基于攻击图的多 Agent 网络安全风险评估模型

杨宏宇 江 华
(中国民航大学计算机科学与技术学院 天津 300300)

摘 要 为了自主保障计算机网络的安全并对网络安全风险进行自动化评估,提出一种基于攻击图的多 Agent 网络安全风险评估模型(Multi-agents Risk Evaluation Model Based on Attack Graph,MREMBAG)。首先提出网络风险评估模型,设计了主从 Agent 的功能架构和关联关系分析流程。利用全局攻击图生成算法,以动态数据信息作为输入,通过主从 Agent 协同分析并构建攻击路径。基于对目标网络的攻击路径、组件、主机、网络的风险指数、漏洞及关联风险指数的计算,获取目标网络的安全风险指标。仿真实验结果验证了该评估方法的可行性和有效性。

关键词 网络安全,风险评估,多 Agent,攻击图
中图法分类号 TP393.08 **文献标识码** A

Multi-Agents Network Security Risk Evaluation Model Based on Attack Graph

YANG Hong-yu JIANG Hua
(School of Computer Science and Technology, Civil Aviation University of China, Tianjin 300300, China)

Abstract In order to protect the network and evaluate the security risk of network automatically, a novel multi-agents risk evaluation model based on attack graph (MREMBAG) was presented. First, a well-structured model to manage entire evaluation process and the function architecture of primary-slave agents were designed. Then primary-slave agents constructed the attack path and generated the attack graph by using the attract graph building algorithm with the input of the dynamic data information collected by components. Finally, the risk indexes of attack path, components, hosts, the vulnerabilities and nodes correlation risk indexes were determined to calculate the target network quantitatively. The experimental results demonstrate that the MREMBAG is a more practical and efficient way to evaluate the network security risk.

Keywords Network security, Risk evaluation, Multi-Agents, Attack graph

1 引言

网络安全风险评估技术是信息安全领域的研究热点之一,但现有网络安全评估技术的准确性过于依赖人员能力和经验,缺乏自主性[1];基于静态数据分析的网络安全风险评估方法难以发现网络运行过程中存在的威胁状况,缺乏实时性;攻击图生成过程存在的网络状态组合爆炸问题,增加了攻击图分析工作的难度。

Phillips 和 Swiler 在 1998 年首次提出基于攻击图的网络安全分析方法[2],这种方法首先建立入侵模式库以及网络安全漏洞威胁库,并在此基础上构造网络入侵关系图。Ammann 等人[3]使用一种基于图论的方法生成攻击图,该方法的空间复杂性和时间复杂性要优于模型检测方法,但是攻击图的状态组合爆炸问题仍然不能得到有效控制。文献[4-8]研究采用模型检测器自动生成攻击图的方法,但是该方法的时间复杂度会随着主机和脆弱性数目的增加呈指数级增长,出现状态空间爆炸问题。

本文基于 Ammann 的研究思路引入了多 Agent 技术[9],提出一种基于攻击图的多 Agent 的网络安全风险评估模型(Multi-agents Risk Evaluation Model Based on Attack Graph,MREMBAG)。该模型采用多 Agent 技术,通过主从 Agent 协同配合动态收集基础数据,采用全局攻击图生成算法生成以组件为基本节点的网络攻击图,通过风险指数算法计算组件、主机、网络的风险指数和漏洞的关联风险指数,从多个角度获取目标网络的安全风险指标。

2 多 Agent 风险评估模型

2.1 风险评估模型

利用攻击图评估网络风险的优越性在于可以获取攻击行为为达到攻击目标所选择的所有可能攻击路径,从而更全面地计算、评估网络所面临的安全风险。基于攻击图的网络安全风险评估工作流程包括 3 个阶段(见图 1)[10]:信息收集、攻

到稿日期:2012-04-15 返修日期:2012-07-21 本文受国家自然科学基金(60776807,61179045),国家 863 计划重点课题(2006AA12A106),天津市科技计划重点项目(09JCZDJC16800),中国民航科技基金(MHRD201009,MHRD201021),中央高校基本科研业务费专项(ZXH2009A006,ZXH2010D009)资助。

杨宏宇(1969—),男,博士,教授,主要研究方向为网络与信息安全,E-mail:yhyxlx@hotmail.com;江 华(1986—),女,硕士生,主要研究方向为网络信息安全。

