state: \$s1==G, \$s2==P    \$s2==A, \$s3==G    \$s3==P, \$s4==P    \$s4==A, \$s5==G    \$s5==P risk: \$r1&lt;&lt;\$r2, \$r2&gt;&gt;\$r3, \$r3&lt;&lt;\$r4, \$r4&gt;&gt;\$r5</pre>	<pre>//rules of normal attacks //15 destip: 211.69.194.13 attack: \$1=1746 111 2, \$2=1390 38093 1, \$3=1228 1 2 state: \$s1==G, \$s2==G    \$s2==P, \$s3==G    \$s3==P risk: \$r1&lt;\$r2, \$r2&gt; \$r3, \$r1&lt;\$r3, \$r2== \$MAX, \$r1== \$MIN  //16 destip: 211.69.194.13 attack: \$1=1147 80 2, \$2=1847 80 2, \$3=360 21 2 state: \$s1==P    \$s1==A, \$s2==P    \$s2==A, \$s3==G    \$s3==P risk: \$r1&lt;\$r2, \$r1&lt;\$r3, \$r3== \$MIN</pre>
---	---

图 4 部分风险描述规则

4.3 杂交和变异运算策略

对于杂交和变异策略,算法采用了通常的轮盘赌方法,即计算每一个候选染色体的适应度,然后得到一个总体的适应度,在总体适应度的范围内取一个随机数,然后适应度大于该随机数的染色体,能够将基因遗传到下一代.而变异算法,则是设置一个阈值,通过对一个新基因的所有 bit 进行测试,产生一个新的随机数,看随机数是否超过阈值,如果超过阈值那么将该 bit 翻转.

在本算法中,通过多次实验,发现杂交率设置得较高会产生较好的结果,容易产生优化结果,而杂交率低的话,通常超过 500 代的遗传也得不到很好的结果,因此本文取了 0.7 这样一个较大的杂交率.

如果变异率设置较高,计算结果会很不稳定,即使遗传的世代次数达到 300 也很难产生合适的结果,这主要是由于变异对染色体影响的方向是随机的,因此本文取一个非常低的阈值 0.002. 如果取更低的值,会导致染色体的变化较少,趋向某个特定序

列,很难继续优化.

5 实 验

5.1 优化的 *Trans* 矩阵和 *Obs* 矩阵效果测试

为了测试优化的 *Trans* 矩阵和 *Obs* 矩阵的效果,本实验如下设计:利用构建的风险描述规则库,将风险描述规则分为两组,一组是训练集,一组是测试集;将训练集中的风险描述规则作为目标,设定适应度为 0.95,可以得到一组 *Trans* 矩阵和 *Obs* 矩阵,然后利用这两个矩阵,计算测试集能够得到的适应度,如果得到的适应度高,说明这两个矩阵可以适用于新的规则,有推广使用的价值.风险描述规则库一共有 29 条规则,那么分别设定训练集规则数目为 1,2,3,...,28,当然,此时的测试集规则数目分别是 28,27,26,...,3,2,1,对每种情况进行测试,得到图 5 结论.

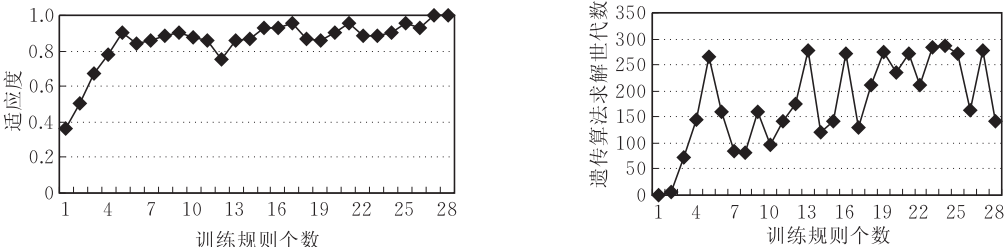


图 5 适应度效果测试

在图 5 的左侧,可以清楚地看到,训练规则超过 5 条,对测试规则就有了接近 0.9 的良好适应度.随着训练集数目的不断增加,测试集数目的减少,适应度会逐步增长,最后达到最高的 1.0. 在增长过程中,会出现一些适应度特别不好的结果,例如用 13

条规则训练的时候,得到的 *Trans* 和 *Obs* 矩阵就较差,对 16 条测试规则的适应度为 0.75,主要是因为用本方法得到的结果有一定的随机误差,可以通过多次运行选择最好的结果来解决.图 5 右侧为测试过程中,为了得到优化结果遗传算法需要运行的世

代数,训练规则超过 6 条以后,常常不能够在一次运算中得到结果,需要多次运算,这和遗传算法随机求解方向的特性有关,一旦正确的求解方向确定,就可以较快地得到结果.

