基于面向对象方法的攻击知识模型

诸葛建伟 徐 辉 潘爱民
(北京大学计算机科学技术研究所 北京 100871)
(zhuge_jianwei@icst.pku.edu.cn)

摘 要 提出了一个可应用于安全主动防御体系的攻击知识模型,并保证其实用性、可复用性和可扩展性.攻击知识以AKDL(attack knowledge description language)攻击知识描述语言来描述,并基于面向对象方法中的属性、方法、关系等概念,对攻击的数据、操作以及攻击之间的关联关系进行建模,一个AKDL→C++编译器将由AKDL语言描述的攻击知识编译成基于组件对象技术的C++攻击类代码,从而实现AKL(attack knowledge library)攻击知识库,并以API接口的形式提供给安全主动防御体系中各个安全组件使用

关键词 网络安全;攻击;攻击描述语言;知识库;面向对象

中图法分类号 TP309

An Attack Knowledge Model Based on Object-Oriented Technology

ZHUGE Jian-Wei, XU Hui, and PAN Ai-Min
(*Institute of Computer Science and Technology, Peking University, Beijing* 100871)

Abstract In this paper, an attack knowledge model that can be applied in the initiative defense architecture is proposed, and the model's practicability, re-useability and extendibility are achieved. The attack knowledge is described using attack knowledge description language(AKDL), and the data, operations and relations of attacks are modeled based on object-oriented technology. An AKDL to C++ compiler is used to compile the attack knowledge in AKDL language into C++ source code of attack component object, then the attack knowledge library(AKL) is implemented, which can provide the attack knowledge to the various security modules through APIs.

Key words network security; attack; attack description language; knowledge library; object-oriented

1 引 言

网络攻击技术正发生着日新月异的变化,一些复杂的攻击方法不断涌现,如2000年2月使得Yahoo,eBay等知名网站停止服务的分布式拒绝服务攻击;同时这些复杂的攻击方法通常都实现为攻击脚本或攻击工具并在互联网上广泛流行,这使得拥有很少专业知识的骇客们非常轻松地就能获得这些攻击脚本和工具并发起网络攻击事件;网络攻击技术还与病毒、蠕虫技术相融合,如泛滥一时的尼姆达病毒,从而给网络安全造成了前所未有的巨大危害.网络安全防护工具要想维护网络安全,在攻击发生前能通过漏洞扫描和安全评估找到潜在的安全漏洞并加以修补,或在遭受网络攻击时正确地识别出正在进行的和将要发生的攻击行为,并做出正确的反应动作以抵御攻击或减少损失,都必须对攻击知识有深入的了解,因此我们需要对攻击知识进行建模.

本文提出了一种可应用于安全主动防御体系的攻击知识模型,并保证其实用性、可复用性和可扩展性.我们以符合XML规范的AKDL(attack knowledge description language)攻击知识描述语言来描述攻击

收稿日期:2003-06-12;修回日期:2003-10-10
基金项目:国家"八六三"高技术研究发展计划基金项目(863-301-06-03)

知识, 用面向对象的方法对攻击知识进行建模, 并将AKDL语言描述的攻击知识通过一个AKDL→C++编译器编译成基于组件对象技术的C++攻击类代码, 从而实现AKL(attack knowledge library)攻击知识库, 然后以API接口的形式提供给安全主动防御体系中的各个安全组件.

本文的组织结构如下: 第2节介绍了一些相关的工作并给出了本文提出的攻击知识模型的意义. 第3节描述攻击知识模型, 包括面向对象的建模方法、AKDL攻击知识描述语言、攻击知识库的具体实现及其在安全主动防御体系中的应用, 最后在第4节, 我们对本文进行总结并给出下一步的工作.

2 相关的工作

目前的安全防护工具大多功能单一, 并只关注与之相关的攻击知识. 漏洞扫描程序的目标是检测出当前网络或系统中是否存在可供攻击者利用的安全漏洞, 因此它只需要了解已经发现的全部安全漏洞及其扫描脚本, 然后执行这些扫描脚本, 以确定出目标网络或系统中是否存在已知的安全漏洞. 这些已知的安全漏洞通常以漏洞库的方式提供给漏洞扫描程序, 著名的有CVE, Bugtraq和CERT/CC的漏洞报告.

