基于攻击图的网络安全概率计算方法

叶 云 徐锡山 贾 焰 齐治昌
(国防科学技术大学计算机学院 长沙 410073)

摘 要 针对基于攻击图的概率计算中循环路径导致的攻击图难以理解和概率重复计算问题以及渗透之间的相关性导致的概率错误计算问题,通过将攻击图与通用安全脆弱点评估系统结合,删除攻击图中的不可达路径,简化了攻击图,提出了适用于大规模网络的最大可达概率的概念和计算方法,解决了概率重复计算问题,有效避免了相关性导致的概率错误计算问题,并通过真实实验和模拟实验验证了所提方法的合理性和有效性. 与相关的研究成果相比,最大可达概率计算方法可以适应于更复杂的攻击图,具有很好的扩展性.

关键词 攻击图;循环路径;相关性;累计成功概率;最大可达概率
中图法分类号 TP311 **DOI 号**: 10.3724/SP.J.1016.2010.01987

An Attack Graph-Based Probabilistic Computing Approach of Network Security

YE Yun XU Xi-Shan JIA Yan QI Zhi-Chang
(*School of Computer Science, National University of Defense Technology, Changsha* 410073)

Abstract To protect critical resources in networked environments, it is important to quantify the likelihood of potential multi-step attacks in attack graphs. Aimed at the problems that the difficulty to understand the attack graphs and the probabilistic re-computing caused by cyclic paths in attack graphs and probabilistic incorrect computing caused by shared dependencies in exploits, a methodology for security risk analysis that is based on the model of attack graphs and the Common Vulnerability Scoring System (CVSS) is presented, attack graph is simplified by removing unreachable paths, and the problem of probabilistic re-computing is solved and the problem of probabilistic incorrect computing is avoided successfully by proposing the concept and computing approach of maximum reachable probability which can be adapted to a large-scale network, reasonableness and effectiveness of proposed method is verified in the real experiment and simulation. Compared with the related research, maximum reachable probability computing approach can be adapted to a more complex attack graph, and have good scalability.

Keywords attack graphs; cyclic paths; shared dependencies; cumulative successful probability; maximum reachable probability

1 引 言

近年来,网络攻击事件的数量不断增长,互联网络的安全性受到了越来越多人的关注. 之所以会有如此众多的攻击行为,最主要、最根本的原因还是计算机系统存在可以被渗透的脆弱点,或者称作安全漏洞[1]. 目前存在许多成熟的脆弱点扫描工具如

收稿日期:2010-08-22. 本课题得到国家"八六三"高技术研究发展计划项目基金(2007AA010502,2009AA01Z436)资助.**叶 云**,男,1981 年生,博士研究生,主要研究方向为网络安全. E-mail: yeyun1234@tom.com. **徐锡山**,男,1963 年生,教授,主要研究领域为网络安全、软件工程和软件可靠性技术. **贾 焰**,女,1961 年生,教授,博士生导师,主要研究领域为网络信息安全、数据库和数据挖掘. **齐治昌**,男,1942 年生,教授,博士生导师,主要研究领域为软件工程、软件体系结构.

Nessus、X-Force 等，可以自动发现目标网络中已知的脆弱点.然而，考虑到软件补丁发布和硬件升级在时间上的滞后，开发软件补丁和升级硬件需要一定的费用以及网络管理员为了保证网络的可用性而必须开放某些存在脆弱点的服务等因素，很多脆弱点被发现后在网络中仍然存在.这样，我们需要对目标网络进行安全态势评估，以便帮助管理员理解网络以及为他们提供更多更有效的安全建议，而安全态势中最重要的指标就是渗透的可能性，也就是渗透的概率.

目前，有不少组织致力于标准化安全脆弱点的发布和属性定义的工作，其中最重要的成果是通用安全脆弱点评估系统 CVSS(Common Vulnerability Scoring System)，它关注于单个脆弱点的属性量化值，但不能分析脆弱点在目标网络中的严重程度.基于攻击图的分析方法为整个目标网络和攻击者建立模型，利用模型分析工具对目标网络系统的脆弱点进行综合分析，发现未知的系统脆弱点以及脆弱点之间的关系，展示了攻击者利用目标网络内不同脆弱点逐步实施攻击各个击破的所有可能的攻击路径.对目标网络进行科学的安全评估，需要识别网络中攻击者利用各个脆弱点之间相互关系产生的潜在威胁，攻击图则是解决该问题的良好途径之一.攻击图是一个有向图，它与 CVSS 系统相结合时，就形成了一个贝叶斯网络，CVSS 系统提供了在目标网络中攻击者成功渗透的概率值.

在基于攻击图的网络安全概率计算中，存在着两个挑战性问题[2]，第 1 个挑战是攻击图中存在的循环路径所导致的攻击图难以理解和概率重复计算问题.攻击图中存在很多不同类型的循环路径，这些循环路径会对基于攻击图的网络安全概率计算带来不同复杂度的影响.图 1 是一个攻击图的示例，其中纯文字节点代表条件节点，椭圆形节点代表渗透节点，也就是攻击的步骤.从图 1 中我们可以看出该攻击图含有两条循环路径，第 1 条是 $c_2 \rightarrow e_3 \rightarrow c_3 \rightarrow e_2 \rightarrow c_2$，第 2 条是 $e_{10} \rightarrow c_{11} \rightarrow e_{11} \rightarrow c_{12} \rightarrow e_{10}$.我们计算第 1 条循环路径中各个节点的概率时，会导致各个节点上的概率重复计算，如对于节点 c_2，我们首先计算出节点 c_2 的概率后，再计算节点 e_3 的概率、节点 c_3 的概率、节点 e_2 的概率，然后又重新计算节点 c_2 的概率，这就造成了节点 c_2 的重复的计算，导致不合理的概率结果.对于第 2 条循环路径，我们可以发现渗透节点 e_{10} 有 3 个前提条件 c_7，c_{10}，c_{12}，而条件节点 c_{12} 是永远不会满足的(因为 c_{12} 的满足依赖于节点 e_{10} 的成功渗透)，故该循环路径在实际的网络中是不会存在的，即该循环路径中的任何节点上的概率都为零，我们可以将此类循环路径从攻击图中删除，以便于管理员更好地查看和分析攻击图.