击图构建、可视化和分析。在信息收集阶段，收集所有对构建攻击图有用的信息，包括关联关系信息、攻击规则信息和漏洞信息，并将其保存到数据库中；在攻击图构建阶段，首先确定目标状态和初始状态，随后确定搜索方向并选择合适的搜索策略，最后确定采用的攻击选择策略，包括是否可以重复攻击、攻击难度或代价是否有所限制、攻击的时间、成功概率与步骤是否有所限制等；在可视化和分析阶段，通过可视化工具展现攻击图，采用相应的理论和分析方法从网络安全或图论的角度，结合节点属性分析攻击图，计算网络风险的量化指数。

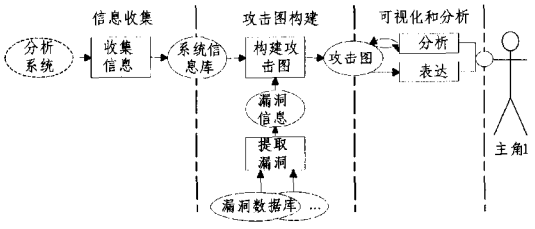

图 1　基于攻击图的网络安全风险评估工作流程

利用攻击图进行网络风险评估的最大优点是：可以对攻击场景进行建模，并对系统所有可能或已经真实存在的攻击行为的路径进行描述，反映攻击动作之间的因果关系。本文基于攻击图的网络安全风险评估工作流程，采用多 Agent 技术设计了一个网络风险评估模型 MREMBAG，如图 2 所示。

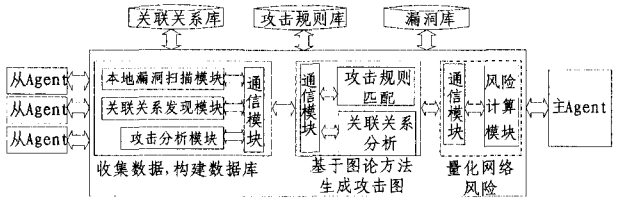

图 2　网络风险评估模型

在 MREMBAG 模型中，采用主 Agent 和从 Agent 自动收集网络漏洞、主机信息和组件信息，通过主从 Agent 的协同配合基于图论方法生成攻击图，而后从多个角度对网络安全风险进行评估。评估过程设计如下：

步骤 1　启动所有代理设备，网络全局主从 Agent 协同构建基础数据，形成漏洞库、攻击规则库和关联关系库。

步骤 2　根据基础数据库，从 Agent 分析网络中可能存在的攻击路径，并计算攻击成功概率。

步骤 3　根据攻击路径分析结果，主 Agent 生成以主机组件为节点、以攻击路径为边的攻击图。

步骤 4　主 Agent 结合攻击成功概率计算组件安全风险，利用主机重要性权重与主机上所有组件安全风险指数量化计算主机安全风险，最后针对主机安全风险指数量化计算网络安全风险。

步骤 5　结合攻击路径成功概率，计算漏洞和关联关系风险指数，提出网络改进措施。

2.2　主从 Agent 设计

多 Agent 系统(Multi-Agent System，MAS)是指由多个可执行网络计算的 Agent 组成的集合，Agent 成员之间相互协调，相互服务，共同完成一个任务。根据风险评估需求，本文设计的多 Agent 系统由主 Agent 和从 Agent 组成，其具体功能和主要模块设计如下：

1)主 Agent：主要部署在专用计算机或性能较好的主机上，负责维护全局网络信息漏洞库、攻击规则库和关联关系库，构建攻击图，计算网络整体安全风险。其具体功能结构图如图 3 所示。主 Agent 由 4 个模块组成：

• 攻击规则匹配模块。主 Agent 首先接收从 Agent 的攻击规则请求，然后分析、返回与从 Agent 所在主机匹配的攻击规则和漏洞量化指标。

• 关联关系分析模块。主 Agent 汇总所有从 Agent 关联关系报告，并分析、返回整理后的关联关系。

• 风险计算模块。主 Agent 依据从 Agent 的分析结果，计算网络中主机组件、主机以及网络整体的安全风险，并对网络中漏洞和关联关系风险指数进行排序。

• 通信模块。从 Agent 之间、主 Agent 之间以及从 Agent 和主 Agent 之间通过发送协同消息实现 Agent 的协同。

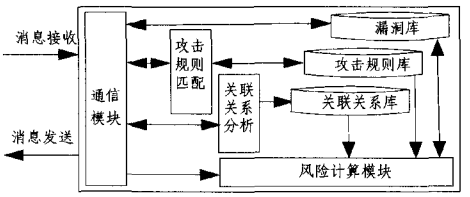

图 3　主 Agent 功能结构图

2)从 Agent：部署在网络中的每台主机上，以主机及组件信息作为输入条件展开评估，动态维护和更新本地漏洞库、攻击规则库以及关联关系库，为网络风险评估提供基础量化数据。从 Agent 功能模块结构图如图 4 所示，主要包括 4 个功能模块：