误用型入侵检测系统通过匹配已知攻击方法的特征对攻击行为进行检测, 因此只需要以检测为目标, 对各种已知攻击的特征进行建模, 如Snort[1]的入侵规则集, 但其仅仅通过单包特征对攻击进行检测, 存在着较高的误报率和漏报率. STATL语言[2]基于状态和状态转移对攻击行为进行描述, 为NetSTAT[3]网络入侵检测系统提供入侵特征库; ESTQ[4]方法通过〈事件, 协议状态, 时间关系, 数量关系〉对网络协议攻击进行描述; IDIOT[5]则采用了有色Petri网对入侵进行建模和检测. 上述建模方法只关注于单一攻击行为的检测, 只能简单地发出各个攻击事件的报警, 而不能为管理员提供正在发生的攻击场景的清晰描述, 因此需要进一步在攻击知识中描述攻击者的攻击目标和各个攻击步骤之间的关联关系. 攻击树[6]方法采用与节点和或节点表示攻击步骤之间的关系; 节点分层拓扑结构[7]是对攻击树方法的改进, 其定义了目标层、状态层和事件层3个节点层次, 并引入了节点间固有联系和外在联系两种关联关系, 以描述多步骤的复杂攻击; JIGSAW[8]则把攻击行为看做为另一攻击提供一组能力(capability), 通过一个需求/提供模型来刻画攻击行为之间的关联关系.

当前的安全评估则主要通过安全测试组对目标网络或系统进行人工的攻击评测, 而没有可以自动进行攻击并对攻击结果进行测评的评估工具.

上述功能单一的传统安全防护工具并不能适应动态变化的、多维互联的网络环境, 我们需要一个综合网络防火墙、通信加密、病毒防范、安全评估、安全审计、入侵检测、入侵预警、实时响应、诱骗以及灾难恢复等安全技术的安全主动防御体系, 各个安全组件之间能够通信和互动, 进行智能化的协作, 从整体上保证网络安全.

在智能协作的安全主动防御体系中, 一个包括攻击描述、漏洞扫描、攻击防护、攻击脚本、攻击验证、攻击检测、攻击关联、攻击反应等的攻击知识库必不可少, 它为各个安全组件的内部实现和安全组件之间协作的智能化提供了知识基础.

LAMDBA攻击描述语言[9]通过前提、后果、攻击步骤、检测过程和扫描方法对一个攻击进行描述, ADeLe语言[10]则包括了攻击脚本语言、攻击检测语言、攻击关联语言和攻击反应语言4部分. 但二者都不足以为整个安全主动防御体系提供全面的知识基础, 同时二者仅仅给出了一种对攻击知识进行描述的语言, 并未提供对攻击知识、攻击之间的关联关系进行建模的方法.

本文提出的攻击知识模型借鉴了已有攻击建模方法的优点, 将面向对象的攻击知识建模方法与AKDL攻击知识描述语言相结合, 不仅能够为安全主动防御体系中智能化的安全组件提供全面的知识基础, 而且保证了其实用性、可复用性和可扩展性.

3 攻击知识模型

3.1 攻击知识建模方法

3.1.1 基本概念

我们采用面向对象的方法[11]对攻击知识进行建模, 首先定义以下一些基本概念:

定义1. 攻击对象: 我们将每个攻击实例封装为一个攻击对象, 表示为一个四元组〈N, R, V, M〉, 其中N为一个字符串, 作为该攻击对象的名称; R为关系集, 规定了该攻击对象和其他攻击对象之间的关联关系; V为属性集, 标识该攻击对象中的数据, M为方法集, 描述了攻击对象拥有的操作.

定义2. 攻击类是对拥有相同名称、关系、属性

和方法的同类攻击对象的集合，同样表示为四元组〈N, R, V, M〉.

定义 **3**. 属性是对攻击对象中的某个数据的描述，表示为一个三元组〈N, T, V〉，其中 N 为该属性的名称，T 为属性值的类型，V 则为属性的值. 我们以 $a_1.v_1$ 来表示攻击对象 a_1 的 V_1 属性的具体值. 对于一个攻击类 A，$\forall a_1, a_2 \in A$ 有 $a_1.v = a_2.v$，则称属性 V 为攻击类 A 的静态属性，否则称为动态属性.

