

融合宏观与微观的双层威胁分析模型

孙澄, 胡浩, 杨英杰, 张红旗
(信息工程大学, 河南 郑州 450001)

摘要: 针对现有威胁分析模型无法兼顾高级安全威胁的宏观发展趋势及微观传播路径的问题, 建立了一种双层威胁分析模型 TL-TAM。模型上层刻画严重程度由低到高的威胁发展趋势, 下层融合技术漏洞攻击、社会工程攻击及网络扫描攻击, 刻画威胁传播路径。据此, 提出了威胁预测分析算法。实验结果表明, 模型能够对威胁传播进行多层次综合分析, 并且克服了基于攻击图的威胁分析模型局限于技术漏洞攻击的缺陷, 更加适用于高级安全威胁的动态跟踪分析。

关键词: 双层模型; 传播路径; 社会工程; 网络扫描

中图分类号: TP393.08

文献标识码: A

doi: 10.11959/j.issn.2096-109x.2021015

Two-layer threat analysis model integrating macro and micro

SUN Cheng, HU Hao, YANG Yingjie, ZHANG Hongqi
Information Engineering University, Zhengzhou 450001, China

Abstract: The existing threat analysis models failed to comprehensively analyze the propagation of advanced security threats integrating the threat development trend and propagation path. In order to solve the problem, a two-layer threat analysis model named TL-TAM was established. The upper layer of the model depicted the threat development trend. The lower layer depicted the threat propagation path considering social engineering and networks can. Based on the model, prediction algorithm of threat development was proposed. The experimental result shows that the model can comprehensively analyze the threat propagation at multiple levels, overcome the defect that the threat analysis model based on attack graph is limited to technical vulnerability attack, and is more suitable for dynamic tracking analysis of advanced security threats.

Keywords: two-layer model, propagation path, social engineering, network scan

收稿日期: 2020-08-27; 修回日期: 2020-10-07

通信作者: 孙澄, suncheng1991@outlook.com

基金项目: 国家自然科学基金 (61902427)

Foundation Item: The National Natural Science Foundation of China (61902427)

论文引用格式: 孙澄, 胡浩, 杨英杰, 等. 融合宏观与微观的双层威胁分析模型[J]. 网络与信息安全学报, 2021, 7(1): 143-156.

YANG C, HU H, YANG Y J, et al. Two-layer threat analysis model integrating macro and micro[J]. Chinese Journal of Network and Information Security, 2021, 7(1): 143-156.

1 引言

作为应对网络安全威胁的有效手段,动态威胁分析技术是近年来网络安全领域的研究重点,而针对 APT 等高级安全威胁的跟踪分析既是当前的研究热点,也是难点所在。威胁的动态跟踪分析需以威胁分析模型为基础,实时融合多源告警信息进行攻击级联,实现威胁的感知^[1]及跟踪,从而为防御决策的制定提供依据^[2]。因此,提高模型刻画威胁发展传播过程的性能是提升威胁分析质量的关键。而高级安全威胁不仅在微观行为层面综合利用社会工程等手段,形成了复杂的传播路径,而且在宏观发展趋势上通过多阶段攻击逐步达成攻击目的,体现出明显的阶段递进特征,因此,要实现对高级安全威胁的动态跟踪分析,需要建立能够兼顾宏观、微观两个层面威胁传播特征的威胁分析模型,提高模型分析的全面性。

现有威胁分析模型大多只针对单个层面的威胁传播特征进行分析和刻画,按层次将其划分为宏观模型与微观模型。其中,宏观模型主要以威胁发展阶段及阶段递进关系刻画了威胁的宏观发展趋势。相关研究包括:Li 等^[3]利用统计分析法,提取 89 例 APT 攻击事件在攻击技术利用方面的特征,构建 APT 阶段模型 (APT stage model),然而,模型受样本事件局限,普适性不强;Chen 等^[4-6]基于杀伤链 (Kill-chain)^[7]这一描述网络入侵过程的一般模型,分别结合威胁特征、检测能力对阶段要素特征展开分析,通过优化一般模型,生成具有较好普适性的宏观模型。然而,该类模型的重点在于攻击技术特征的分析,忽视了阶段递进关系的提炼,无法较好地反映威胁严重程度逐步提升的发展趋势,且模型以文字描述为主,未进行形式化表示,不利于进一步应用。

微观模型主要对威胁在目标系统内的传播路径进行了刻画。由于攻击图^[8]能够结合攻击知识以及系统场景知识,实现威胁传播路径的细粒度分析,因此成为研究微观层面威胁传播的主要模型。吴迪^[9]及 Wu^[10]分别提出了基于颜色 Petri 网及本体语义推理的攻击图构建技术,为基于攻击图的威胁分析模型构建提供了有效方法;Wang