• 本地漏洞扫描模块。从 Agent 扫描主机上的安全漏洞并向主 Agent 报告，根据主 Agent 的返回数据初始化本地漏洞库和攻击规则库。

• 关联关系发现模块。从 Agent 利用工具，采用自动检测和手动分析相结合的方式发现网络中的关联关系，形成本地关联关系库。

• 攻击分析模块。从 Agent 根据主机的漏洞库和关联关系库，结合相应的攻击规则，对可能发生的非法访问过程进行全面的分析。

• 通信模块。该模块与主 Agent 中的通信模块功能相同。

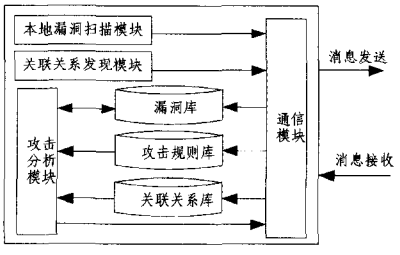

图 4　从 Agent 功能结构图

2.3　基于多 Agent 的数据采集

数据采集阶段的主要任务是通过收集漏洞量化指标、与漏洞相关的攻击规则、与各组件相关的关联关系等信息，来构建本地漏洞库、攻击规则库和关联关系库，为后期的攻击分析和风险评估提供基础数据。在数据采集阶段，部署在各主机上的从 Agent 以漏洞列表作为输入生成本地关联关系库，以主机上各组件 trust 列表和 visit 列表作为输入生成本地关联

关系库，同时维护攻击规则库。可见，数据采集和分析的核心环节是关联关系分析，为此，本文提出了一个关联关系分析流程，如图5所示。

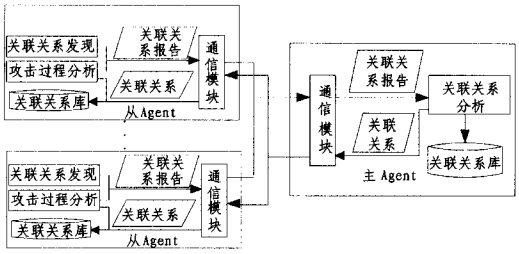

图5　关联关系分析流程图

按照该流程，所有从Agent用工具自动探测和手动分析相结合的方法来发现本地关联关系，向主Agent报告。主Agent对所有从Agent发送的关联关系信息汇总后，将全网的关联关系存入关联关系库，然后分析库中数据，向从Agent返回与其所在主机组件相关的关联关系。一般来说，主机用户很清楚主机各组件对其他服务的访问关系，但未掌握网络全局信息，因此需要主Agent对全网的关联关系汇总分析后，把与每个从Agent所在主机组件相关的关联关系发送给从Agent，供其分析攻击者可能发起的攻击过程。需要注意的是，随着攻击过程分析的进行，会出现由非法入侵等攻击行为引起的关联关系改变，此时需要由通信模块及时通知从Agent和主Agent更新关联关系库。

3　基于攻击图的风险评估算法

3.1　攻击图生成算法

攻击图是实施网络风险评估的基础，本文采用基于图论的方法设计了一个全局攻击图生成算法（见图6），以此生成攻击图。在该算法中，将整个网络系统抽象为一个有向图 $G=\{V,\{R\}\}$，其中，顶点表示主机上的组件节点，如WWW服务、SNMP服务等；V 为顶点的有限非空集合；有向弧连接具有访问关系的相邻顶点，R 为2个顶点之间的关系集合。全局攻击图生成算法的基本思路是：