定义 **4**. 方法是对攻击对象某个操作的描述，表示为一个二元组〈N, M〉，其中 N 为该方法的名称，M 为该方法的实现，在模型中并不指定 M 的具体定义，允许对方法的多种描述方式.

定义 **5**. 关系描述了攻击对象间或攻击类之间的关联关系，包括直接关联关系和间接关联关系，直接关联关系指直接可见的关系，包括抽象-具体关系和组合-部分关系两种；而间接关联关系指攻击对象通过其前提、后果关联起来的关系.

定义 **6**. 抽象-具体关系：如果攻击类 A 中的全部攻击对象都是攻击类 B 中的攻击对象，而且攻击类 B 中存在不属于攻击类 A 的攻击对象，即 $A \subset B$，则攻击类 A 称为攻击类 B 的具体类，而 B 是 A 的抽象类.

定义 **7**. 组合-部分关系：如果攻击对象 A 是攻击对象 B 的一个组成部分，则称 B 为 A 的组合对象，A 为 B 的部分对象，并把 B 和 A 之间的关系称为组合-部分关系. 攻击类与攻击类之间的组合-部分关系指一个攻击类的攻击对象，以另一个攻击类的攻击对象为其组成部分.

定义 **8**. 组合逻辑指的是一组部分攻击对象合成一个组合攻击对象时必须满足的条件，表示为一个二元组〈A, LR〉，其中 A 指定了组成该组合攻击对象的部分攻击对象集合，LR (logic relation)为各个部分攻击对象的属性值应满足的逻辑关系，为一个布尔表达式.

定义 **9**. 系统状态：系统中与安全相关的信息，表示为 $P(X)$的形式，其中 P(predication)为谓词，X 为参数集，如 *open*(*host*, *port*)表示 *host* 主机的 *port* 端口是否打开，系统状态包含状态检查和状态设置两个操作方法，其中状态检查为取得 $P(X)$的真假值，而状态设置则是设定 $P(X)$的真假值，如 *open*(*host*, *port*)= true 表示改变 *open*(*host*, *port*)这个系统状态的值为真.

定义 **10**. 攻击者状态：描述攻击者拥有的系统安全知识及资源，与系统状态类似，表示为 $P(X)$的形式，如 *access*(*host*, *root*)表示攻击者可以以 *root* 的身份访问 *host* 主机.

定义 **11**. 系统安全态势：整个系统及攻击者所有与安全相关的信息集合，即系统状态与攻击者状态的全集.

定义 **12**. 状态断言：由一个或多个系统状态或攻击者状态组成的布尔表达式.

定义 **13**. 状态改变：对一个或多个系统状态或攻击者状态进行状态设置.

3.1.2 模型表示方法

我们用攻击对象封装所有攻击行为，以属性描述其数据，其中以静态属性描述类层次的属性，如攻击名称、类型、受攻击目标、利用的漏洞、攻击原理和过程、出现时间、防护方法、特征、前提、后果集等；以动态属性描述实例层次上的属性，如发生时间、后果、攻击源、攻击目标等. 其中前提(precondition)为一个状态断言，表示攻击发生必要的系统环境，如果其结果为真，则表示在当前的安全态势下，该攻击可以发生，反之则不能发生；后果(effect)集为一组状态改变的集合，表示该类攻击发生后所有可能发生的状态改变，比如一些缓冲区溢出攻击，可能成功的获取权限，也可能导致进程拒绝服务，同时也可能不会造成任何结果；而动态属性后果表示特定的攻击对象发生后对系统安全态势带来的状态改变.

攻击类的操作则用方法表示，如扫描方法指定了对目标进行扫描的操作，以确定其是否能抵抗此攻击；防护方法定义了对此攻击进行防护的一些措施，如升级软件以修补安全漏洞等；攻击方法则是攻击脚本或攻击方法的具体实现；验证方法则提供了对攻击效果进行验证的途径；检测方法对于输入的观察事件进行检测，确定其是否满足特征，从而检查攻击是否发生，抽象攻击类可以通过基于抽象特征的方法进行检测，而组合攻击类的检测则需要检查各个部分攻击是否都发生，而且满足其组合逻辑才能判断此组合攻击是否发生；反应方法则规定了预警到、检测到或事后对此攻击的反应动作，可能包括报警、审计、阻截、跟踪、反击、诱骗等.