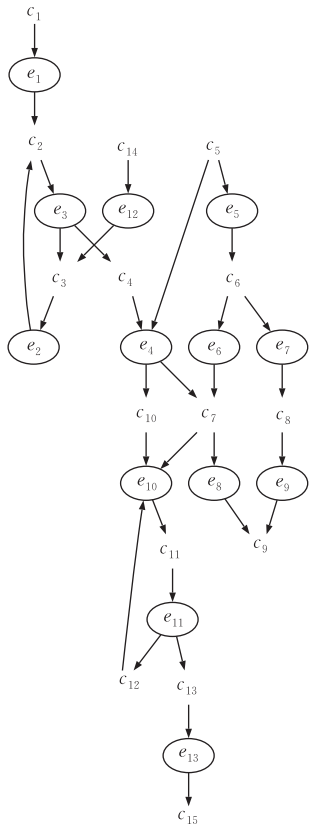

图 1 攻击图示例

第 2 个挑战是渗透之间的相关性所导致的概率错误计算问题.如图 1 中的节点 c_9 的概率可以通过渗透节点 e_8 和 e_9 的概率计算得到，如果渗透节点 e_8 和 e_9 之间是独立的，那么节点 c_9 的概率为 $P(e_8)+P(e_9)-P(e_8)\cdot P(e_9)$，其中 $P(e_8)$ 和 $P(e_9)$ 分别为渗透节点 e_8 和 e_9 的概率；然而图 1 中 e_8 与 c_6 具有一定的相关性，e_9 完全与 c_6 相关，所以 e_8 和 e_9 并不是独立的.

本文首先通过删除攻击图中的不可达路径，减少攻击图中的循环路径，简化了攻击图，增强了管理员对攻击图的理解，然后提出最大可达概率的概念和方法来解决攻击图中存在的循环路径所导致的概率重复计算问题，回避渗透之间的相关性所导致的概率错误计算问题.本文第 2 节介绍相关的研究；第 3 节给出计算攻击图中各个节点的概率所需的基础数据的获取方法、攻击图的简化算法和基于攻击图的最大可达概率计算算法；第 4 节运用真实实验和模拟实验验证了本文所提概念、方法的合理性和

有效性；最后总结本文的工作并对未来的研究进行展望.

2　相关研究

网络安全风险具有传播性，即网络安全风险会在目标网络中通过其多个脆弱点在相关服务和主机间传递. Wang 等[3-4]基于属性攻击图考虑了攻击的难度、重配置网络的代价以及网络中关键信息资产的价值等，提出了网络安全度量方法；Feng 等[5]将可靠性的思想引入攻击图来分析网络的脆弱点；石进等[6]提出了一种基于攻击图的入侵响应模型，考虑了系统、攻击者的收益等因素；Mehta 和 Sawilla 等[7-8]考虑攻击图中各个节点由于在攻击路径中所处位置不同而具有不同的重要性，如某些原子攻击是多条攻击路径中的关键点等，基于 Google Rank（网页级别）的思想来计算各个节点的重要度.

由于攻击图中可能存在循环路径，在进行网络安全概率计算时，会导致循环节点概率值的重复计算，产生与实际情况不相符的错误概率值. 大部分文献[3-8]都未考虑这种情况. Ou 等[9]首先提出了攻击图复杂的原因之一是攻击图存在循环路径问题，并研究发现图中循环路径不能简单地通过删除某些原子攻击解决，否则会丢失一些重要的无圈攻击路径，但没有提出寻找所有无圈攻击路径的方法；Wang 等[10]讨论了 3 种不同类型的循环路径对风险计算的影响，通过删除循环路径中每个节点的后继可达节点和边的方法消除循环路径影响，该方法对循环路径中的节点的处理十分复杂，同时 Wang 没有给出计算各节点概率的详细算法，也没有考虑渗透之间的相关性所导致的概率错误计算问题；陈锋[11]提出了有效攻击路径分析技术，采取前向搜索的方式和深度优先的搜索策略寻找每个节点的有效攻击路径，通过一个存放中间节点的集合来防止产生循环路径，该算法的时间复杂度是指数级的，也不适用于大规模网络；Homer 等[12]将含有循环路径的攻击图转换为等价的不含循环路径的攻击图，使得一个节点在任意一条路径中只出现一次. 这种方法增加了很多的节点和有向边，攻击图将变得十分复杂. 他同时考虑到了渗透之间的相关性问题，根据贝叶斯网络中的分割定理，给出了解决渗透之间的相关性问题的一种方法. 这种方法没有考虑渗透节点具有多个结果的情形，同时联合概率的获得和计算也十分复杂，仅适用于小规模网络.

3　基于攻击图的概率计算

攻击图展示了攻击者从初始攻击能力出发，利用目标网络中的多个脆弱点，在实施多步骤网络组合攻击的过程中产生新的攻击能力，这些新的攻击能力即为目标网络所面临的潜在威胁，它直接反映了目标网络中安全威胁传播的途径和方式. 我们首先给出攻击图的定义.

定义 1.　攻击图 $G=(C_o\cup C_d,T,E)$ 是一个有向图，其中，C_o 表示初始条件节点集合，C_d 表示中间条件节点集合，T 表示渗透节点集合. G 满足约束：(1) $E\sqsubset((T\times C_d)\cup((C_o\cup C_d)\times T))$；(2) 对 $\forall t\in T$，令 $Pre(t)$ 是 t 的父节点集合，$Post(t)$ 是 t 的子节点集合，则父节点之间存在“与”关系，且满足 $(\wedge Pre(t))\Rightarrow(\wedge Post(t))$；(3) 对 $\forall c\in C_d$，$Pre(c)$ 是 c 的父节点集合，则父节点之间存在“或”关系，且满足 $(\vee Pre(c))\Rightarrow c$.

