计算机科学 2007Vol. 34№. 10

计算机主机及网络脆弱性量化评估研究

夏 阳 陆余良

(解放军电子工程学院网络工程系 合肥 230037)

摘 要 在计算机网络安全领域,针对计算机主机及网络的脆弱性量化评估是目前的研究热点。本文提出了一种网络脆弱性量化评估方法,并在该评估方法的基础上开发出了相应的评估系统。系统通过对主机漏洞存在可能性以及漏洞利用可能性进行量化评估,得到目标主机的脆弱性度量值。在此基础上,结合网络拓扑结构,利用优化的最短路径算法,分析网络中存在的危险路径和关键结点,从而可以有针对性地进行网络脆弱性修补,增强网络的总体安全性能。

关键词 量化评估,脆弱性,入边等权有向图,中心性

The Research of Quantitative Vulnerability Assessment of Computer Host and Network

XIA Yang LU Yu-Liang

(Teaching and Research Office of Network Engineering of Electronic Engineering Institute, Hefei 230037)

Abstract In the field of network security, the research of quantitative vulnerability assessment of computer host and network is becoming highlighting. This paper brings forward a method of network-vulnerability quantitative assessment, and a corresponding system, based on this method, is established closely after that. In order to obtain the vulnerability measurement of target host, the possibility of the existence and the exploitability of vulnerability are evaluated quantitatively in the system. On the basis of that, by combining with network topology, the dangerous path and critical node of network are analyzed in terms of the refined shortest-path algorithms. Thereby, the administrator can patch the vulnerability with a definite purpose, and which undoubtedly would enhance the general security performance of the network.

Keywords Quantitative assessment, Vulnerability, Ingoing edge equi-weight directed graph, Centrality

1 引言

网络的飞速发展已把我们推入了一个日新月异的信息世界。当网络提供给我们无穷无尽的发展机遇时,也给我们带来了最大的挑战:网络安全性。目前,网络安全性对政府、军事单位、商界以及民用基础设施等都构成了威胁。网络安全性的根源在于网络脆弱性(vulnerability),即网络协议、网络软件、网络服务、主机操作系统及各种主机应用软件在设计及实现上存在种种安全隐患和安全缺陷。因此,网络脆弱性评估已成为新兴的网络安全研究领域,特别对网络脆弱性的量化评估是众多学者的研究热点。由于网络脆弱性评估没有统一的量化标准,脆弱性影响因素众多,网络服务的动态性、网络联接的复杂性以及网络主机之间的依赖性也决定了网络脆弱性量化评估是一项非常复杂的工作。

目前,许多科研人员都从不同的角度致力于脆弱性评估的研究上,也取得了不少可喜的研究成果,下面列举一些该领域的研究方法:

(1)基于 Kolmogrov 复杂度的脆弱性分析[1]。利用 Kolmogrov 复杂度作为脆弱性度量来检测主机系统的异常行为。当系统中 Kolmogrov 复杂度改变,并且这个改变并不是由于系统中已知操作所引起的,则表示可能有未授权的行为发生,以此来检测系统脆弱性。

(2)网络不安全路径评估[2](Network Insecurity Path Assessment)。将网络中每个主机作为图的一个结点,结点间的有向弧代表主机之间的连接,每条弧上的权值代表攻击者从一个主机入侵到另一个主机的成功可能性。利用相关的数学算法对图中的不安全路径进行评估来分析网络的脆弱性。

(3)基于网络权限图(privilege graph)的脆弱性分析[3,4]。图中每一个结点代表一个或一组用户的权限,结点 x 到结点 y 的有向边权值代表拥有 x 权限的用户转变为拥有 y 权限用户的可能性。通过对网络主机信息的信任度融合进行脆弱性关联分析,从入侵者的角度出发,将网络权限图转换为网络入侵过程状态图,从而对网络脆弱性进行量化评估。

(4)基于电路理论[5](circuit theory)的脆弱性分析。以电流作为攻击者的信息流,电阻作为网络脆弱性的存在状况,将目标网络的拓扑结构图转变成等价的电路分析图,对目标网络的脆弱性进行评估和分析。

(5)基于贝叶斯网络脆弱性评估[6]。贝叶斯网络是一个有向无循环图,图中每个结点代表一个主机状态,结点 x 到结点 y 的有向边权值代表由 x 状态到 y 状态的可能性,表示为:$P(y|x)$。通过求解贝叶斯网络中的条件概率分别(Conditional Probability Distributions),从而求解网络中的可能攻击路径及其攻击成功概率。

(6)基于攻击图的网络脆弱性分析[7]。将网络拓扑和攻击模板库结合起来形成网络攻击图,依据网络结点状况和攻击模板中的条件进行匹配,逐步形成攻击图,在攻击图的基础上,分析图中的威胁路径和关键结点,以此对网络脆弱性进行评估。

(7)其他方法。基于攻击树[8](Attack Tree)、模型检

夏 阳 讲师,博士;陆余良 教授,硕士。

测[9](Model Checking)、软件测试[10](Software Testing)、利用关系图[11](Exploit Dependency Graphs)等若干方法的网络脆弱性分析。