①根据收集的基础数据库形成初始状态序列，从Agent利用攻击规则检查和关联关系传递分析可能存在的攻击路径；

②主Agent根据从Agent的攻击路径分析结果，比较各攻击路径的风险指数大小，仅保留风险指数最大的攻击路径作为攻击图的边，构建攻击图。

Input：网络主机上的所有组件components，从Agent攻击路径分析结果
Output：网络攻击图
Begin
1）监听从Agent消息；
2）While(消息队列不空){
3）getMessage();// 从消息队列取出一条消息
4）If(消息是从Agent攻击路径报告){
5）If(节点 A_i，B_j 间不存在边 ‖ 带来风险大于已有边的风险指数)更新攻击图节点 A_i 和 B_j 间的边为 E(A_i，B_j，Attseq，P)；
6）If(消息队列为空 || 时间超时)向从Agent发出攻击图构建结束消息；} }
End

图6　全局攻击图生成算法

3.2　网络风险指数计算

在攻击图中，节点代表主机组件，边代表攻击路径。通过风险指数计算算法分别计算攻击路径、组件、主机和网络风险指数，具体计算步骤如下。

步骤1　攻击成功概率计算

一般攻击执行过程需要目标主机具有相应的漏洞或配置，若这些条件不满足，则攻击成功的可能性较低；反之，攻击成功的可能性较高[11]。因此，攻击路径上的漏洞可被成功利用的概率直接影响攻击成功概率。将攻击成功概率定义为 $P(Attack)$，其计算公式为：

$$P(Attack)=\prod_{i=1}^{n} v_i \times p_i \tag{1}$$

式中，v_i 为攻击路径上被攻击者利用的第 i 个漏洞，p_i 为攻击者成功利用第 i 个漏洞的概率。

步骤2　攻击路径风险指数计算

攻击路径是一条包含了攻击者从初始条件到达攻击目标所经历的节点和利用条件的序列。利用条件不仅包括漏洞和关联关系，而且被攻击节点的资产价值和权重对攻击路径也会产生影响。将攻击路径的风险指数定义为 $R(AttSeq)$，其计算公式为：

$$R(AttSeq)=P(Attack)\times C\times W \tag{2}$$

式中，C 为被攻击节点的资产价值，W 为加权系数，表示攻击者攻击成功对组件的危害系数。

步骤3　组件风险指数计算

主机上的组件可能被攻击者从多条路径进行不同程度的攻击，在这些攻击中，把给组件带来最大风险的攻击所造成的风险作为组件的安全风险指数。将组件风险定义为 $R(Component)$，其计算公式为：

$$R(Component)=\max(R(AttSeq_1),R(AttSeq_2),\cdots,R(AttSeq_n)) \tag{3}$$

步骤4　主机风险指数计算

主机安全风险来源于主机上各组件的安全风险。假定主机 A 上包含 n 个组件节点，然后根据网络提供服务的分布情况确定主机节点的权重。将主机安全风险指数定义为 $R(Host)$，其计算公式为：

$$R(Host)=H\cdot\sum_{i=1}^{n} R(Component_i) \tag{4}$$

式中，H 为主机在网络中的重要性系数。

步骤5　网络风险指数计算

网络的整体安全风险指数由网络中各主机的安全指数累加获得。将网络安全风险指数定义为 $R(Network)$，其计算公式为：

$$R(Network)=\sum_{i=1}^{n} R(Host_i) \tag{5}$$

式中，$Host_i$ 为网络上的第 i 个主机。

在整个风险评估过程中，主机上的本地漏洞库、攻击规则库和关联关系库都是实时更新的，所以生成攻击路径的关键是从Agent需要一直监听来自其它从Agent的消息，动态更新漏洞和关联关系，并实时进行攻击路径分析。

3.3　漏洞和关联关系风险指数计算

进行网络安全风险评估的一个重要目的是给网络管理者采取措施增强网络安全性提供建议，因此，本文的网络安全风险评估方法中加入对漏洞及其关联关系风险指数的计算，网络管理者可以根据该风险指数的高低决策优先修补的漏洞和

关联关系。攻击者利用漏洞和关联关系获得非法的访问权限，同一个漏洞或关联关系可能被不同的攻击路径利用，被利用的次数越多，说明此漏洞或关联关系给网络带来的安全威胁的概率越大。因此，用包含某一漏洞或关联关系的所有攻击路径给网络带来风险的累加值作为漏洞或关联关系风险指数，能准确反映其给网络带来的威胁大小。

$$R(Vul)=\sum_{i=1}^{n}R(AttSeq_i) \tag{6}$$

式中，$AttSeq_i$ 为攻击图中包含漏洞 Vul 的所有攻击路径。需要注意的是，漏洞和关联关系的风险指数仅用于对漏洞和关联关系威胁程度的排序，其数值可能比网络整体风险还要大，因此两者并不具有可比性。