3.1　基础数据的获取

攻击图 $G=(C_o\cup C_d,T,E)$ 能够有效识别目标网络中所有潜在威胁，为了确定条件节点 $c_i\in C_o\cup C_d$ 在攻击图中发生的概率 $P(c_i)$ 和相关渗透节点 $e_i\in T$ 在攻击图中发生的概率 $P(e_i)$，必须首先获取基础数据，即条件节点 c_i 和渗透节点 e_i 的自身概率 $d(c_i)$ 和 $d(e_i)$，然后再计算各个节点在攻击图中的发生的概率 $P(c_i)$ 和 $P(e_i)$.

攻击者在实施多步骤网络攻击时，不仅会利用目标网络中的脆弱点实施非法网络攻击提升自身对目标网络的控制能力，有时候也会通过实施正常的网络操作行为提升其攻击能力，如远程登陆操作等. 为此，为了与针对脆弱点的渗透区分，我们将这类渗透归为 B 类，而将针对脆弱点的渗透归为 A 类.

对于攻击图中的 A 类渗透节点，自身概率取自遵循 CVSS 标准的 NVD 数据库（National Vulnerability Database）中的“AccessComplexity”属性值 E. NVD 数据库中的“AccessComplexity”字段表征攻击者渗透该脆弱点的难易程度，我们将此视为成功渗透该脆弱点的自身概率. 按照 CVSS 的推荐，$d(e_i)$ 取值如下：

$$d(e_i)=\begin{cases}0.35, & E\text{ 为“High”}\\ 0.61, & E\text{ 为“Medium”}\\ 0.71, & E\text{ 为“Low”}\\ 0.71, & E\text{ 为“Undefined”}\end{cases}.$$

对于攻击图中的 B 类渗透节点和条件节点，其

自身概率 $d(c_i)$和 $d(e_i)$都设为 1(也可根据经验设定为 0～1 之间的任意值).

3.2 攻击图的简化

攻击图包含大量的纯文字(条件节点)、椭圆(渗透节点)和有向边,可读性极差,不利于管理员对攻击图进行深入的分析,并且影响计算攻击图中成功渗透各节点的概率的效率.通过研究发现,某些攻击路径虽然在攻击图中存在,但是在实际中却是不可能发生的.如图 1 中的路径 $e_{10}\rightarrow c_{11}\rightarrow e_{11}\rightarrow c_{12}\rightarrow e_{10}$ 在实际的网络中是不可能发生的,但却存在于攻击图中,我们可以删除这些不可能到达的路径,以达到简化攻击图的目的,并提高计算各可达节点概率的效率.因为攻击图中的渗透节点前提条件全部满足时,它才会成功发生,所以我们改进了广度优先搜索策略,为每个渗透节点增加一个计数器来判断它的前提条件是否全部满足,下面我们给出一个基于广度优先搜索策略的简化攻击图算法.算法 1 的 1～3 行,先计算每一个渗透节点的父节点数目,而且将每个渗透节点标记为 false;4～5 行,将初始条件节点插入一个队列 q 中;6～12 行,重复地从队列 q 中取出条件节点,并将它的子节点(是渗透节点)计算器 *number* 加 1,如果 *number* 等于该子节点的父节点数目,那么说明该子节点的所有前提条件都已满足,可以成功发生,将该子节点的标记改为 true,再将该子节点的所有子节点插入队列 q 中;13 行,当最终队列为空时,返回不含不可达路径的攻击图.

算法 1. 简化攻击图算法.

Input: 攻击图 $G=(C_o\cup C_d,T,E)$

Output: 消除不可达路径的攻击图

Method:

1. For each $e_i\in T$ Do
2. $Count[e_i]=Pre(e_i)$的数目;
3. $R(e_i)=$false;
4. For each $c_i\in C_0$ Do
5. $Insert(q,c_i)$; //将 c_i插入队列 q
6. For each $t=Delete(q)$ //出队
7. For each $e_j\in Post(t)$ Do
8. $number[e_j]=number[e_j]+1$;
9. If $number[e_j]=Count[e_j]$ Then
10. $R(e_j)=$true;
11. For each $c_k\in Post(e_j)$ Do
12. $Insert(q,c_k)$;
13. Return 删除所有 R 为 false 的节点和与之相连的边后的攻击图.

图 1 中攻击图经过算法 1 简化后,得到图 2 中的攻击图.从图中可以看出,路径 $e_{10}\rightarrow c_{11}\rightarrow e_{11}\rightarrow c_{12}\rightarrow e_{10}$ 都已删除,并且与 e_{11} 相连的 c_{13} 及其 e_{13} 和 c_{15} 都已删除.化简后的攻击图删除了不可达路径,更加具有可读性,符合真实网络中的实际情形.化简后的攻击图含有更少的节点和有向边,在一定程度上提高了下节介绍的基于攻击图的概率计算算法的效率.

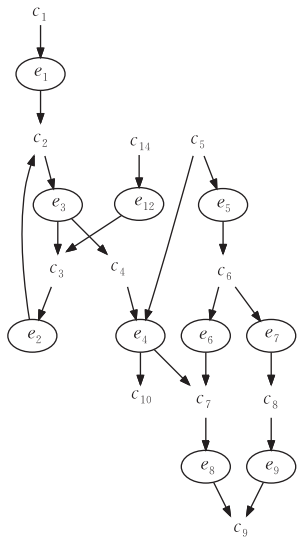

图 2 简化后的攻击图示例

3.3 最大可达概率的计算

研究中我们发现,计算机网络系统以网络化为特征,环节多,成分复杂,所以其安全问题需要进行全面的考虑,并遵循木桶原则——木桶的容量是由最短的那块木板长度决定的,即系统的安全强度取决于它最薄弱环节的安全强度.故对于攻击图中的任意条件节点 $c\in C_d$,我们将其所有父节点的概率的最大值与该条件节点的自身概率的乘积作为其累计成功概率值.下面我们给出累计成功概率函数 P 和最大可达概率的定义.