上述的各种脆弱性分析方法分别用不同的技术从不同的侧面对网络脆弱性进行分析,每种方法都有其可取之处,但也都存在着不足之处。本文提出一种网络脆弱性评估方法,首先对目标网络中各主机结点进行评估,得到主机的安全量化值,然后结合网络拓扑信息,对网络脆弱性进行量化评估,指出网络中存在的危险路径和关键结点。该评估方法的优越性在于:

(1)把网络拓扑结构和主机信息结合在一起考虑。目前经常使用的安全扫描器并不考虑网络拓扑结构,它们仅对单个主机进行脆弱性检测,并将检测结果以列表的方式提供给检测者。如 SATAN[12](Security Administrator Tool for Analyzing Networks)、NESSUS[13]、NMAP[14] 等。文[15]的评估方法也没有考虑网络拓扑结构,它把目标网络看作一个整体,将各影响因素施加在这个整体上,建立层次评价结构,进行 Internet 安全性评估。该方法不适合对网络进行安全评估,因为目标网络不可能作为一个整体来看待,它是由各个网络结点和网络联接路径组成的,脱离具体的评估实体而虚构一个独立的网络并对其进行评估意义不大;

(2)当网络规模较大时,评估系统仍可以正常工作,不会出现状态空间指数增长问题。对于一些研究方法,如:文[3,7,9]等,当网络主机数量增多时存在可度量性问题(scalability),其计算复杂程度以指数级增长。

依据此评估方法,我们开发了"计算机网络脆弱性量化评估系统"(以下简称"评估系统"),目前正在测试阶段。本文以下章节安排如下:第 2 节详细介绍了主机脆弱性量化评估技术;第 3 节对网络脆弱性量化评估技术进行了分析;最后对目前工作进行了总结。

2 主机脆弱性量化评估

主机脆弱性量化评估分为 3 个阶段:①评估主机上各漏洞的存在可能性;②评估主机上各存在漏洞被利用的可能性;③评估主机总体安全度量值。首先评估主机上各漏洞的存在可能性。

影响主机安全性能的信息主要有:目标主机的操作系统、系统上运行的进程和服务、开放的端口、用户及组相关信息,甚至还包括主机的日志及驱动加载等信息。为了更加准确地评估主机脆弱性,我们可以从不同渠道获取主机信息,例如由系统管理员提供,也可以通过不同的安全扫描器获取。

通过各渠道收集到的主机信息存储在评估系统的数据库中,多源信息之间可能会有不同程度的重叠部分,甚至有些信息之间还存在冲突,因此必须对这些信息进行综合分析及信任度融合处理。评估系统中利用 Dempster-Shafer 证据理论[16]进行多源信息融合。关于 D-S 证据理论的基础知识本文不再介绍。

获取的主机信息中,主机开放端口、运行服务以及平台配置等信息比较重要,利用这些信息可以进一步获取主机的漏洞信息。假设有 r 个信息源提供针对某主机的信息,r 个信息源记为$\{S_1,S_2,\cdots,S_r\}$,每个信息源的信息通过知识数据库进行信息融合,得到目标主机可能存在的漏洞信息,记为:S_1:$\{v_{11},v_{12},\cdots,v_{1i}\}$,$S_2$:$\{v_{21},v_{22},\cdots,v_{2j}\}$,$\cdots$,$S_r$:$\{v_{r1},v_{r2},\cdots,v_{rk}\}$,其中 v 代表目标主机可能存在的漏洞。我们把所有信息源的信息经过信息融合后得到的漏洞信息集合称为辨别框架 Θ,记做 $\Theta=\{V_1,V_2,\cdots,V_n\}$。评估者对 r 个信息源$\{S_1,S_2,\cdots,S_r\}$具有一定的知识,各信息源的信任度分别为 $w_1,w_2,\cdots,w_r$,其中 w_i/w_j 表示信息源 i 相对信息源 j 的权威程度。

评估系统知识数据库中定义了各种信息与漏洞之间的关联关系,也称为融合规则,融合规则的数据结构定义为一个多维向量集合 $V=$(条件$_1$:结论$_1$,条件$_2$:结论$_2$,$\cdots$,条件$_n$:结论$_n$),条件 $i=(p_i,q_i)$,其中 p_i 代表条件中一些关键属性的名字,q_i 表示相应属性的值,结论 i 代表对应与条件 i 的知识结果,该知识结果可以是针对条件 i 的漏洞信息、攻击程序甚至也可以是相应的策略等。表 1 是融合规则简单示例,左栏代表多个信息源获取的主机信息,右栏是对应该信息可能存在的漏洞列表。

表 1 融合规则表

条件信息	结论信息
OS=Windows2000;Port=21	$V_1;V_2;V_3$
OS=AIX5.1;Service=FTP2.5.0	$V_4;V_5$
OS=RadHat7.0;Port=21	$V_5;V_6$
OS=Windows2000;Port=137	$V_7;V_8;V_9$
OS=Windows2000;Por=139	$V_8;V_9;V_{10}$
Service=Apache1.3.32	$V_9;V_{10};V_{11}$
……	……