4 仿真实验

4.1 数据集与实验平台

为验证 MREMBAG 模型的可行性和有效性，基于文献[12]的网络实验环境，构建了一个实验网络并收集相关数据。在该网络中，设计了网络中各节点的访问控制策略（见表 1）。节点 A 为外网中的一台 PC 机，代表目标网络外的访问用户；节点 B,C,D,E 分别为网络信息服务器、数据库、管理机和个人计算机，它们和防火墙构成内网。

表 1 网络访问控制策略

编号	访问控制策略	关联关系表示
1	节点 B 的 WWW 服务对 A 开放	$NC_{A,anyB,WWW}$=(A, any, root, B, WWW, access, 0)
2	节点 D 的所有 Linux 端口对 A 开放	$NC_{A,anyD,Linux}$=A, any, root, D, Linux, access, 0)
3	节点 B 的 WWW 服务可向 C 的数据库读写信息	$NC_{B,WWW,C,MySql}$=(B, WWW, user, C, MySql, user, 0)
4	节点 B 的 Rsh 服务可监听到本地 WWW 服务的数据流	$NC_{B,RshB,WWW}$=(B, Rsh, user, B, WWW, access, 0)
5	节点 D 通过 Rsh 服务管理节点 B	$NC_{D,LinuxB,Rsh}$=(D, Linux, user, B, Rsh, root, 0)
6	节点 D 通过 Snmp 服务管理节点 B	$NC_{D,LinuxB,Snmp}$=(D, Linux, user, B, Snmp, root, 0)
7	节点 D 上的用户可以远程登录 E	$NC_{D,LinuxE,Windows}$=(D, Linux, user, E, Windows, user, 0)
8	节点 E 的管理员用户可向 C 的数据库中读写信息	$NC_{E,WindowsC,MySql}$=(E, Windows, root, C, MySql, user, 0)

由于 JADF(Java Agent Development Framework)是一个多 Agent 系统开源开发框架，具有跨平台的优点，因此，基于 JADF 平台并采用 Java 语言实现了 MREMBAG 原型系统。MREMBAG 原型系统中所实现的主从 Agent 的 UML 类图如图 7 所示。

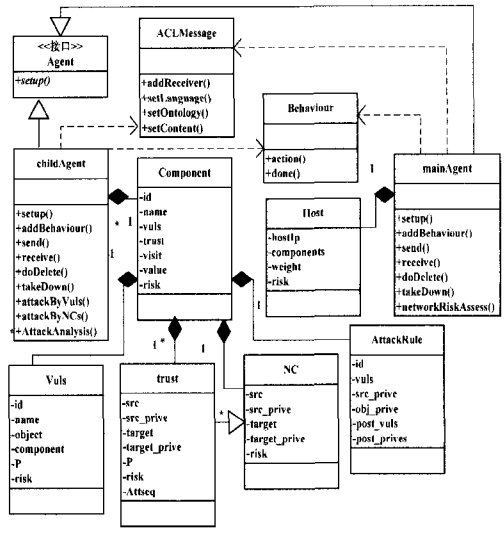

图 7 主从 Agent 的 UML 类图

4.2 实验结果与分析

通过 MREMBAG 原型系统中主从 Agent 协同运行全局攻击图生成算法，利用制图工具 GRAPHVIZ 生成网络攻击图（见图 8）。图 8 标识了与非法入侵相关的主机组件，其中节点 B 的 WWW 组件、节点 B 和 C 的操作系统组件均未标出。主机组件节点之间用有向边连接，表示在一种攻击行为下由一种状态向另一种状态转移。其中，曲线表示是非法的访问关系或一条攻击路径，直线则表示被攻击者利用的合法关联关系。

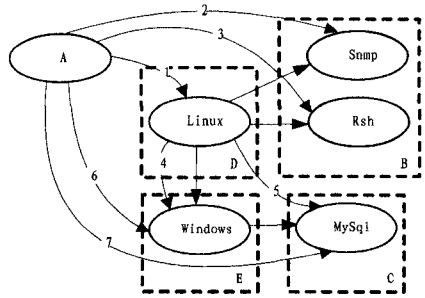

图 8 网络攻击图

主 Agent 采用自上而下的评估方法，依次对图 8 中的每条攻击路径、组件节点、主机节点和整个网络风险指数进行量化计算，对漏洞和关联关系风险指数进行计算排序，风险计算结果如表 2 所列。