以抽象-具体关系描述攻击方法和基于该攻击方法实现的攻击脚本和工具之间的关系，或者攻击分类和属于该分类的攻击方法及攻击工具之间的关系，相当于与或树中的或节点. 同时，抽象-具体关系结合系统的攻击分类法使得我们可以对庞大数量的攻击方法及工具构建一个完整的攻击知识库.

组合-部分关系描述复杂攻击和构成该复杂攻击的多个攻击步骤之间的关系, 并在复杂攻击类中定义组合逻辑, 确定多个部分攻击合成该复杂攻击时应满足的逻辑关系, 若不需满足任何逻辑关系, 则该组合-部分关系相当于与或树中的与节点.

而非节点可以通过系统状态和攻击者状态来实现, 如图 1 所示:

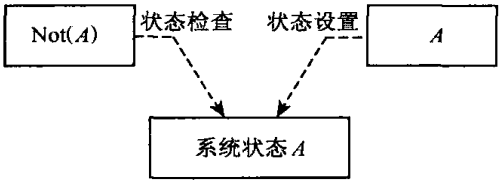

图 1　非节点的实现

事件 A 的发生对状态 A (记录事件 A 是否发生的状态类)进行设置, 而 Not(A) 事件的发生的前提是一个系统断言, 即状态 A 的状态值为假, 只有在事件 A 未发生的情况下, 此断言才为真, Not(A) 事件的前提才满足, 这也恰好符合非节点的语义, 非节点的实现和攻击类的两种直接关系(相当于或节点和与节点)结合, 保证了整个模型的完备性. 如异或节点可以如图 2 通过同时应用非节点和攻击类的两种直接关系来实现.

攻击类之间的间接关系指攻击通过其前提、后果关联起来的关系, 如 R2L(remote to local) 攻击使得攻击者获得本地账号访问权, 即设置"拥有本地账号访问权"这个攻击者状态为真, 而这又作为一个系统断言, 是 U2R(user to root)攻击的前提.

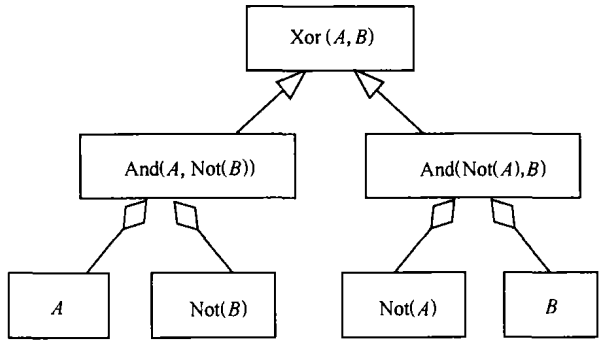

图 2　异或节点的实现

在我们的攻击知识模型中, 并不直接描述攻击之间的间接关联关系, 而是以系统状态与攻击者状态谓词的布尔表达式来定义攻击的前提及后果, 通过前提、后果属性中相同的状态来关联攻击.

如图 3 所示, 从系统角度上看, 系统遭受到攻击, 若此攻击的前提在系统状态 n 中满足, 则会导致其后果中的状态改变(包括系统状态和攻击者状态). 而从攻击者的角度来看, 攻击者会基于先前状态(即对系统的知识集和资源集)和他的攻击目标去选择前提已经成立的攻击行为并执行, 使得系统遭受攻击后, 其状态改变能够达到攻击目标或为其下一步攻击行为准备前提条件.