5.2 随机、手工、优化 *Trans* 矩阵和 *Obs* 矩阵比较测试

参考文献[7]要求管理人员手工配置 *Trans* 和

$$Trans = \begin{bmatrix} 0.6 & 0.3 & 0.09 & 0.10 \\ 0.3 & 0.4 & 0.25 & 0.25 \\ 0.1 & 0.2 & 0.6 & 0.10 \\ 0.01 & 0.09 & 0.1 & 0.8 \end{bmatrix},$$

$$Obs = \begin{bmatrix} 0.45 & 0.45 & 0.02 & 0.02 & 0.02 & 0.01 & 0.01 & 0.01 & 0.005 & 0.05 \\ 0.05 & 0.05 & 0.35 & 0.35 & 0.05 & 0.05 & 0.04 & 0.04 & 0.01 & 0.01 \\ 0.02 & 0.03 & 0.04 & 0.05 & 0.25 & 0.25 & 0.15 & 0.15 & 0.05 & 0.01 \\ 0.005 & 0.005 & 0.01 & 0.01 & 0.01 & 0.02 & 0.02 & 0.02 & 0.45 & 0.45 \end{bmatrix}.$$

然后随机抽取 15 条风险描述规则,适应度设置为 0.97,训练出一个优化的 *Trans* 矩阵和 *Obs* 矩

$$Trans = \begin{bmatrix} 0.25 & 0.2916 & 0.2604 & 0.1979 \\ 0.1951 & 0.0731 & 0.7317 & 0.0 \\ 0.2 & 0.42 & 0.18 & 0.2 \\ 0.1512 & 0.1463 & 0.1073 & 0.24 \end{bmatrix},$$

$$Obs = \begin{bmatrix} 0.1512 & 0.1463 & 0.1073 & 0.1512 & 0.0487 & 0.078 & 0.0536 & 0.1024 & 0.0731 & 0.0878 \\ 0.0344 & 0.062 & 0.1103 & 0.1310 & 0.1034 & 0.2 & 0.062 & 0.1241 & 0.1103 & 0.062 \\ 0.0316 & 0.0189 & 0.1962 & 0.1772 & 0.0063 & 0.1582 & 0.1265 & 0.0569 & 0.1202 & 0.1075 \\ 0.0 & 0.0135 & 0.2027 & 0.081 & 0.0675 & 0.1216 & 0.1013 & 0.1261 & 0.2094 & 0.0405 \end{bmatrix}.$$

然后每一次测试,都随机产生一个 *Trans* 矩阵和 *Obs* 矩阵作为参照.通过这三组 *Trans* 矩阵和 *Obs* 矩阵,测试在 1,2,3,...,29 条风险描述规则的情况下各自得到的适应度,并比较这三种配置方法产生的效果.

在图 6 中,优化的 *Trans* 和 *Obs* 矩阵取得了非常好的效果,整个测试过程保持了很高的适应度,而随机产生的 *Trans* 和 *Obs* 矩阵除了较大的波动起伏,适应度平均为 0.5,主要原因是风险描述规则各自的独立性,使得测试结果基本保持在这个范围.而

*Obs* 矩阵,为了保证本文得到的优化 *Trans* 矩阵和 *Obs* 矩阵比随机配置和手工配置的矩阵有更好的效果,本实验仍然利用风险描述规则库来测试这 3 种情况下得到的适应度,并作为和参考文献[7]的比较实验.测试过程是找到一个经验丰富,对蜜网系统和 IDS 都很熟悉的网络管理员,手工配置一个 *Trans* 矩阵和 *Obs* 矩阵,作为一个测试项目,具体如下:

阵,具体如下(只显示小数点后 4 位):

手工设置的矩阵在规则数较少的时候效果较好,这主要是因为网络管理人员对少数几条规则还是能够比较清楚地判断参数范围,但是一旦规则数目较多,适应度就急剧下降,最终的效果接近随机设置的矩阵.这表明网络管理人员很难手工处理复杂情况.