表 2 风险指数计算结果

网络风险	主机风险		组件风险		漏洞风险		关联关系风险	
	主机	风险	组件	风险	漏洞	风险	关联关系	风险
27.86	B	16.2	WWW	0	主机 D 上 Linux 组件缓冲区溢出漏洞	39.4	节点 D 的所有 Linux 端口对 A 开放	39.4
			Snmp	6			节点 D 通过 Snmp 服务管理节点 B	6
			Rsh	12				
	C	7.84	Mysql	11.2	主机 E 上 Windows 组件的 SMB 漏洞	26.95	节点 D 通过 Rsh 服务管理节点 B	12
	D	3	Linux	0			节点 D 上的用户可以远程登录 E	26.95
			Linux	6			节点 E 的管理员用户可向 C 的数据库中读写信息	19.6
	E	0.82	Windows	4.2				

根据表 2 中的风险指数计算结果，可看出实验网络中的网络信息服务器 B 的风险指数最高，可能给网络造成最大风险的漏洞为主机 D 上的 Linux 组件缓冲区溢出漏洞，风险指数最大的关联关系为节点 D 的所有 Linux 端口对 A 开放。综上，为降低网络安全风险，网络管理员应优先修复管理机 D 上的缓冲区溢出漏洞，并限制节点 D 的所有 Linux 端口对 A

开放。

结束语 网络安全风险量化评估对网络系统的安全保障和主动防护具有重要的现实意义。针对现有风险评估技术存在的自主性不足等问题，本文提出了一种基于攻击图的多 Agent 风险评估模型——MREMBAG。通过在风险评估过程中引入多 Agent 技术并采用全局攻击图生成算法自动生成网络攻击图，依据该攻击图计算攻击路径、组件、主机、网络的风险指数和漏洞及其关联关系风险量化指标，通过计算和分析获取目标网络的安全风险指标。实验结果表明，MREMBAG 模型为解决网络安全风险的量化评估问题提供了一个可行、有效的方法。

在未来的研究中，将以 MREMBAG 模型为基础并综合考虑已有安全措施及管理因素对网络风险的影响，通过利用网络数据对评估模型和评估方法进行改进，从而进一步完善评估效果。

参 考 文 献

[1] 江常青. 信息安全评估需要研究和解决的几个关键问题[J]. 国家信息安全测评认证，2007(5)：1-4

[2] Phillips C, Laura S P. A graph-based system for network vulnerability analysis[C]// Proceedings of the 1998 workshop on New security paradigms. VA, USA: ACM Press, 1998: 71-79

[3] Ammann P, Pamula J, Ritchey R, et al. A host based approach to network attack chaining analysis[C]// Proceedings of the 21st Annual Computer Security Applications Conference. Tucson, Arizona, USA: IEEE Computer Society Press, 2005: 72-84

[4] Ramakrishnan C, Sekar R. Model-based vulnerability analysis of computer systems[C]// Proceedings of the 2nd International Workshop on Verification. Pisa, Italy: Model Checking and Abstract Interpretation Press, 1998: 1-8

[5] Ritchey R, Ammann P. Using model checking to analyze network vulnerabilities[C]// Proceedings of the 2000 IEEE Symposium on Security and Privacy. Berkeley, California, USA: IEEE Computer Society Press, 2001: 156-165

[6] Sheyner O. Scenario Graphs and Attack Graphs [D]. School of Computer Science, Carnegie Mellon University, Pittsburgh, USA, 2004

[7] Sheyner O, Haines J, Jha S, et al. Automated Generation and Analysis of Attack Graphs[C]// Proceedings of the 2002 IEEE Symposium on Security and Privacy. Oakland, California, USA: IEEE Computer Society Press, 2002: 254-265

[8] Jha S, Sheyner O, Wing J. Two Formal Analyses of Attack Graphs[C]// Proceedings of the 15th Computer Security Foundations Workshop. Beijing, China: Chinese Academy of Sciences Press, 2002: 49-63

[9] 李冠君. 基于安全代理的网络自保护系统模型研究[D]. 天津：中国民航大学，2009

[10] Roschke S, Cheng F, et al. Towards Unifying Vulnerability Information for Attack Graph Construction[J]. Computer Science, Information Security, 2009(5735): 218-233