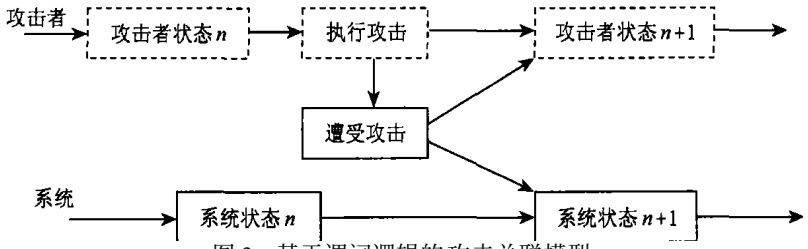

图 3　基于谓词逻辑的攻击关联模型

3.1.3　实例说明

图 4 给出了攻击知识库的一个示例图, 从这个示例图中, 我们就可以找到一种从远程攻击并完全控制一台开放着有 too-open 安全漏洞的 openssl 服务的主机的方法, 即通过端口扫描和漏洞扫描确定 openssl-too-open 攻击能奏效, 通过操作系统辨识取得发起攻击必需的 OS 类型参数, 然后运行 openssl-too-open 远程缓冲区溢出攻击脚本对目标主机进行攻击, 从而取得在目标主机上的 apache 账号使用权, 接着运行 efstool 本地缓冲区溢出攻击脚本, 使得存在缓冲区溢出漏洞的 efstool 工具溢出并执行传入的 shell code, 获得目标主机的 root 权限, 最后使用 Linux Root Kit 工具对内核进行木马化, 以完全控制该主机.

在这个示例图中, openssl-too-open 远程缓冲区溢出攻击作为远程缓冲区溢出攻击技术的一个实现脚本, 它们之间为具体-抽象关系, 而端口扫描、操作系统辨识和漏洞扫描与获取信息之间的关系为攻击分类上的从属关系, 也同样表示为具体-抽象关系. openssl-too-open 远程攻击作为一个复杂的攻击行为, 包含了端口扫描、操作系统辨识、openssl-too-open 漏洞扫描和 openssl-too-open 远程缓冲区溢出攻击 4 个部分攻击步骤, 因此它们之间的关系为组合-部分关系.

R2L 和 U2R 通过是否有本地访问权这一状态关联在一起;而 U2R 和安装后门之间也通过是否获得 root 权限这一状态关联起来

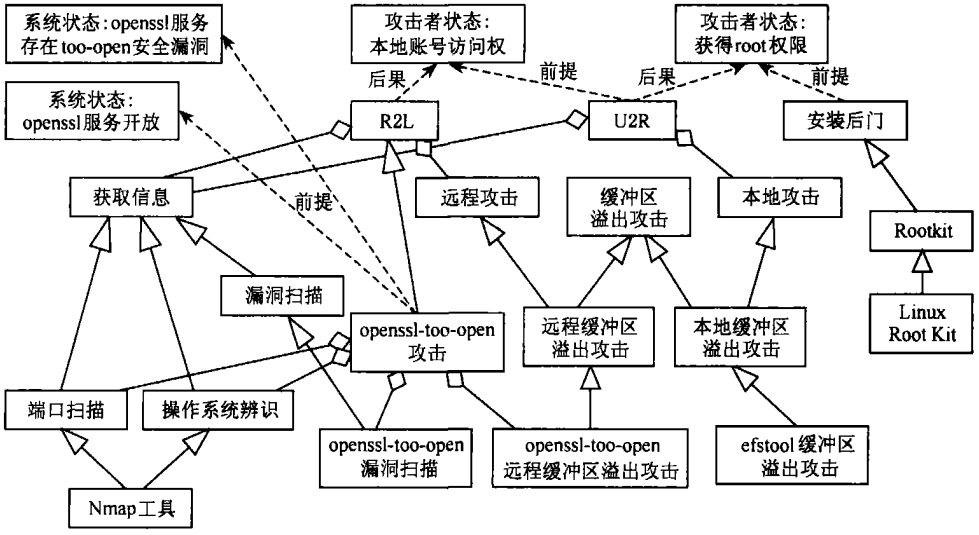

图 4 攻击知识库示例图

3.2 AKDL 攻击知识描述语言

上一节提供的攻击知识建模方法完全可以通过 C++ 语言编写所有已知的攻击类,但这样的实现方法使得攻击知识库缺乏一定的实用性和可扩展性,同时不复用已有的基于各种语言实现的攻击脚本和攻击工具,而在攻击知识库中重新用C++语言编写是完全不现实的. 因此我们提出了一种通用化的,能够复用已有的攻击脚本和攻击工具并且容易理解和编写的攻击知识描述语言 AKDL.