5.3 DARPA 2000 数据集测试

模拟测试验证了优化 *Trans* 矩阵和 *Obs* 矩阵的效果.但脱离蜜网的环境后,本方法是否还能得到较好的结果,需要进一步的测试.本文利用实际的 IDS 数据来检测本方法的有效性. Lincoln 实验室的 DARPA 2000 数据集是在一个拥有 4 个 C 类子网的网络中采集流量得到的,还包括 Solaris BSM 主机的系统日志.这些数据已经被 Snort IDS 和 USTAT 主机 IDS 处理成为告警.该数据集包含了监视子网 172.16.112.0/24,172.16.113.0/24,172.16.114.0/24 和 172.16.115.0/24 的两台 Snort IDS 产生的告警,主要攻击对象是 Mill (172.16.115.20), Pascal (172.16.112.50)和 Locke(172.16.112.10)这 3 台

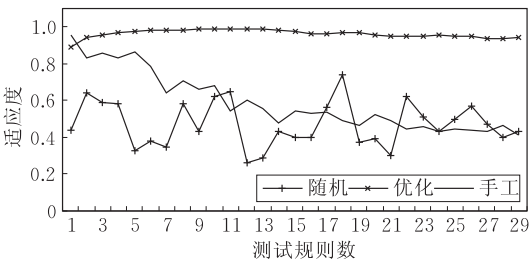


图 6 随机、手工、优化的 *Trans* 和 *Obs* 矩阵比较

主机. 告警主要反映了一个复杂的多步 DDoS 攻击, 该攻击分为 5 步, 包括 IP sweep, Sadmind ping, Break into Mill, Pascal 和 Locke, Install of DDoS tools on Mill, Pascal 及 Locke, Outbound DDoS. 除了 5 步攻击还有一些背景攻击. 在处理 DARPA 2000 数据的时候, 对于同一个 Signature ID 的攻击, 如果在一个较短的时间内, 我们认为其产生的威胁相同, 因此采用了聚类技术将这些攻击聚合起

$$Trans = \begin{bmatrix} 0.415 & 0.2075 & 0.1886 & 0.1886 \\ 0.3294 & 0.3529 & 0.2117 & 0.1058 \\ 0.2307 & 0.123 & 0.3692 & 0.2769 \\ 0.3541 & 0.2083 & 0.3958 & 0.4166 \end{bmatrix},$$

$$Obs = \begin{bmatrix} 0.1694 & 0.1525 & 0.1525 & 0.096 & 0.0621 & 0.0677 & 0.0677 & 0.0056 & 0.0508 & 0.1751 \\ 0.1925 & 0.0888 & 0.1111 & 0.1555 & 0.1851 & 0.0222 & 0.1111 & 0.0666 & 0.0074 & 0.0592 \\ 0.1733 & 0.0066 & 0.04 & 0.12 & 0.0866 & 0.1666 & 0.16 & 0.04 & 0.16 & 0.04666 \\ 0.0134 & 0.0134 & 0.1140 & 0.0671 & 0.0402 & 0.0939 & 0.1476 & 0.1006 & 0.2013 & 0.208 \end{bmatrix}.$$

通过重放 DARPA 告警以及构造相关主机信息, 测试结果如图 7, 列出了所有被聚合后攻击的威胁度. 可以看到真正对网络具有很大风险的攻击数目其实很少, 例如 *threat* 值为 7 的攻击仅仅有 18 个, 达到 8 的只有 3 个, 大部分是 *threat* 为 3 的攻击. 但是少量的攻击对网络造成了较大的影响, 例如 3 个高威胁度的攻击就攻破了 3 台主机, 因此通过 *threat* 值的计算, 准确地发现了高风险的攻击.

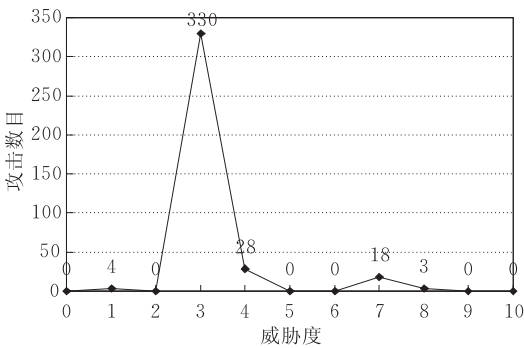
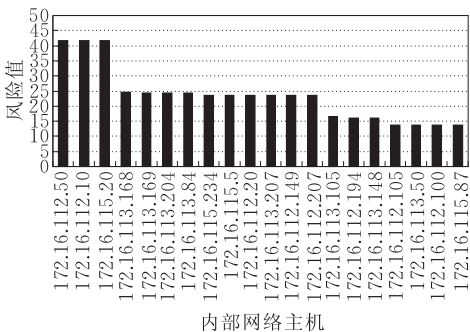


图 7 攻击的威胁度分布



来<sup>[15]</sup>, 聚合后的攻击数目减少, 结果也更加清晰. 为了测试该数据集, 初始向量和代价向量的设置参考了文献[7]提出的原则, 具体设置如下:

初始向量: 0.7, 0.1, 0.1, 0.1;

代价向量: 1, 15, 30, 50.