等^[11-13]提出了基于攻击图的告警信息前后件级联算法,对攻击场景进行识别跟踪,验证了攻击图在动态威胁分析领域的效用。而冷强等^[14]在属性攻击图的构建过程中增加了网络服务关联关系引起的威胁传播,使模型对威胁传播路径的刻画更加完整,并依据权限提升原则,设计了告警前件推断、后件预测等推断算法^[15],为基于攻击图的动态威胁分析提供了新的有效方法。然而,以上微观模型均未考虑社会工程攻击等非技术手段引起的威胁传播,路径完整性有待提高。

除以上不足外,两类模型单独应用于威胁跟踪分析时也存在局限性。其中,宏观模型虽然能够在全局层面直观反映威胁的发展趋势,但其结合目标系统场景能力弱,无法刻画威胁在节点间的传播细节,难以进行攻击级联。而微观模型虽然对威胁传播的细节具有较强的分析能力,但未能结合威胁的宏观发展趋势进行路径预测,容易生成大量冗余的路径。针对以上问题,本文首先利用混合原子攻击本体建模了社会工程攻击及网络扫描攻击,将两类攻击融入威胁传播路径的分析中,构建威胁微观传播层 (micro transfer layer),提高模型刻画威胁传播路径的完整性;在此基础上,从设备威胁度出发,划分威胁发展阶段,并分析威胁在不同威胁度的设备间的发展动向,构建威胁宏观发展层 (macro development layer),为结合趋势分析的威胁发展预测提供框架;最后基于双层威胁分析模型 (TL-TAM, two-layer threat analysis model),提出了威胁发展趋势预测算法,为高级安全威胁的动态跟踪分析提供支持。

2 TL-TAM 模型结构设计

双层威胁分析模型 TL-TAM 是在总结当前宏观及微观模型研究成果的基础上,针对不足进行优化改进,涉及概念较为广泛,为便于下文论述,本节首先定义相关概念,然后对模型的总体结构进行介绍。

2.1 相关概念定义

定义 1 原子攻击 a ^[16]。指单步攻击动作,威胁利用原子攻击获取权限 p ,以此作为下一步传播的条件,实现在目标系统中的持续传播。

定义 2 威胁度 (dot, degree of threat)。指

系统设备被攻陷对核心资产构成的威胁程度，设备被攻陷后产生的威胁度越高，表示威胁越容易由此为起点向核心资产进行传播。

定义 3 攻击行为框架 (af, attack framework)。指单个目标设备上的攻击场景，本文基于杀伤链进行刻画，af=(Recon, Weapon, Delivery, Exploit, Install, C&C, Obj-Action)，每个元素表示发生在该设备上的相应类型的原子攻击。

定义 4 设备访问路径 (dp, device path)。一组由业务访问关系链接而成的无环设备序列，表示为 $d_0 \xrightarrow{br_1} d_1 \xrightarrow{br_2} \dots \xrightarrow{br_n} d_n$ ，记系统中业务访问关系集为 BR， $br_i \in BR$ 表示设备 d_{i-1} 对设备 d_i 的业务访问关系， $\forall d_i, d_j \in dp, i \neq j$ ，则 $d_i \neq d_j$ 。

定义 5 威胁传播路径 (ap, attack path)。一组前后依赖的原子攻击组成的无环序列，表示为 $a_0 \rightarrow a_1 \rightarrow \dots \rightarrow a_n$ ， $\forall a_i, a_{i+1} \in ap, post(a_i) = pre(a_{i+1})$ 且 $\forall a_i, a_j \in ap, i \neq j, a_i \neq a_j$ 。定义 $pre(ap) \equiv pre(a_0)$ ， $post(ap) \equiv post(a_n)$ 。

定义 6 路径映射 $f: ap \rightarrow dp$ 。表示威胁传播路径与其依托的设备访问路径的映射关系，给定威胁传播路径 ap，可通过映射 f 提取并输出其依托的设备访问路径。其逆向映射记为 f^{-1} ，需要注意的是，同一设备访问路径上可能存在多条威胁传播路径，因此给定 dp 的情况下，通过 f^{-1} 返回的是威胁传播路径集 AP。

定义 7 威胁发展阶段 s 。指威胁在实现最终攻击目的过程中所处的层次，反映了威胁发展对核心资产安全的危害程度。每个阶段由目标设备集 D 、攻击行为框架矩阵 AF 、阶段威胁度区间