由于 XML 的易读性和结构化,我们将 AKDL 设计为一种符合 XML 规范的攻击知识描述语言,图 5 给出了AKDL 语言的 XML Schema[12] 结构图,logic-relation 作为攻击对象属性值之间关系的布尔表达式,不再进行进一步的详细定义.

其中 attack 标签中给出了攻击名称,其主体部分包括关联关系、静态属性、动态属性和操作方法四大部分.

关联关系部分描述了该攻击类在整个攻击知识模型库中和其他攻击类的抽象-具体关系和组合-部分关系. 由于抽象类可能并不关注其所有的具体类,但具体类必须清楚它所属的攻击分类或应用的抽象攻击方法,因此我们在具体类中指定其抽象类来描述抽象-具体关系. 而在组合-部分关系中,部分攻击类可能并不知道其作为某个复杂攻击类的组

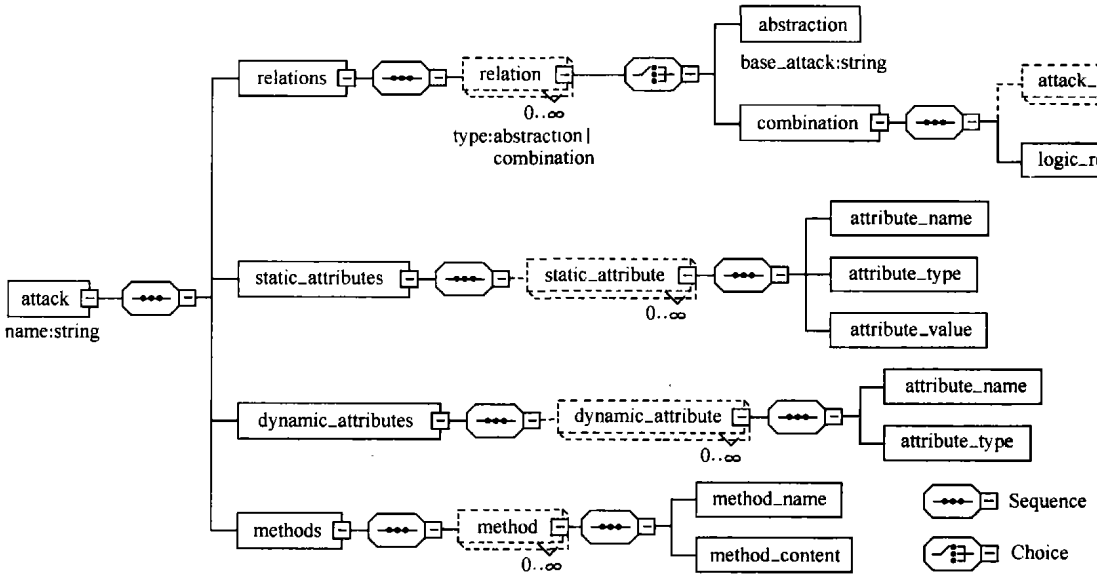

图 5 AKDL 语言的 XML Schema 结构图

成部分, 但组合的复杂攻击必须明确所包含的各个部分攻击对象, 因此我们在组合攻击类中以组合逻辑来描述组合-部分关系

静态属性部分定义了攻击类层次上的静态属性 动态属性部分则定义了攻击对象层次上的动态属性, 由于这些动态属性只能在运行时刻才会被赋值, 因此在 AKDL 语言中只需定义其名称和类型

操作方法部分包含了扫描、防护、攻击、验证、检测和反应 6 个方法的脚本语言. 我们采用以下两种方法尽量达到攻击知识库的实用性、可复用性和可扩展性

一是通过 Shell 命令方式(即 exec-shell-cmd 函数)对已有的基于 C, C++, Perl, Shell 脚本等语言的源码或二进制代码进行编译、执行, 以尽量复用这些资源

另外的一个途径是在攻击类的基类中实现一般化的扫描、防护、验证、检测和反应过程, 而在 AKDL 语言中通过用特定的输入参数调用这些一般化的过程, 使其成为特定攻击的扫描、防护、验证、检测和反应方法

3.3　攻击知识库的实现

攻击知识库的实现结构如图 6 所示, 通过一个 AKDL→C++ 编译器将用 AKDL 语言描述的攻击知识编译成为以C++ 源码表示的攻击类和相关的状态类, 然后由C++编译器将这些类编译成为二进制的库文件, 并以 API 接口的形式为安全主动防御体系中的各个安全组件程序提供攻击知识