[11] 陈天平，许世军，等. 基于攻击检测的网络安全风险评估方法[J]. 计算机学报，2010，37(9)：94-96

[12] 张永铮，方滨兴，迟悦，等. 网络风险评估中网络节点关联性的研究[J]. 计算机学报，2007，30(2)：234-240

(上接第 123 页)

器差异度仍然较低。基分类器个数 $L=400$ 时，差异度仅为 28.2%。最后采用 SRFS 算法，大大提高了基分类器之间的差异度，$L=200$(对每个作者构造 5 个基分类器)时，差异度为 36.8%，当 L 提高到 400(对每个作者构造 10 个基分类器)时，差异度达到了 39.1%，比其他方法最高高出了约 10%，Kappa 值最高能达到 75.6 的最高点，说明识别结果的可信度较高。但值得注意的是，基分类器的数量太大也会降低集成系统的运行效率，虽然增加基分类器个数能一定程度提高集成分类性能，但超过某个阈值会使性能降低，并减小基分类器之间的差异度。

以上实验结果说明了，通过从特征空间中挖掘作者个体书写纹特征并结合随机子空间的划分方法能显著地提高书写纹识别的正确率和鲁棒性，另一方面也体现了基分类器之间的差异度对集成分类器性能具有重要的影响。

结束语 书写纹识别对于匿名网络主体行为的规范化和网络安全的维护具有重要的实际意义，但在中文语境下，目前仍存在较多问题需要进一步解决。针对 N-gram 字符特征集的高稀疏度和噪音数据较多等问题，提出一种基于作者个体特征选择和随机子空间的集成分类方法，其充分利用特征空间中的个体鉴别信息来提高基分类器之间的差异度，降低集成分类系统对噪音数据的敏感度。实验结果表明，该算法能有效提高书写纹的识别性能。下一步将就作者个体特征选择对识别性能的理论依据展开进一步的研究，并结合对样本空间的划分机制设计更加优化的中文书写纹识别算法。

参 考 文 献

[1] 索绪尔 F，等. 普通语言学教程[M]. 高名凯，译. 北京：商务印书局，1980

[2] Li J, Zheng R, ChenH. From fingerprint to writeprint[J]. Communications of the ACM, 2006, 49(4): 76-82

[3] Zheng R, Li J, Chen H, et al. A framework for authorship identification of online messages: writing style features and classification techniques[J]. Journal of the American Society of Information Science and Technology, 2006, 57(3): 378-393

[4] Zhao Y, Zobel Y. Effective and scalable authorship attribution using function words[C]// Proceedings of the 2nd Asian Information Retrieval Symposium. Berlin: Springer, 2005: 174-189

[5] 武晓春，黄萱菁，吴立德. 基于语义分析的作者身份识别方法研究[J]. 中文信息学报，2006，20(6)：61-68

[6] Stamatatos E. A survey of modern authorship attribution methods[J]. Journal of the American Society of Information Science and Technology, 2009, 60(3): 538-556

[7] Houvardas J, Stamatatos E. N-gram feature selection for authorship identification[C]// Proceedings of the 12th International Conference on Artificial Intelligence: Methodology, Systems, Applications. Berlin: Springer, 2006: 77-86

[8] Stamatatos E. Authorship attribution based on feature set subspacing ensembles[J]. International Journal on Artificial Intelligence Tools, 2006, 15(5): 823-838

[9] Koppel M, Schler J, Argamon S. Authorship attribution in the wild[J]. Language Resources and Evaluation, 2010, 45(1): 83-94

[10] 孙建文，杨宗凯，刘三妍，等. 基于集成学习与遗传算法的网络书写纹识别研究[J]. 计算机科学，2011，38(6)：242-245

[11] Ho T. The random subspace method for constructing decision forests[J]. IEEE Trans. on PAMI, 1998, 20(8): 832-844

[12] 韩宏，杨静宇. 多分类器组合及其应用[J]. 计算机科学，2000，27(1)：58-61

[13] Kuncheva L, Wgitaker C. Measures of diversity in classifier ensembles[J]. Machine Learning, 2003, 51(2): 181-207

[14] 顾亚祥，丁世飞. 支持向量机研究进展[J]. 计算机科学，2011，38(2)：14-17

[15] Breiman L. Bagging predictor [J]. Machine Learning, 1996, 24(2): 123-140