以下以一实例分析主机信息融合过程。

第一步:首先为该模型选取 3 个信息源,并根据对这 3 个信息源所掌握的知识依次为其分配信任度参数,假设为:$w_1=0.8$,$w_2=0.6$,$w_3=0.9$;

第二步:3 个信息源针对主机的信息获取结果如表 2 所示。

表 2 主机信息表

信息源	信息结果	信息源信任度
信息源 1	OS=W;port=137	80%
信息源 2	OS=W;port=139	60%
信息源 3	Service=Apache/1.3.32	90%

第三步:根据 Dempster 组合规则,假设先对信息源 1 和 2 的进行规则合成,得出结论如表 3 所示。

表 3 Dempster 组合规则合成结果表

	$m_1(\{V_7,V_8,V_9\})=0.8$	$m_1(\Theta)=0.2$
$m_2(\{V_8,V_9,V_{10}\})=0.6$	$m'(\{V_8,V_9\})=0.48$	$m'(\{V_8,V_9,V_{10}\})=0.12$
$m_2(\Theta)=0.4$	$m'(\{V_7,V_8,V_9\})=0.32$	$m'(\Theta)=0.08$

然后将表 3 中的结果与第 3 个信息源信息进行 Dempster 规则合成,结果如表 4。

表 4 Dempster 组合规则合成结果表

	$m'(\{V_8,V_9\})=0.48$	$m'(\{V_8,V_9,V_{10}\})=0.12$	$m'(\{V_7,V_8,V_9\})=0.32$	$m'(\Theta)=0.08$
$m_3(\{V_9,V_{10},V_{11}\})=0.9$	$m''(\{V_9\})=0.432$	$m''(\{V_9,V_{10}\})=0.108$	$m''(\{V_9\})=0.288$	$m''(\{V_9,V_{10},V_{11}\})=0.072$
$m_3(\Theta)=0.1$	$m''(\{V_8,V_9\})=0.048$	$m''(\{V_8,V_9,V_{10}\})=0.012$	$m''(\{V_7,V_8,V_9\})=0.032$	$m''(\Theta)=0.008$

第四步：根据第3步得到的基本信任指派函数 $m(A)$，$A \subset \Theta$，计算信任函数 $Bel(A)=\sum_{B\subseteq A} m(B)$，函数 $Bel(A,B)$ 代表 A 或 B 的信任程度：

$$Bel(V_9)=\sum_{B\subseteq V_9} m(B)=0.432+0.288=0.72$$

$$Bel(V_9,V_{10})=\sum_{B\subseteq(V_9,V_{10})} m(B)=0.432+0.288+0.108=0.828$$

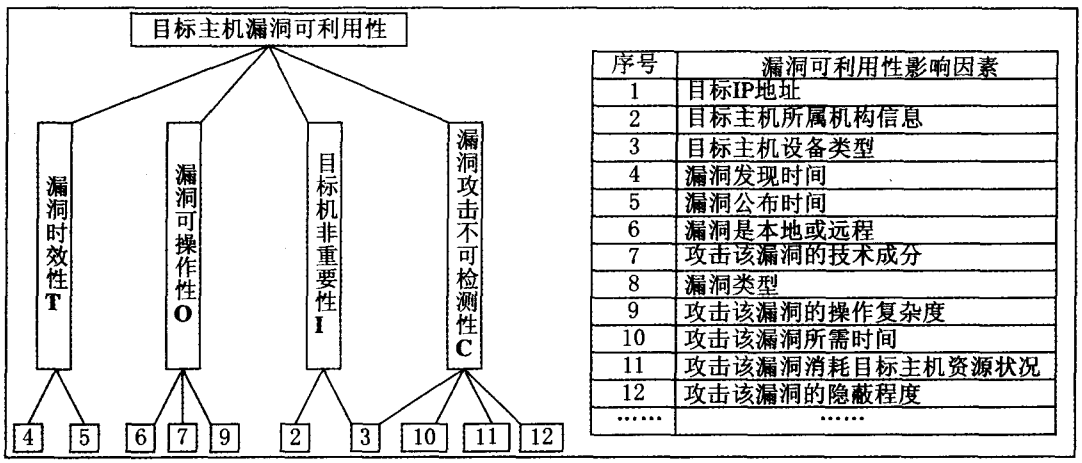
$$Bel(V_8,V_9)=\sum_{B\subseteq(V_8,V_9)} m(B)=0.432+0.288+0.048=0.768$$

根据公式，$1-(1-P(V_9))(1-P(V_{10}))=P(V_9,V_{10})$ 以及 $P(V_9)=0.72$，$P(V_9,V_{10})=0.828$，可以计算得出 $P(V_{10})=0.386$；同理 $P(V_8)=0.171$。因此，该主机上各漏洞存在可能性的大致比例为：$P(V_9):P(V_{10}):P(V_8)=0.768:0.386:0.171\approx 4:2:1$。