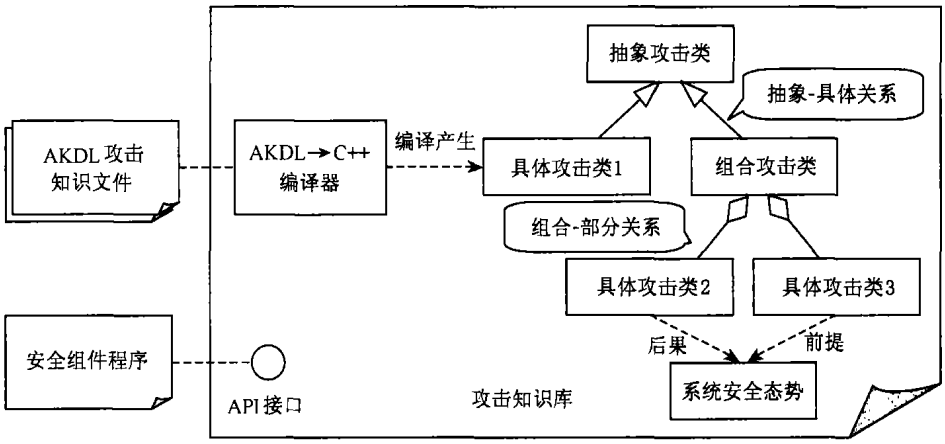

图 6　攻击知识库的整体结构

我们在 Linux 平台下基于组件对象技术实现了攻击知识库. 将每个攻击类实现为组件, 并提供统一的 IAttack 接口供客户程序调用, IAttack 接口包括了扫描、防护、攻击、验证、检测和反应方法, 同时提供读取属性和设置属性两个方法, 使得客户程序可以随时获取和设置攻击对象的各个属性的值

我们还实现了一个 AKDL→C++ 编译器 akdlc, 它可以将以 AKDL 语言描述的攻击类编译成组件类的C++代码, 并修改AKL 库的头文件、实现文件、Makefile 和组件位置信息配置文件, 这样我们只需重新编译和连接就可以把编译过的攻击类加入到攻击知识库中.

生成的攻击知识库以各个攻击组件类组合起来的组件库形式存在, 客户程序通过一个组件对象模型库就可以创建攻击组件对象, 然后调用 IAttack 接口中的方法使用该攻击组件.

利用组件对象模型(COM)实现攻击知识库, 使得编译之后的攻击知识库能够直接挂接到客户软件中, 而无需重新编译客户软件, 这使得攻击知识库和客户软件的开发和扩展都相互独立. 攻击知识库也可以被多个客户软件同时使用, 从而保证了攻击知识库的实用性、可复用性和可扩展性

3.4　攻击知识库的应用

攻击知识库可广泛应用于安全主动防御体系中的各个安全组件, 如漏洞扫描工具可以利用攻击知识库中每个攻击的攻击扫描方法对目标网络或系统进行漏洞扫描, 并可以通过攻击防护方法自动地对目标网络或系统进行修补加固.

入侵检测系统可以通过获取攻击知识库中的攻击检测方法对简单攻击进行检测, 同时也可以通过发现攻击的前提、后果之间的关联关系和应用规划识别技术对正在发生和将要发生的复杂攻击做出准确的检测和预警, 并可以根据攻击知识库中提供的攻击反应知识, 与自动反应模块协作, 做出正确的反应动作.

安全评估工具则更需要全面了解攻击知识, 首先需要通过对目标网络或系统进行攻击扫描, 以找出可以利用的安全漏洞. 然后根据扫描结果和对各

种攻击的前提后果等各种知识的学习,确定出对目标网络或系统的攻击目标,并进行攻击的自动规划,从而发起对目标的评测性攻击. 在每次攻击结束后,都必须对攻击效果进行验证,必要时对攻击规划进行调整. 攻击评估结束后,可以通过攻击效果对目标网络或系统进行安全等级评定. 在评估后,还应该通过攻击防护方法对目标网络或系统进行升级,从而避免这些安全漏洞被真正的攻击者利用.