定义 2. 给定不含循环路径的攻击图 $G=(C_o\cup C_d,T,E)$,累计成功概率函数 P 定义为:对任意 $e\in T$, $P(e)=d(e)\cdot\prod_{c\in Pre(e)}P(c)$;对任意 $c\in C_o$, $P(c)=d(c)$;对任意 $c\in C_d$, $P(c)=d(c)\cdot \mathrm{Max}\{P(e)\mid e\in Pre(c)\}$.

定义 3. 对于攻击图中任意节点 t,称攻击者从初始节点出发到节点 t 的所有路径中的最佳路径所需的成功渗透概率值,即所有路径中所需的成功渗透概率的最大值称为 t 的最大可达概率.

对于定义 2 给出的累计成功概率函数 P 的意义,我们给出如下定理.

定理 1.　由累计成功概率函数 P 计算出的各节点 t 的概率为最大可达概率.

证明.　设每个节点在攻击图中的最大深度为 L，当节点的最大深度 L 为 0 时，即节点为初始条件节点时，该定理显然成立；假设节点的最大深度 L 为 k 时，该定理成立，那么对于最大深度 L 为 $k+1$ 的节点 t，若 t 为渗透节点，根据定义 1 中的约束 2 可知，t 的最大可达概率是它的所有父节点的最大可达概率之积乘以 t 的自身概率，即 $d(t)\cdot\prod_{c\in Pre(t)}P(c)$；若 t 为条件节点，根据定义 1 中的约束 3 可知，t 的最大可达概率是所有父节点 e 的最大可达概率中的最大值与其自身概率的乘积，即 $d(t)\cdot\mathrm{Max}\{P(e)\mid e\in Pre(t)\}$.　证毕.

通过定义累计成功概率函数 P，用最大可达概率替代成功渗透该节点的实际概率，我们有效地避免了渗透节点之间的相关性对概率计算的影响. 下面我们看一个简单的例子，攻击图如图 3 所示，$c_1\sim c_4$ 为初始条件. 其中条件节点的自身概率 d 全为 1，e_1、e_2、e_3 的自身概率分别为 0.3、0.5、0.7. 那么 e_1 的最大可达概率为 $d(e_1)\cdot\prod_{c\in Pre(e_1)}P(c)=0.3\times1\times1=0.3$，同理可得 e_2 的最大可达概率为 0.5. 对于 c_6，最大可达概率为 $d(c_6)\cdot Max\{P(e)\mid e\in Pre(c_6)\}=1\times\mathrm{Max}(0.3,0.5)=0.5$. 其余各节点的最大可达概率同理可以得到.

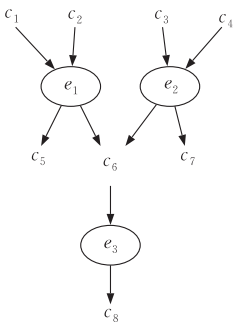

图 3　基于攻击图的累积成功概率计算示例

若攻击图中含有循环路径，我们必须避免处于循环路径中的节点的概率值的重复计算. 我们只需要特别关注于循环路径的入口节点的计算. 一旦循环路径的入口节点的最大可达概率计算完成，就可以按照不含循环路径的攻击图的概率计算方法计算循环路径其它节点的最大可达概率. 如对于图 2 中的循环路径 $c_2\rightarrow e_3\rightarrow c_3\rightarrow e_2\rightarrow c_2$，如果该循环路径的入口节点 c_2 的最大可达概率已经计算完成，那么 e_3、c_3、e_2 的最大可达概率就可以按照定义 2 中定义的累计成功概率函数 P 计算得到. 下面，我们给出攻击图中各节点最大可达概率计算算法. 该算法运用带有计数器的广度优先搜索策略计算已满足条件的节点最大可达概率，并按照定义 2 定义的累计成功概率函数 P 计算已满足计算条件的节点的最大可达概率，当所有的已满足计算条件的节点的最大可达概率计算完毕，算法将停在各个循环路径的入口节点上.

对于各个循环路径的入口节点，我们采用迭代函数 $loop\_prob(v_i)$ 计算入口节点的最大可达概率. 该函数的主要思想是从指定的节点 t 出发，采取前向搜索的方式和深度优先的搜索策略迭代计算 t 所有父节点的最大可达概率，然后在此基础上计算 t 的最大可达概率. 在计算过程中为了保证沿着不含循环路径的有效攻击路径，引入了节点集合 $trace$ 来存放已搜索的条件节点的轨迹. 从 t 节点出发在深度搜索迭代之前如果该节点是条件节点则加入到该轨迹中，迭代结束后将该节点从轨迹中擦除. 迭代前若该节点已在轨迹中，则说明继续该次迭代所沿攻击路径含有循环，故终止该次迭代. 由于函数 $loop\_prob(v_i)$ 在迭代的过程中，部分路径上的节点的最大可达概率已经计算完毕，不需要继续进行迭代计算，所以我们只对最大可达概率未计算完毕的节点进行迭代，提高了算法运行的效率. 最后按照定义 2 定义的累计成功概率函数 P 计算剩余的其它节点的最大可达概率.

算法 2.　最大可达概率计算算法.