用 29 条风险描述规则训练得到的适应度为 0.97 的 *Trans* 和 *Obs* 矩阵(只显示小数点后四位)如下:

而图 8 的左侧图形记录了所有内部网络被攻击的主机其平均风险值, 其中可以清楚地看到, 被攻击并安装了 DDoS Tools 的 3 台主机风险值明显高于其他主机. 另外对于 IP 地址为 172.16.115.20 的主机 Mill, 其风险值的变化如图 8 的右侧, 风险明显上升, 由于最后没有新的攻击, 一直处于一个高风险状态. 图 8 表明本文方法正确地反映了网络的风险情况.

## 6 总结和未来工作

实时量化网络安全风险评估方法为网络管理员提供了强有力的工具, 可以方便监控网络状态, 发现安全焦点, 并及时进行改进. 本文主要对基于 HMM 的方法进行了改进和优化, 降低了该方法的使用难度, 同时为参数提供了一种可靠的设置和评价方法, 也设计了一种通用的攻击和风险关系描述规则. 在研究过程中发现, 威胁度算法不仅仅可以起到对攻

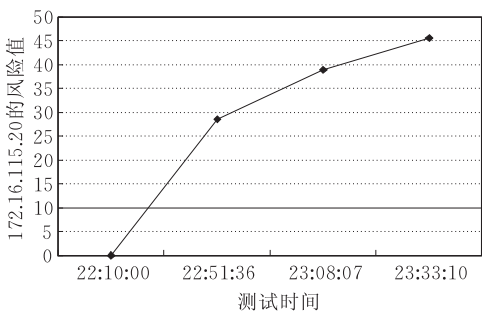


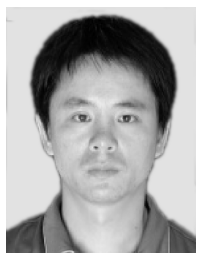
图 8 DARPA 2000 测试集中的网络风险值分布

击分类的作用,通过对威胁度的排序可以降低 IDS 的误报,减少管理员对无关攻击的关注,因此下一步可以继续对威胁度算法进行修改和测试,并可以和 IDS 系统直接集成. 风险描述规则在研究中发现具有直观易用特点,并能够对攻击和风险进行形式化描述,建立的风险描述规则库也可以在不同安全风险量化方法中使用,因此下一步的工作是希望建立一个通用的风险描述规则库,同时在实际复杂网络中进行测试,使之成为一个可以复用的方法. 最后,本文对于网络安全风险的计算是直接来自主机风险相加得到的,这种做法简化了网络的实际情况,很多网络中不同主机对于风险敏感程度是不一样的. 参考文献[16]也提出了定义主机关键资产的方法,但是其配置过程过于复杂,而且是静态配置,不符合实时评估的要求. 另外文献[12]也研究了网络中主机风险会互相关联的问题. 如果再考虑存在的网络之间信任关系例如 Windows 域信任关系,使网络风险互相影响,导致更加复杂的情况出现. 所以这个问题希望能够在下一步工作中进行深入研究.

### 参 考 文 献

- [1] Howard M, Pincus J, Wing J M. Computer Securing in the 21st Century. Springer, 2005: 109-137
- [2] Ortalo R, Deswarte Y, Kaâniche M. Experimenting with quantitative evaluation tools for monitoring operational security. IEEE Transactions on Software Engineering, 1999, 25(5): 633-651
- [3] Ammann P, Wijesekera D, Kaushik S. Scalable, graph-based network vulnerability analysis//Proceedings of the 5th ACM Conference on Computer and Communications Security. Washington DC, USA, 2002: 217-224
- [4] Gehani A, Kedem G. Rheostat: Real-time risk management//Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection. French Riviera, France, 2004: 296-314
- [5] Jonsson E, Olovsson T. An empirical model of the security intrusion process//Proceedings of the 11th Annual Conference on Computer Assurance. Gaithersburg, 1996: 176-186
- [6] Jonsson E, Olovsson T. A quantitative model of the security intrusion process based on attacker behavior. IEEE Transactions on Software Engineering, 1997, 23(4): 235-245