以上已经评估了主机上各漏洞的存在可能性，下面我们评估各漏洞被利用的可能性。

漏洞可利用性的影响因素很多，如何正确地分配众多因素的影响权重值，合理地处理各影响因素之间的相互加强及削弱关系，将是目标主机漏洞可利用性量化融合的主要问题。该评估系统采用层次分析法(AHP)评估目标主机漏洞可利用性，层次分析法的原理及相关概念可以参考文[17]。

图1从总体上描述了AHP算法评估目标主机漏洞可利用性的层次结构。



序号	漏洞可利用性影响因素
1	目标IP地址
2	目标主机所属机构信息
3	目标主机设备类型
4	漏洞发现时间
5	漏洞公布时间
6	漏洞是本地或远程
7	攻击该漏洞的技术成分
8	漏洞类型
9	攻击该漏洞的操作复杂度
10	攻击该漏洞所需时间
11	攻击该漏洞消耗目标主机资源状况
12	攻击该漏洞的隐蔽程度
……	……

图1 AHP算法评估目标主机漏洞可利用性层次结构图

主机各漏洞可利用性影响因素值可以通过转换函数转换为量化指标值，如表5所示。

表5 影响因素转换表

量化指标	影响因素	转换函数
漏洞时效性 T	f_4，f_5	$T=Ft(f_4,f_5)$
漏洞可操作性 O	f_6，f_7，f_9	$O=Fo(f_6,f_7,f_9)$
目标机非重要性 I	f_2，f_3	$I=F_i(f_2,f_3)$
漏洞攻击不可检测性 C	f_3，f_{10}，f_{11}，f_{12}	$C=F_c(f_3,f_{10},f_{11},f_{12})$
……	……	……

下面以一具体实例描述AHP算法评估主机漏洞可利用性的过程(已知主机上存在的漏洞)。

第一步：将获取的主机各漏洞可利用性影响因素值通过量化指标转换函数转换为量化指标值，得到的漏洞与量化指标值对应关系如表6所示。

表6 漏洞量化指标表

漏洞	量化指标值
V_1	T_1，O_1，I_1，C_1
V_2	T_2，O_2，I_2，C_2
V_3	T_3，O_3，I_3，C_3

第二步：利用第一步得到的各漏洞与量化指标的对应关系，把各漏洞针对每个量化指标的取值通过权重分配函数 $weigh()$ 进行权重分析，即：$x_{ij}=weigh(x_i,x_j)$；$x\in\{t,o,i,c\}$，$i,j\in\{1,2,3\}$。x_{ij} 代表针对量化指标 x，漏洞 V_i 相对于漏洞 V_j 的重要程度。从而得到各个漏洞相对于每个量化指标的判断矩阵如图2所示。

T	V_1	V_2	V_3	W_t
V_1	1	t_{12}	t_{13}	w_{t1}
V_2	$1/t_{12}$	1	t_{23}	w_{t2}
V_3	$1/t_{13}$	$1/t_{23}$	1	w_{t3}

指标T判断矩阵

I	V_1	V_2	V_3	W_i
V_1	1	i_{12}	i_{13}	w_{i1}
V_2	$1/i_{12}$	1	i_{23}	w_{i2}
V_3	$1/i_{13}$	$1/i_{23}$	1	w_{i3}

指标I判断矩阵

O	V_1	V_2	V_3	W_o
V_1	1	o_{12}	o_{13}	w_{o1}
V_2	$1/o_{12}$	1	o_{23}	w_{o2}
V_3	$1/o_{13}$	$1/o_{23}$	1	w_{o3}

指标O判断矩阵

C	V_1	V_2	V_3	W_c
V_1	1	c_{12}	c_{13}	w_{c1}
V_2	$1/c_1$ 2	1	c_{23}	w_{c2}
V_3	$1/c_{13}$	$1/c_{23}$	1	w_{c3}

指标C判断矩阵

图2 量化指标判断矩阵

目标层	T	O	I	C	W
T	1	b_{to}	b_{ti}	b_{tc}	w_t
O	$1/b_{to}$	1	b_{oi}	b_{oc}	w_o
I	$1/b_{ti}$	$1/b_{oi}$	1	b_{ic}	w_i
C	$1/b_{tc}$	$1/b_{oc}$	$1/b_{ic}$	1	w_c

图3 目标层判断矩阵

图2中 $W_k(k\in\{t,o,i,c\})$ 代表针对量化指标 k 判断矩阵

的特征向量。

第三步：计算各个量化指标相对于目标层的判断矩阵，如图3。

图3中$b_{xy}(x,y\in\{t,o,i,c\})$代表量化指标$x$相对于量化指标$y$的重要程度。$W$代表目标层判断矩阵的特征向量。

第四步：通过第二步和第三步已经得到了各个漏洞相对于各量化指标的判断矩阵以及目标层判断矩阵，合成判断矩阵，计算各漏洞可利用性量化值{R1,R2,R3}：

$$\begin{pmatrix} w_{t1} & w_{o1} & w_{i1} & w_{c1} \\ w_{t2} & w_{o2} & w_{i2} & w_{c2} \\ w_{t3} & w_{o3} & w_{i3} & w_{c3} \end{pmatrix} \begin{pmatrix} w_t \\ w_o \\ w_i \\ w_c \end{pmatrix} =$$

$$\begin{pmatrix} w_{t1}w_t + w_{o1}w_o + w_{i1}w_i + w_{c1}w_c \\ w_{t2}w_t + w_{o2}w_o + w_{i2}w_i + w_{c2}w_c \\ w_{t3}w_t + w_{o3}w_o + w_{i3}w_i + w_{c3}w_c \end{pmatrix} = \begin{pmatrix} R_1 \\ R_2 \\ R_3 \end{pmatrix}$$