Input：每个节点都具有自身概率的简化后的攻击图 $G=(C_o\cup C_d,T,E)$

Output：攻击图中各节点的最大可达概率

Method：

1. For each $t_i\in C_o\cup C_d\cup T$ Do
2. $Count[t_i]=Pre(t_i)$的数目；$number[t_i]=0$；$M(t_i)=$false；//$number[t_i]$为 t_i的计数器
3. If $t_i\in C_d$ Then
4. $P(t_i)=0$；//$P(t_i)$为 t_i的累积成功概率，为方便 18 行的条件判断，将其初始化为 0
5. If $t_i\in T$ Then
6. $P(t_i)=1$；//为方便 12 行的计算，将其初始化为 1
7. For each $c_i\in C_o$ Do
8. $Insert(q,c_i)$；$P(c_i)=1$；$M(c_i)=$true；//将 c_i插入队列 q，将初始节点的最大可达概率设为 1
9. For each $t_i=Delete(q)$ Do //出队
10. For each $m_j\in Post(t_i)$ Do
11. If $m_j\in T$ Then //按照定义 2 定义的累计成功概率函数 P 计

```
        算渗透节点的最大可达概率
12.     P(m_j)=P(m_j)*P(t_i);
13.     number[m_j]=number[m_j]+1;
14.     If number[m_j]=Count[m_j] Then
15.       P(m_j)=P(m_j)*d(m_j);
                  //d(m_j)表示 m_j的自身概率
16.       Insert(q,m_j);M(m_j)=true;
          //将计算完成的渗透节点加入队列,并将
            该节点标识为 true
17.   If m_j∈C_d Then
          //按照定义 2 定义的累计成功概率函数
            P 计算条件节点的最大可达概率
18.     If P(m_j)<P(t_i) Then
19.       P(m_j)=P(t_i);
20.     number[m_j]=number[m_j]+1;
21.     If number[m_j]=Count[m_j] Then
22.       P(m_j)=P(m_j)*d(m_j);
23.       Insert(q,m_j); M(m_j)=true;
          //将计算完成的渗透节点加入队列,并
            将该节点标识为 true
24. For each 0<number[v_i]<Count[v_i]的节点 v_i Do
                  //v_i为每个循环路径的入口节点
25.  loop_prob(v_i);M(v_i)=true;
                  //采用迭代函数 loop_prob(v_i)计算
26. For 剩余的其它节点 t_i Do
27.  按照定义 2 计算 t_i的最大可达概率;
28. Return 攻击图各节点的最大可达概率 P(t_i);
```

其中迭代函数 $loop\_prob(v_i)$ 定义如下：

```
Function loop_prob(v_i)
1. If v_i∈C_d Then
2.   If v_i∈trace Then
3.     Return P(v_i)=0;
4.   trace=trace∪{v_i};
5.   For each τ_j∈Pre(v_i) Do
6.     If M(τ_j)=false Then
          //如果节点 τ_j未计算完毕,则递归计算 τ_j
7.       P(τ_j)=loop_prob (τ_j);
8.   P(v_i)=Max(P(τ_j));
9.   trace=trace\{v_i};
10. If v_i∈T Then
11.   For each a_j∈Pre(v_i) Do
12.     If M(a_j)=false Then
          //如果节点 a_j未计算完毕,则递归计算 a_j
13.       P(a_j)=loop_prob(a_j);
14.   P(v_i)=d(v_i)*∏P(a_j);
15. Return P(v_i).
```

算法 2 的 1～6 行，将每个节点的父节点数目保存在 Count 中，初始化最大可达概率（为方便之后的计算，条件节点和渗透节点的最大可达概率分别初始化为 0 和 1），并将各个节点标记为 false，说明该节点没有计算完毕；算法的 7～9 行将初始条件节点插入队列 q，将其最大可达概率设为 1，并依次出队，队列 q 中的元素是已经计算完毕的节点；算法的 10～16 行，按照定义 2 定义的累计成功概率函数 P 计算渗透节点的最大可达概率，将计算完成的渗透节点加入队列 q，并将该节点标识为 true；算法的 17～23 行，按照定义 2 定义的累计成功概率函数 P 计算条件节点的最大可达概率，将计算完成的渗透节点加入队列，并将该节点标识为 true；算法的 24～25 行，采用迭代函数 $loop\_prob(v_i)$ 计算入口节点的最大可达概率；算法的 26～27 行，按照定义 2 定义的累计成功概率函数 P 计算剩余的其它节点的最大可达概率；算法的 28 行返回各个节点的最大可达概率结果.

对于算法 2 计算出的各节点的最大可达概率，我们给出如下定理.

定理 2. 通过算法 2 计算出的各节点的累计成功概率为最大可达概率.

证明. 对于循环路径中的入口节点 v，若 $v\in T$，v 的最大可达概率是它的所有父节点的最大可达概率与 v 的自身概率之积，即 $d(v)\cdot\prod_{a\in pre(v)}P(a)$；若 $v\in C_d$，$trace$ 存放了 v 的所有后继渗透节点的后果. 若 $v\in trace$，则说明把 v 加入到该路径中会产生圈，故终止计算；否则，根据定义 2 和定理 1 可知，v 的最大可达概率是所有父节点 τ 的最大可达概率中的最大值，即 $\mathrm{Max}\{P(\tau)\mid\tau\in Pre(v)\}$. 对于其它节点，根据定理 1 可知，由累计成功概率函数 P 计算出的各节点 t 的概率为最大可达概率. 证毕.

3.4 算法复杂度分析

若攻击图中节点的数目为 n，有向边的数目为 e，循环路径的入口节点数目为 n_1，攻击图中的路径的最大深度为 L，每个节点的最大父节点数为 M. 通过分析，我们可知算法 2 的时间复杂度为 $O((n-n_1+e)+n_1\times M^{2L+1})$，其中，计算攻击图中循环路径的入口节点的时间复杂度为 $O(n_1\times M^{2L+1})$，计算其它节点的时间复杂度为 $O(n-n_1+e)$). 由大规模网络产生的攻击图中，含有成千上万的节点和有向边，循环路径的入口节点的数目在总节点数目中只占很小的比例，所以我们的算法的时间复杂度近似为 $O(n)$，远远小于陈锋[11]算法的时间复杂度（陈锋[11]算法的时间复杂度为 $O(n\times M^{2L+1})$）.