Inv、设备威胁度矩阵 DOT 、威胁传播路径矩阵 AP 以及阶段内业务访问关系集 BR_s 组成，即 $s = (D, AF, Inv, DOT, AP, BR_s)$ ，其中， AF 按攻击行为框架存储了阶段内所有目标设备上可能发生的攻击场景， $\forall AF_{i,j} \in AF$ ， $AF_{i,j}$ 为本阶段第 i 个设备上发生的第 j 类攻击行为的集合， $0 \leq i < |D|, 0 \leq j < 6$ ； DOT 存储了本阶段内各个目标设备被攻陷时对核心资产产生的威胁度值， $\forall DOT_{i,k} \in DOT$ ， $DOT_{i,k}$ 为本阶段第 i 个设备对第 k 个核心资产的威胁度，该威胁度产生所依托的威胁传播路径存储于矩阵 AP 中相应位置的元素 $AP_{i,k}$ 中， $DOT_i = \sum_k DOT_{i,k}$ 为该设备的综合威胁度；阶段威胁度区间 Inv 由用户根据需要自行确定；业务访问关系 $b \in BR_s$ 则只保留阶段内由低威胁度设备到高威胁度设备的单向关系，以刻画威胁度递增的发展轨迹。

定义 8 本体 OT。由五元组 (C, I, R, L, T) 表示，其中， C 表示类的集合， I 表示实例集合， R 表示关系的集合，包括类与类之间关系的集合 R_C ，类与实例之间的关系集合 R_{C-I} 以及实例与实例之间的关系集合 R_I ， L 表示规则集合， T 表示关系类型的集合。

2.2 模型总体结构

为兼顾高级安全威胁在宏观、微观两个层面的发展传播，TL-TAM 模型总体结构分为两个部分，如图 1 所示，上层为威胁宏观发展层，刻画威胁严重程度逐步提升的发展趋势，下层为威胁微观传播层，刻画威胁利用原子攻击产生的传播路径。模型具体说明如下。

模型整体以二元组 (Ma, Mi) 表示。其中，

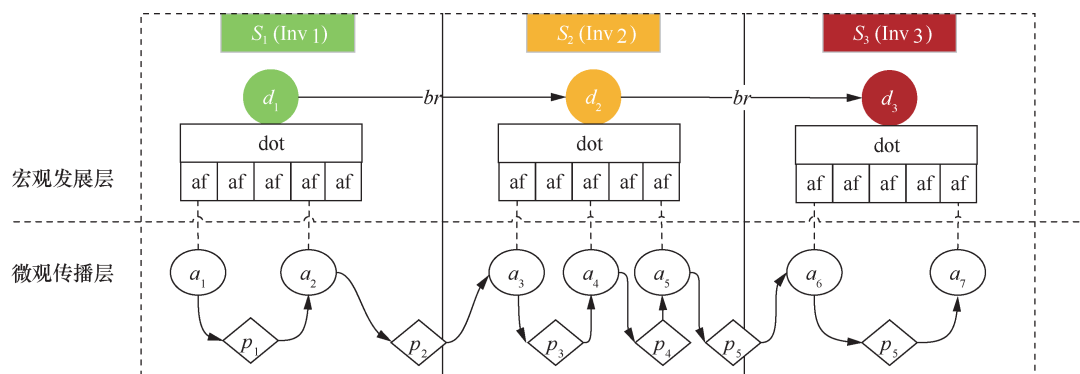


图 1 TL-TAM 模型总体结构
Figure 1 Overall structure of TL-TAM

Ma 为威胁宏观发展层,以二元组 $(S, BR_{S,S})$ 表示,其中 S 为威胁发展阶段集, $BR_{S,S}$ 为跨阶段的业务访问关系集,由低威胁度阶段的设备指向高威胁度阶段的设备,表示不同威胁发展阶段间的递进关系。Mi 为威胁微观传播层,以四元组 (P, A, E, Pr) 表示,其中 P 为权限节点集, A 为原子攻击节点集, $E = \{P \times A\} \cup \{A \times P\}$ 为有向边,表示权限节点与原子攻击节点的前后置关系, Pr 为原子攻击成功概率的集合。单个权限 p_i 及原子攻击 a_i 以 OWL 的对象属性形式表示,下缀 $i \in N$ 。函数 $pre(\cdot)$ 与 $post(\cdot)$ 分别为前置节点、后置节点提取函数。

3 威胁宏观发展层

威胁宏观发展层主要负责刻画威胁的严重程度随发展阶段递进而逐步提升的宏观发展趋势,为威胁发展现状判定及发展趋势预测提供框架。威胁的不同发展阶段本质上代表了对核心资产不同程度的威胁,当前宏观模型存在阶段递进与威胁严重程度提升趋势契合度不高的问题,其主要原因在于阶段划分未从威胁度这一本质出发,转而以攻击技术等外在特征作为依据。针对这一问题,本文提出了基于威胁度的威胁宏观发展层构建算法,通过聚合威胁度相近的目标设备构建威胁发展阶段,并利用设备间业务访问关系刻画威胁由低威胁度设备向高威胁度设备发展的轨迹。