在对主机漏洞存在性及漏洞可利用性进行量化评估的基础上，可以进一步得到主机总体脆弱性度量。我们把漏洞存在可能性与漏洞被利用可能性乘积的最大值作为主机的脆弱性量化评估值。

3 网络脆弱性量化评估

通过对目标网络中各主机进行脆弱性量化评估，我们可以得到各主机的安全度量值。在此基础上可以进一步对目标网络进行脆弱性分析，该分析主要从两个方面进行：(1)目标网络关键路径分析；(2)目标网络关键结点分析。

首先，我们分析网络的关键路径。图4是一个简单网络拓扑示意图，图中有向边Ⓧ$\xrightarrow{W}$Ⓨ代表存在从结点X到达结点Y的信任关系，w代表结点Y的脆弱性度量。w值越高，主机的脆弱性越高，主机的安全性越低。由于主机脆弱性与安全性成反比，故可以将网络结点脆弱性度量图4转换为网络结点安全性度量(图5)，图5中w代表有向边所指结点的安全性度量(注：w也可理解为攻击者从结点X攻击结点Y的代价)。

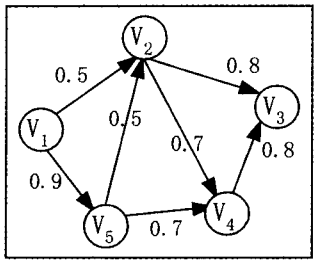


图4 网络脆弱性度量示意图

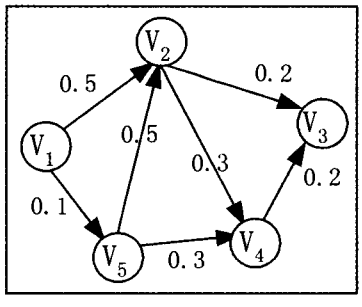


图5 网络安全性度量示意图

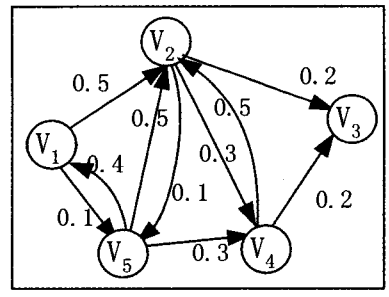


图6 入边等权有向多重图

以图5为例，假设结点V1为攻击者起始结点，结点V3为目标结点，我们可以利用最短路径算法求解从V1到V3的最短路径为：V1→V5→V4→V3。同理，我们可以求解任何结点之间的最短路径，该最短路径代表从源结点到目标结点的最佳攻击路径，也是网络脆弱性关键路径。网络管理员可以分析网络最短路径上各主机的脆弱性状况，找出关键结点，修补其脆弱性，从而增强网络整体安全性能。

网络路径脆弱性分析是网络安全性能评估的重要组成部分，而网络路径分析必须解决网络拓扑图的最优化道路问题。因此，图的最短路径问题在网络安全评估领域具有重要的研究价值。

在图5所示网络中，每个主机对于其信任主机的脆弱性度量是相同的。例如结点V1和V5均是结点V2的信任结点，攻击者在结点V1和V5上可以以相同的概率攻击结点V2。结合该图的特征，本文提出如下定义：在无自环和重边的正权有向图G中，对任何结点V_i，所有以V_i为终点的边的权值相等，我们称该图为入边等权有向简单图。

Dijkstra提出了按路径长度递增次序产生最短路径的算法[18]，但是该算法应用在入边等权有向图中将增加不必要的计算，而且算法的执行循环次数明显增多。本文基于Dijkstra算法，提出了更为有效的计算入边等权有向图的最短路径算法。

针对入边等权有向图求解最短路径的算法具体描述如下。

第一步：初始化。

①用二维数组$weight[i,j]$表示图中结点v_i到v_j的边$\langle v_i,v_j\rangle$上的权值，若$\langle v_i,v_j\rangle$不存在，则$weight[i,j]=\infty$。

②S为已找到从v_1出发的最短路径的终点的集合，它的初始状态为空。

③从v_1出发到图上其他各结点v_i的最短路径长度的初始值为：

$length[i]=weitht[1,i]$　　其中$v_i\in V$；

④集合$path[i]$存储从结点v_1到其他各结点v_i的最短路径上的结点。其中$v_i\in V;i\neq 1$

第二步：选择v_j，使得$length[j]=Min\{length[i]\}$，其中$v_i\in\{V-S\}$，$v_j$就是当前求得的一条从$v$出发的最短路径的终点。令$S=S\cup\{j\}$。