与陈锋[11]的算法相比，我们的算法可以适用于更加复杂的攻击图，而且陈锋[11]的算法不能处理攻击图中含有不可达路径的情况；与 Wang[10]的方法相比，我们采用最大可达概率有效回避了节点之间的相关性导致的概率错误计算问题，并给出了计算各节点概率的详细算法；与 Homer 等[12]的方法相比，我们的方法不仅适用于小规模网络，而且适用于中、大规模网络.

4　实　验

本文以图 4 中的真实网络和 3 个不同规模和复杂性的模拟网络，验证所提方法的合理性和有效性. 实验环境为 AMD Athlon 64 PC 3600+(2.09GHz)、2GB 内存、Window XP，算法在 Eclipse3.4 下实现.

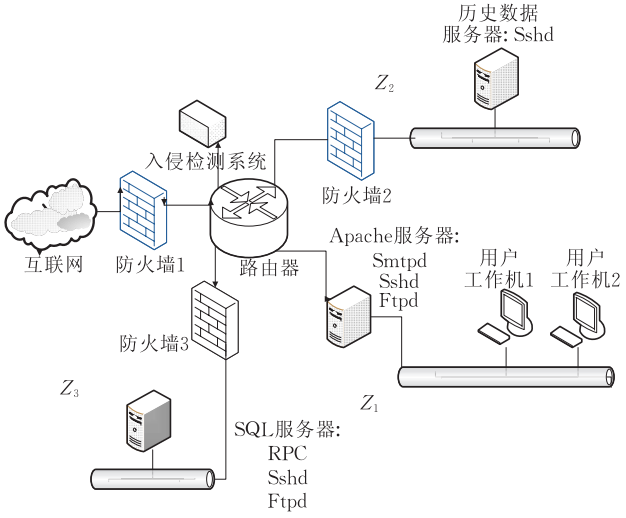

图 4　实验网络部署图

图 4 是实验网络部署图，Z_1 是对外交流区域，有一台 Apache 服务器和两台用户工作机，其中 Apache服务器上运行 Apache 对外提供 Web 服务，Smtpd 提供邮件服务，Sshd 提供远程管理控制服务，Ftpd 提供文件传输服务. Z_3 是内部服务区，有一台 SQL 服务器，为 Apache 服务器上的 Apache 提供数据库 SQL Server 服务以及为用户工作机提供 RPC、Sshd 和 Ftpd 服务. Z_2 是数据备份区，有一台历史数据服务器，运行的 SQL Server 数据库每天定时对 Z_3 中的数据库进行备份. Apache 服务器可访问任意主机，也可被任意主机访问；用户工作机可访问 SQL 服务器；SQL 服务器可访问历史数据服务器；同一区域的所有主机之间可以互相访问.

使用 Nessus 脆弱点扫描器对各网段内进行扫描，得到各主机上的脆弱点信息，如表 1 所示. A 类渗透节点对应的脆弱点的自身概率值可以通过查询 NVD 数据库中的“AccessComplexity”属性值获得，查询结果如表 2 所示；B 类渗透节点和条件节点，其自身概率都设为 1.

表 1　各主机上的脆弱点信息

主机	CVE 编号	公布时间
Apache 服务器	CVE-2006-3747	2006.07.20
Apache 服务器	CVE-2002-0640	2002.07.03
SQL 服务器	CVE-2002-1123	2008.09.25
SQL 服务器	CVE-2008-4250	2002.08.07
SQL 服务器	CVE-2005-2558	2005.08.16
历史数据服务器	CVE-2002-1123	2002.08.07

表 2　各脆弱点被成功渗透的自身概率

CVE 编号	自身概率 d
CVE-2006-3747	0.35(High)
CVE-2002-0640	0.71(Low)
CVE-2002-1123	0.71(Low)
CVE-2008-4250	0.71(Low)
CVE-2005-2558	0.71(Low)
CVE-2002-1123	0.71(Low)

IDS 部署在路由器的某个镜像端口上检测网络中可能的攻击行为. 从 IDS 预警事件中分析发现了以 202.103.96.21 为源主机，以用户工作机 1 为目的主机的预警事件，程度为严重级，以 202.103.96.7 为源主机，以用户工作机 2 为目的主机的预警事件，程度为普通级，由此我们假设攻击者的初始攻击能力为攻击者在用户工作机 1 具有最高级用户权限，在用户工作机 2 具有普通用户权限.

图 5 为我们产生的攻击图. 由于图中各节点对应的语义过于冗长，本文不做展示. 该实验网络包含 5 台主机，6 个脆弱点，其攻击图包含大量的纯文字(条件节点)、椭圆(渗透节点)和有向边，并且含有循环路径，影响管理员对攻击图进行进一步的分析. 应用第 3 节提出的算法 1，我们对该攻击图进行化简，删除攻击图中不可能到达的节点和有向边，得到简化后的攻击图如图 6 所示. 图 6 中的攻击图与图 5 相比，更加简洁，而且不含不可达路径. 通过简化，得到的攻击图含有更少的节点和有向边，提高了之后的计算基于攻击图中各节点最大可达概率的效率.

应用第 3 节提出的算法 2，计算图 6 中简化后的攻击图的各条件节点和渗透节点的最大可达概率，程序执行到算法 2 的第 24 行，会停留在循环路径的入口节点 c_{27} 和 c_{31}，然后通过改进的迭代函数计算 c_{27} 和 c_{31} 的最大可达概率，最后按照带有计数器的广度优先搜索算法继续计算其余节点的最大可达概率. 结果如表 3、表 4 所示.