- [7] Arnes A, Valeur F, Vigna G et al. Using hidden Markov models to evaluate the risk of intrusions//Proceedings of the RAID'06. Hamburg, Germany, 2006: 145-164
- [8] Haslum Kjetil, Årnes André. Multisensor real-time risk assessment using continuous-time hidden Markov models//Proceedings of the International Conference on Computational Intelligence and Security (CIS). Guangzhou, China, 2006: 694-703
- [9] Chen Xiu-Zhen, Zheng Qing-Hua, Guan Xiao-Hong, Lin Chen-Guang. Quantitative hierarchical threat evaluation model for network security. Journal of Software, 2006, 17(4): 885-897(in Chinese)  
(陈秀真, 郑庆华, 管晓宏, 林晨光. 层次化网络安全威胁态势量化评估方法. 软件学报, 2006, 17(4): 885-897)
- [10] Lu Yu-Liang, Xia Yang. Research on target-computer secure quantitative fusion model. Chinese Journal of Computers, 2005, 28(5): 914-920(in Chinese)  
(陆余良, 夏阳. 主机网络安全量化融合模型研究. 计算机学报, 2005, 28(5): 914-920)
- [11] Wang Yi-Feng, Li Tao, Hu Xiao-Qin, Song Cheng. A real-time method of risk evaluation based on artificial immune system for network security. Acta Electronica Sinica, 2005, 33(5): 945-949(in Chinese)  
(王益丰, 李涛, 胡晓勤, 宋程. 一种基于人工免疫的网络安全实时风险检测方法. 电子学报, 2005, 33(5): 945-949)
- [12] Zhang Yong-Zheng, Fang Bin-Xing, Chi Yue, Yun Xiao-Chun. Research on network node correlation in network risk assessment. Chinese Journal of Computers, 2007, 30(2): 234-240(in Chinese)  
(张永铮, 方滨兴, 迟悦, 云晓春. 网络风险评估中网络节点关联性的研究. 计算机学报, 2007, 30(2): 234-240)
- [13] Holsopple Jared, Yang Shanchieh Jay, Sudit Moises. TANDI: Threat assessment for network data and information//Proceedings of the SPIE Defense and Security Symposium. Orlando FL, 2006: 624200.1-624200.12
- [14] Riebach S, Toedtman B, Rathgeb Erwin P. Risk assessment of production networks using honeynets-some practical experience//Proceedings of the ISPEC05 Conference. Singapore, 2005: 1-12
- [15] Debar H, Wespi A. Aggregation and correlation of intrusion-detection alerts//Proceedings of the RAID'01. USA, 2001: 87-105
- [16] Porras P, Fong M, Valdes A. A mission-impact-based approach to INFOSEC alarm correlation//Proceedings of the 5th International Symposium on Recent Advances in Intrusion Detection (RAID 2002). Zurich, Switzerland, 2002: 95-114



**LI Wei-Ming**, born in 1975, Ph.D., lecturer. His main research interests include network architecture and network security.

**LEI Jie**, born in 1983, Ph. D. candidate. His main research interests focus on network security.

**DONG Jing**, born in 1984, M. S. candidate. Her main research interests focus on network security.

**LI Zhi-Tang**, born in 1953, professor, Ph. D. supervisor. His main research interests include network architecture, P2P network and network security.

Background

If you can't quantify something, you can't control it. The same is network security risk. So, Security assessment of a network is the key to improving the security level of the network. There are many methods to quantify the network security risk, but they also have many faults. The Hidden Markov Model based method, result of lately research, is real time, dynamically and timely, but it is also complex to configure and it tends to acquire errors. These faults are resolved in an optimized method presented in this paper. The optimized method improves the accuracy and simplifies the configuration with automatically calculate matrixes in HMM. It presents the way to define the threat of an attack and uses the genetic algorithm to generate the HMM status transfor-

mation matrix and observation matrix automatically. The most important innovation is that it defines risk description rules as the genetic algorithm optimization target. The risk description rule provides a formal method to characterize the network security risk, and the rule base can be used as the test criterion for other risk assessment methods. Several experiments are provided in the paper to verify the effect of this method. The work is supported in part by the National Natural Science Foundation of China under grant No.60573120, "The P2P Network Key Security Problems Research". This paper is to resolve the problem in the project that whether a network is security and its security risk level.