第三步：如果$length[i]\neq\infty$，则$length[i]=length[i]$；如果$length[i]=\infty$，则$length[i]=length[j]+weight[j,i]$。同时$path[i]=path[j]+[i]$。其中$i\notin S$。

第四步：如果所有$length[i]\neq\infty$，则算法结束，否则，如果所有$length[i]=\infty$，则算法结束(注：存在不可达路径)。否则继续执行第二步，其中$i\notin S$。

通过对该算法执行可以看出，在计算从 v_1 到其他各结点的最短路径时，减少了对一些结点不必要的计算，而且算法执行的循环次数也有所减少，提高了最短路径算法的效率（注：当图中结点数目较多时效果更加明显）。

该算法也适合在多重图中计算最短路径，多重图更加体现出真实网络各结点间的信任关系。我们在图 5 的基础上增加一些有向边，使该简单图变成图 6 所示入边等权有向多重图。利用优化最短路径算法计算图 6 中从 v_2 到其他各结点的最短路径的过程描述如表 7。

表 7　入边等权有向多重图中最短路径算法执行过程表

终点	从 v_2 到其他各结点的最短路径及 length 值	
v_1	length [1] = ∞ path [1] =[]	length [1] = 0.5 path [1] =[v_2, v_5, v_1]
v_3	length [3] =0.2 path [3] =[v_2, v_3]	length [3] =0.2 path [3] =[v_2, v_3]
v_4	length [4] = 0.3 path [4] =[v_2, v_4]	length [4] = 0.3 path [4] =[v_2, v_4]
v_5	length [5] = 0.1 path [5] =[v_2, v_5]	
v_j 及 S	v_j=v_5　S={ v_5}	

可见，在多重图中，由于两两结点之间的直接相连路径增多，因此算法的执行效率将更高。

接下来，我们进一步分析目标网络中存在的关键结点。图 7 是一个简单网络拓扑图，该图中各结点符合入边等权的特性，因此，上述优化最短路径算法也适用于该图。

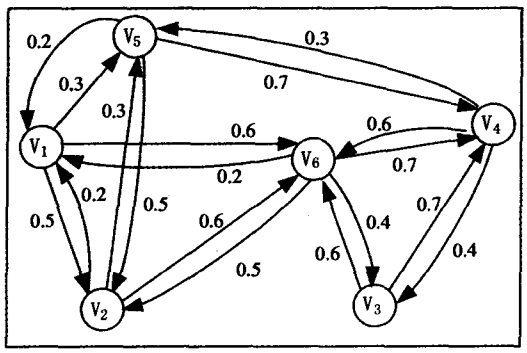


图 7　简单网络拓扑示意图

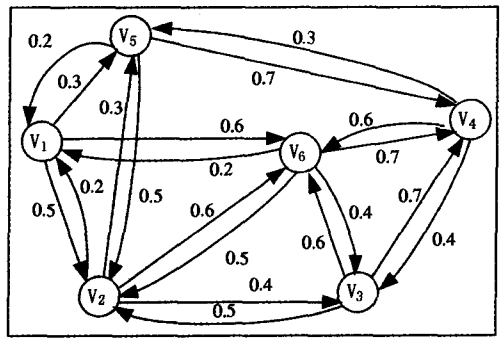


图 8　改动后的简单网络拓扑示意图

在该网络上求解关键结点的算法描述如下：

```
for (i=1; i<=max; i++) k(i)=0;
//max 是网络中的结点总数
//k(i)存储各结点在所有最短路径中出现的次数
for (i=1; i<max; i++)
{
  for (j=i+1; j<=max; j++)
  {
    利用优化最短路径算法计算结点 Vi 到 Vj 最短路径上所遍历的
    结点:Vi…Vp,Vq…Vj;
    K(i)++;…;K(p)++;
    K(q)++;…;K(j)++;
    }
}
按从大到小的顺序排序数组 K(i);// i∈[1,max]
//K(i)值越高，则结点 Vi 在网络中的关键性越强
```

对应于图 7，则有 $K(5)=K(6)=7$；$K(1)=K(4)=6$；$K(2)=K(3)=5$。因此图 7 中关键结点应该是 V_5 和 V_6，其他结点可以看作次要结点。

当我们对图 7 稍作改动，增加结点 V_2 和 V_3 之间的连接，如图 8 所示。

依据上述求解关键结点的算法，在图 8 中有 $K(2)=7$；$K(1)=K(3)=K(5)=6$；$K(4)=K(6)=5$。因此，图 8 中最关键的结点应该是 V_2。可见，网络结构稍作改动，关键结点就发生了变化。原先关键结点 V_6 现在变为次要结点，而原先次要结点 V_2 现在却成为关键结点。