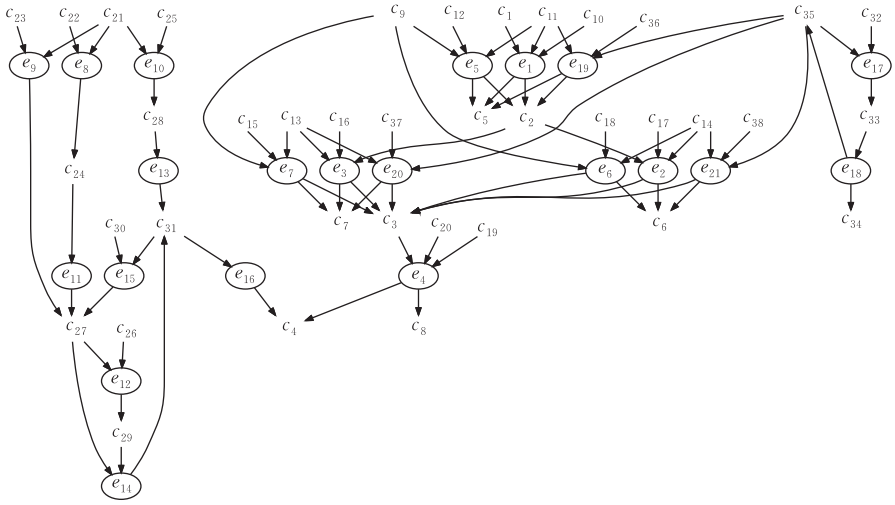

图 5 原始的攻击图

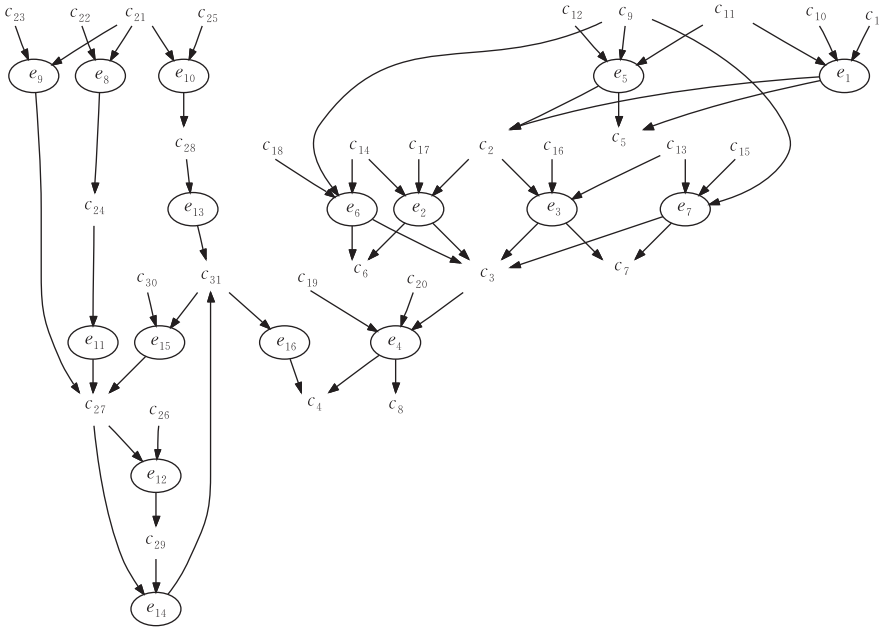

图 6 简化后的攻击图

表 3 各条件节点的最大可达概率

条件节点	累计成功概率	条件节点	累计成功概率
c_1	1	c_{17}	1
c_2	0.35	c_{18}	1
c_3	0.71	c_{19}	1
c_4	0.71	c_{20}	1
c_5	0.35	c_{21}	1
c_6	0.71	c_{22}	1
c_7	0.71	c_{23}	1
c_8	0.50	c_{24}	1
c_9	1	c_{25}	1
c_{10}	1	c_{26}	1
c_{11}	1	c_{27}	1
c_{12}	1	c_{28}	1
c_{13}	1	c_{29}	1
c_{14}	1	c_{30}	1
c_{15}	1	c_{31}	1
c_{16}	1		

表 4 各渗透节点的最大可达概率

渗透节点	累计成功概率	渗透节点	累计成功概率
e_1	0.35	e_9	0.71
e_2	0.25	e_{10}	1
e_3	0.25	e_{11}	1
e_4	0.50	e_{12}	1
e_5	0.35	e_{13}	1
e_6	0.71	e_{14}	1
e_7	0.71	e_{15}	0.71
e_8	1	e_{16}	0.71

根据各个不同节点的最大可达概率信息，目标网络的管理员就可以知道哪些节点容易被成功渗透，哪些节点很难被成功渗透，然后按照自己的需要，采取相应的弥补措施，保证目标网络的相对

安全.

为了测试本文所提算法 2 的运行效率，我们建立了 3 个不同规模和复杂性的网络模型，测试的结果如表 5 所示.

表 5　不同算法的测试结果对比

测试模型	目标网络中的主机数	攻击图中最大循环路径中的节点数	运行时间/s		
			陈锋算法[11]	Homer 算法[12]	本文算法 2
A	5	43	7	36	3
B	10	67	689	3654	14
C	20	94	3856	29832	58

随着网络模型中主机数的增多，由网络模型产生的攻击图的节点数和有向边将急剧增多，攻击图中的循环路径也变得越来越复杂. 从测试结果中我们可以看出，陈锋的算法由于对每一个节点都用迭代函数进行计算，复杂度会随着攻击图复杂性的提高而大幅上升；Homer 的算法将含有循环路径的攻击图转换为等价的不含循环路径的攻击图，增加了很多的节点和有向边，同时在考虑渗透之间的相关性问题时，联合概率的获得和计算十分复杂，无法适应复杂的目标网络；由于 Wang 的文中[10]没有给出计算各节点概率的详细算法，我们未对该方法做模拟实验；本文的算法 2 实际上是一个带有计数器的广度优先搜索算法，仅在循环路径的入口节点上才采用改进的迭代函数进行计算，同时采用最大可达概率有效避免了相关性导致的概率错误计算问题，可以适用于更为复杂的攻击图各节点的概率计算，具有很好的扩展性.