同时,整个安全主动防御体系中的各个安全组件要实现智能化的协作,必须需要对攻击的各方面知识有深入的了解,而攻击知识库就扮演了这样的角色. 由此可见,攻击知识库在整个智能协作的网络安全主动防御体系中起着核心的关键作用.

4 结 论

本文提出了一个可应用于安全主动防御体系的攻击知识模型,并基于组件对象技术实现了一个实用、可复用和可扩展的攻击知识库.

本文的攻击知识模型可以直接应用于网络安全主动防御体系中,而不再是一个理论框架或一种攻击描述语言. 以AKDL语言来描述攻击知识使得攻击知识库可以复用已有的各种形式的攻击源码,同时作为一种与实现独立的攻击描述方法,AKDL语言描述的攻击知识也可以被其他的系统使用,因此具有很好的可复用性. 另外,AKDL语言很容易理解和编写,这使得用它来对攻击知识库进行更新和扩充并不困难,基于组件技术的实现方法使得各个攻击之间相互独立,攻击知识库与客户软件相互独立,所以具有良好的可扩展性.

本文的攻击知识模型还需要在以下方面进行进一步的研究工作并对其加以完善:

(1) 进一步完善AKDL攻击知识描述语言,使其更具表达能力和通用性;

(2) 对攻击的分类方法进行研究,给出系统化分类方法,以支持知识库构造;

(3) 本文提出了攻击知识库的构建方法,但真正实现一个完整可用的攻击知识库需要对已知攻击知识进行收集和描述,需要大量的、具体的工作.

参 考 文 献

1 M Roesch, C Green. Snort users manual(2.1.2). http://www.snort.org/docs/, 2004-03-31/2004-04-19

2 S T Eckmann, G Vigna, R A Kemmerer. STATL: An attack language for state-based intrusion detection. Journal of Computer Security, 2002, 10(1): 71~104

3 G Vigna, R A Kemmerer. NetSTAT: A network-based intrusion detection system. Journal of Computer Security, 1999, 7(1): 37~71

4 王晓程, 刘恩德, 谢小权. 攻击分类研究与分布式网络入侵检测系统. 计算机研究与发展, 2001, 38(6): 727~734
(Wang Xiaocheng, Liu Ende, Xie Xiaoquan. Attack classification research and a distributed network intrusion detection system. Journal of Computer Research and Development (in Chinese), 2001, 38(6): 727~734)

5 S Kumar, E H Spafford. A pattern matching model for misuse intrusion detection. The 17th National Computer Security Conf, Baltimore, MD, USA, 1994

6 B Schneier. Attack trees, modeling security threats. Dr Dobb's Journal of Software Tools, 1999, 24(12): 21~29

7 K Daley, R Larson, J Dawkins. A structural framework for modeling multi-stage network attacks. The Int'l Conf on Parallel Processing Workshops, Vancouver, B C, Canada, 2002

8 S J Templeton, K Levitt. A requires/provides model for computer attacks. The New Security Paradigms Workshop, Cork Ireland, 2000

9 F Cuppens, R Ortalo. Lambda: A language to model database of detection of attacks. In: Proc of the 3rd Int'l Workshop on the Recent Advances in Intrusion Detection, LNCS 1907. New York: Springer, 2000. 197~216

10 C Michel, L M'e. ADeLe: An attack description language for knowledge-based intrusion detection. In: Proc of the 16th Int'l Conf on Information Security. Dordrecht, Holland: Kluwer, 2001. 353~368

11 邵维忠, 杨芙清. 面向对象的系统分析. 北京: 清华大学出版社, 1998
(Shao Weizhong, Yang Fuqing. Object-Oriented System Analysis (in Chinese). Beijing: Tsinghua University Press, 1998)

12 W3C XML Schema Working Group. The XML schema specification (W3C recommendation). http://www.w3.org/XML/Schema#dev, 2001

诸葛建伟 男, 1980 年生, 博士研究生, 主要研究方向为网络与信息安全

徐 辉 男, 1977 年生, 博士研究生, 主要研究方向为网络与信息安全

潘爱民 男, 1970 年生, 副研究员, 硕士生导师, 研究方向为网络与信息安全、软件技术