当网络拓扑图中任意两个结点 V_i 和 V_j 之间存在两条以上相等的最短路径时，如图 9 中结点 V_1 和 V_4 之间存在两条相同的最短路径 $V_1 \rightarrow V_5 \rightarrow V_6 \rightarrow V_4$ 和 $V_1 \rightarrow V_2 \rightarrow V_3 \rightarrow V_4$，上述求解关键结点的算法将不知道选择哪一条最短路径。鉴于此，我们还可以通过求解网络中心性的方法找出网络的关键结点。（注：图 10 是与图 9 等同的有向图。）

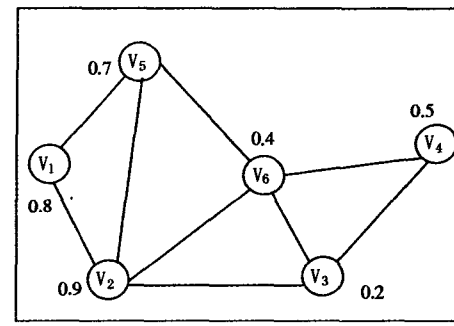


图 9　简单无向网络拓扑示意图

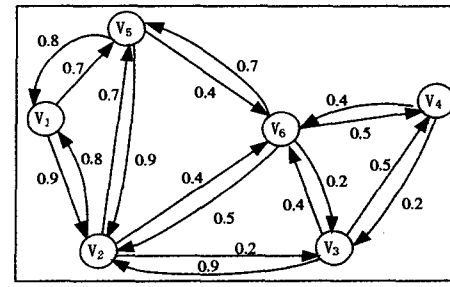


图 10　与图 9 等同的有向图

网络中心性（centrality）研究是网络安全领域的一个重要研究分支。网络中心性是网络结点的结构属性，在网络分析中常被用来检测网络结点取得资源、控制资源的可能性。目前，对网络中心性研究的方法比较多[19~21]，主要包括度中心性（degree centrality）、接近中心性（closeness centrality）、中介中心性（betweenness centrality）和特征向量中心性（eigenvector centrality）等。

度中心性为衡量一个结点控制范围大小的指标，度中心性越高者，表示其在网络中与较多的结点有所关联，其拥有的非正式权力和影响力也较多。度中心性定义为一个结点的度

数，如定义1所示，其中A是该网络的邻接矩阵。

$$c_D(i)=\sum_j A_{ij} \quad \text{（定义1）}$$

为了标准化度中心性，使其值在[0,1]之间，将度中心性除以一个结点可能连接的最大度数$n-1$，得到标准度中心性，其描述如定义2[22,23]：

$$C_D(i)=\frac{c_D(i)}{n-1} \quad \text{（定义2）}$$

一个结点与其他结点之间的直接联系越多，该结点的位置越重要。由于该结点有许多联系，因此它很少依赖别的结点，它可以访问并且调用其他结点的资源，而且它通常作为其他双方通信的中转者，具有通信的决定权。因此，度中心性是一种简单而有效的网络中心性测量方法。

度中心性仅仅考虑了一个结点和其他结点的直接联系，没有总体考虑该结点和整个网络的间接联系。比如一个结点和其他许多结点直接相连，但所有相连的结点都是孤立结点，在这种情况下，度中心性就存在其不足之处。

接近中心性衡量准则是去判断一个结点与其他结点的接近程度，与其他结点距离越短，接近中心性越高，表示其能较快速的分发和获取信息。接近中心性定义为一个给定结点到其他各结点最短路径的总和的倒数，描述如定义3：

$$c_C=(v)=\frac{1}{\sum_{x\in V} d(v,x)} \quad \text{（定义3）}$$

其中，$d(v,x)$是结点v和结点x之间的最短路径。接近中心性的标准化定义为[23,24]：

$$C_C(v)=(n-1)\cdot c_C(c) \quad \text{（定义4）}$$

接近中心性仅限于使用在连通图中，若结点i和j之间无连通路径，则$d(i,j)=\infty$，接近中心性的定义将无意义。

中介中心性是衡量一个结点作为其他任何两个结点通信中介的重要程度。具体的说，如果g_{ij}代表从结点i到结点j的最短路径数量，$g_{ij}(v)$代表从结点i到结点j的最短路径中通过结点v的路径数量。结点v的中介中心性定义如下：

$$c_B(v)=\sum_{i\neq v\neq j\in V \atop i<j} \frac{g_{ij}(v)}{g_{ij}} \quad \text{（定义5）}$$

中介中心性的标准化定义为[23,25]：

$$C_B(v)=\frac{c_B(v)}{\frac{1}{2}(n-1)(n-2)} \quad \text{（定义6）}$$

一个网络中两两结点之间的通信，必须通过另一个结点的中间介绍，对于中介中心性较高的结点，其获得信息流的机会较多，同时也占据了操纵信息流通的关键性位置。

可以在上述中介中心性定义的基础上对其进行扩展，当不知道整个网络的拓扑结构时，任何两个结点间的信息传输无法基于最短路径进行通信，这时的传输路径是随机选择的，基于随机路径的中介中心性方法描述详见文[26]。