5　结束语

本文讨论了目标网络中攻击图存在的两个挑战性问题，通过删除攻击图中不可达路径来简化攻击图，增强了管理员对攻击图的理解，并有利于进一步分析攻击图；采用最大可达概率有效避免了渗透节点之间的相关性导致的概率错误计算问题；提出最大可达概率计算算法解决了攻击图中存在的循环路径所导致的攻击图难以理解和概率重复计算问题；在真实实验和模拟实验中验证了所提概念和方法的合理性.

未来的工作包括进一步研究攻击图中节点之间的相关性问题，在真实的大规模网络中进一步验证本文所提算法的合理性和有效性以及深入研究基于攻击图的网络安全分析.

参　考　文　献

[1] Xing Xu-Jia, Lin Chuang et al. A survey of computer vulnerability assessment. Chinese Journal of Computers, 2004, 27(1): 1-11(in Chinese)
(邢栩嘉，林闯等. 计算机系统脆弱点评估研究. 计算机学报，2004，27(1)：1-11)

[2] Singhal A, Ou X M. Security risk analysis of computer networks: Techniques and challenge//Proceedings of the 16th ACM Computer and Communications Security(CCS). Chicago, USA, 2009

[3] Wang L Y, Singhal A, Jajodia S. Measuring the overall security of network configurations using attack graphs//Proceedings of the 21th IFIP WG 11.3 Working Conference on Data and Applications Security (DBSec). Redondo Beach, CA, USA, 2007: 98-112

[4] Wang L Y, Singhal A, Jajodia S. Toward measuring network security using attack graphs//Proceedings of the 3rd International Workshop on Quality of Protection(QoP). Alexandria, USA, 2007: 49-54

[5] Feng Ping-Hui, Lian Yi-Feng, Dai Ying-Xia et al. A vulnerability model of distributed systems based on reliability theory. Journal of Software, 2006, 17(7): 1633-1640(in Chinese)
(冯萍慧，连一峰，戴英侠等. 基于可靠性理论的分布式系统脆弱性模型. 软件学报，2006，17(7)：1633-1640)

[6] Shi Jin, Guo Shan-Qing et al. An intrusion response method based on attack graph. Journal of Software, 2008, 19(10): 2746-2753
(石进，郭山清等. 一种基于攻击图的入侵响应方法. 软件学报，2008，19(10)：2746-2753)

[7] Mehta V, Bartzis C, Zhu H F. Ranking attack graphs//Proceedings of the 9th International Symposium on Recent Advances in Intrusion Detection(RAID). Hamburg, Germany, 2006: 127-144

[8] Sawilla R, Ou X M. Identifying critical attack assets in dependency attack graphs//Proceedings of the 13th European Symposium on Research in Computer Security(ESORICS). Malaga, Spain, 2008: 18-34

[9] Ou X M, Boyer W F. A scalable approach to attack graph generation//Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS). Alexandria, USA, 2006: 336-345

[10] Wang L Y, Tania I. An attack graph-based probabilistic security metric//Proceedings of the 22nd Annual IFIP WG 11.3 Working Conference on Data and Applications Security (DBSec). London, UK, 2008: 283-296

[11] Chen Feng. A hierarchical network security risk evaluation approach based on multi-goal attack graph [Ph. D. dissertation]. National University of Defense Technology, Changsha, 2009
(陈锋. 基于多目标攻击图的层次化网络安全风险评估方法研究[博士学位论文]. 国防科技大学, 长沙, 2009)

[12] Homer J, Ou X M, Schmidt D. A sound and practical approach to quantifying security risk in enterprise networks. Kansas State University: Technical Report 2009-3, 2009

YE Yun, born in 1981, Ph. D. candidate. His research interests focus on network security.

XU Xi-Shan, born in 1963, professor. His research interests include network security, software engineering and software reliability technology.

JIA Yan, born in 1961, professor, Ph. D. supervisor. His research interests include network information security, database and data mining.

QI Zhi-Chang, born in 1942, professor, Ph. D. supervisor. His research interests include software engineering and software architecture.

Background

An essential type of security risk analysis is to determine the level of compromise possible for critical resources in a network. This is a complex task as it depends on the network topology, security policy in the network as determined by the placement of firewalls, routers and switches and on vulnerabilities in hosts and communication protocols. There are two challenges existing in computing Probability in attack graphs, one is difficult to understand the attack graphs and the Probabilistic re-computing caused by cyclic paths in attack graphs, the other is probabilistic incorrect computing caused by shared dependencies in exploits. At present, only a few studies considered the two challenges, overall, the methods which were provided by these studies were too complicated, not suited to large-scale network. This paper presents a methodology for security risk analysis that is based on the model of attack graphs and the Common Vulnerability Scoring System (CVSS), simplifies the attack graphs by removing unreachable paths, and proposes the concept and computing approach of maximum reachable probability which solved the problem of probabilistic re-computing and successfully avoided the problem of probabilistic incorrect computing. The real experiment and simulation verified that the methods can be adapted to a large-scale network.

This paper was supported by the National High Technology Research and Development Program (863 Program) of China under grant Nos. 2007AA010502, 2009AA01Z436. These research topics aimed actual requirements of situation analysis and prediction of botnets, worms, DDoS and other network security events in large-scale networks and enterprise networks, developed network security situation analysis and prediction system for large-scale networks and enterprise networks. This paper is focused on the network security situation analysis in enterprise networks, and assessed the network security situation based on attack graph model. The attack graphs is simplified, to solve the problems of probabilistic re-computing and probabilistic incorrect computing. These studies ensure accurate analysis of network security situation for follow-up analysis in enterprise networks.