以网络中介中心性为例，针对图9求解出各结点的中介中心性结果分别为：$C_B(v_1)=0$；$C_B(v_2)=3/2$；$C_B(v_3)=3/2$；$C_B(v_4)=0$；$C_B(v_5)=3/2$；$C_B(v_6)=5/2$。结点v_6的中介中心性值最大，其在图中占据了操纵信息流通的关键性位置。因此，结点v_6是该网络中的关键结点。

总结 本文提出了一种网络脆弱性量化评估方法，并在该评估方法的基础上开发了相应的评估系统。评估系统对目标网络进行脆弱性量化评估的步骤如下：

步骤1：通过多源信息渠道获取目标网络及各主机的安全信息；

步骤2：利用证据理论评估网络中各主机的漏洞存在可能性；

步骤3：利用层次分析法评估网络中各主机的漏洞被利用可能性；

步骤4：计算出网络中各主机的总体脆弱性量化值；

步骤5：结合具体的网络拓扑结构，利用改进的最短路径算法分析网络脆弱性路径；

步骤6：利用网络中心性算法定位网络关键结点。

该评估系统的研制得到了充足的研究经费支持，目前在实验室环境下已基本完成编码工作，能够初步进行网络脆弱性信息的收集、分析及量化评估，取得了预期的结果。通过使用该系统，增强了网络管理员对网络主机脆弱性的分析能力和对网络中危险路径、关键结点的识别能力，从而可以有针对性的进行脆弱性修补，增强网络的总体安全性能。

在进一步的研究工作中，我们将重点放在网络脆弱性的量化评估研究上，争取从其他不同的角度结合相关的数学模型对网络脆弱性进行度量。另外，各主机脆弱性之间的关联关系也是我们下一步研究的重点内容。

参考文献

1 Evans S, Bush S F, Hershey J. Information assurance through kolmogorov complexity. DISCEX-II California, 2001
2 Bush S F. Network Vulnerability Analysis Tool Precis. General Electric Corporate Research and Development Center, 1999,3
3 Dacier M, Deswartes Y, Kaaniche M. Quantitive assessment of operational security models:[LAAS Research Report]. 96493, May 1996
4 Ortalo R, Deswarte Y, Ka^aniche M. Experimenting with quantitative evaluation tools for monitoring operational security. IEEE Transactions on Software Engineering, 1999,25(5):633～650
5 Shake T, Hazzard B, Marquis D. Assessing Network Infrastructure Vulnerabilities to Physical Layer Attacks. In: MIT Lincoln Laboratory, 22nd National Information Systems Security Conference, 1999,10
6 Liu Y, Hong M. Network vulnerability assessment using Bayesian networks. Stevens Institute of Technology, 2004
7 Phillips C, Swiler L. A Graph-based System for Network-Vulnerability Analysis. In: Proceedings of the New Security Paradigms Workshop, Charlottesville, VA, 1998
8 Schneier B. Attack Trees. Doctor Dobbs Journal, 1999(12)
9 Names left out for anonymous refereeing, Model-based Vulnerability Analysis of Computer Systems. In: 2nd Int'l Workshop on Verification, 1998
10 Krsul I, Spafford E, Tripunitara M. Computer Vulnerability Analysis. COAST Laboratory, 1998,3
11 Noel S, Jajodia S, O'Berry B, et al. Efficient Minimum-cost Network Hardening via Exploit Dependency Graphs. In: 19th Annual Computer Security Applications Conference, Las Vegas, Nevada, December 2003
12 SATAN (Security Administrator Tool for Analyzing Networks) tool. SATAN's creators, Mr. Dan Farmer and Mr. Wietse Venema, made SATAN widely available over the Internet without cost starting April 1995
13 http://www.nessus.org/
14 http://www.insecure.org/nmap/
15 刘怀亮. Fuzzy-AHP 法评价 Internet 安全. 计算机工程, 2002(1)
16 Yager R R. On the Dempster-shafter framework and new combination rules. Information System, 1989(4):93～137
17 Saaty T L. How to make a decision: the analytic hierarchy process. European Journal of Operational Research, 1990,1(48):9～26
18 严蔚敏,吴伟民. 数据结构. 清华大学出版社, 1995
19 Latora V, Marchiori M. A measure of centrality based on the network efficiency. Elsevier Science,2004
20 Borgatti S P. Centrality and Network Flow. In: International Social Networks Conference, 2002
21 Stang T, Pourbayat F, Burgess M, et al. A Network Security Analysis Tool. LISA,2003. 149～158
22 Nieminen J. On centrality in a graph. Scandinavian Journal of Psychology,1974,15:322～336
23 Freeman L C. Centrality in social networks: I. Conceptual clarification. Social Networks,1979, 1:215～239
24 Wasserman S, Faust K. Social Network Analysis. Cambridge: Cambridge University Press, 1994
25 Anthonisse J M. The rush in a graph. Amsterdam: University of Amsterdam Mathematical Centre, 1971
26 Newman M E J. A measure of betweenness centrality based on random walks. 2003