3.1 威胁度计算

给定系统设备 d_0 , 核心资产 ass , $dp = \{d_0, d_1, \dots, d_n\}$ 为被攻陷设备 d_0 至资产所在设备 d_n 的设备访问路径,路径中无重复设备, br_i 表示设备 d_{i-1} 与设备 d_i 间的业务访问关系, $i \in [1, n]$ 。设备 d_0 被攻陷后,威胁沿设备访问路径 dp 对资产 ass 产生的威胁度计算如下。

(1) 当 $dp = \{d_0\}$ 时,表示资产所在设备被攻陷,威胁度取值如下。

$$\text{dot}_{dp}(d_0, ass) = 1$$

(2) 当 $dp \neq \{d_0\}$ 时,由于威胁本质上是通过攻击实现转移,若设备访问路径 dp 上存在威胁传播路径 ap ,则威胁度可通过多步威胁转移概率进行计算,否则,表示威胁无法通过攻击沿该访问路径转移到资产所在位置,威胁度为 0。设

$AP = f^{-1}(dp)$, 具体取值如下。

$$\text{dot}_{dp} = \begin{cases} \max_{ap \in AP} \left(\prod_{a \in ap} pr(a) \right), & AP \neq \emptyset \\ 0, & AP = \emptyset \end{cases}$$

系统设备 d_0 至核心资产 ass 存在多条设备访问路径时,记路径集为 DP ,设备 d_0 被攻陷对资产 ass 的威胁度取所有路径中威胁度最大者,即

$$\text{dot}(d_0, ass) = \max_{dp \in DP} (\text{dot}_{dp}(d_0, ass))$$

系统中存在多个核心资产时,记核心资产集合为 ASS ,则设备 d_0 综合威胁度为其对全部资产的威胁度之和,即

$$\text{dot}(d_0, ASS) = \sum_{ass \in ASS} \text{dot}(d_0, ass)$$

3.2 威胁发展层构建

基于设备威胁度,用户可根据自身需要设置合适的威胁度区间,划分威胁发展阶段,对威胁度同属一个区间的设备进行聚类,并根据威胁度提升原则,利用低威胁度设备与高威胁度设备间单向的业务访问关系刻画威胁在不同阶段、不同设备间的发展方向,具体算法如下。

算法 1 威胁宏观发展层构建算法

输入 系统设备集 D , 设备间业务访问关系集 BR , 阶段威胁度区间集 INV

输出 威胁宏观发展 $Ma = (S, BR_{S,S})$

1) Initialize $S, BR_{S,S}, ASS, AP = \emptyset$;

2) $ASS = \text{AssetIdentification}()$; //资产识别,提取核心资产集

3) Classify S according to INV //按照既定威胁度区间划分阶段

4) for each $d \in D$

5) for each $ass \in ASS$

6) $d' = \text{Asset Holder}(ass)$; //识别核心资产所在设备

7) $DP_a = \text{PathIdentification}(d, d', D, BR)$;

//提取设备到资产所在位置的访问路径

8) for each $dp \in DP_a$

9) $AP = f^{-1}(dp)$; //提取访问路径上存在的威胁传播路径

10) Calculating $\text{dot}_{dp}(d, ass)$; //计算设备沿该访问路径对该资产的威胁度

```

11)      end for
12)      dot=max(dotdp(d,ass)); //提取设备
对该资产的最大威胁度
13)      s.DOT←dot;
14)      s.AP ←argmax(dotdp: ap ∈ AP, dp=
f(ap));
15)  end for
16)  dot(d, ASS)=  $\sum_{ass \in ASS} dot(d, ass)$ ; //计算设
备的综合威胁度
17)  for each s ∈ S
18)      if dot(d,ASS) ∈ s.Inv
19)          s.D←d; //将设备聚类到对应威
胁度区间的阶段中
20)          Break;
21)      end if
22)  end for
23) end for
24) for each br ∈ BR
25)  if dot(br.source,ASS) ≤ dot (br.destination,
ASS)&&s(br.source)=s(br.destination)
26)      s(br.source).BRS←br; //刻画阶段内
部威胁发展轨迹
27)  else if dot(br.source,ASS) ≤ dot (br.destination,
ASS)&&s(br.source)≠s(br.destination)
28)      BRS-S←br; //刻画阶段间威胁发展
轨迹
29)  end if
30) end for
31) Return Ma=(S, RS-S)

```

以上算法的核心为步骤 4)~步骤 23), 通过计算获取了系统中每个设备被攻陷后的综合威胁度。基于设备的威胁度及用户既定的阶段威胁度区间, 步骤 17)~步骤 22)将目标设备聚合到相应威胁发展阶段。最终, 基于威胁度提升原则, 步骤 24)~步骤 31)对设备间存在的业务访问关系进

行了过滤, 保留低威胁度设备指向高威胁度设备的业务访问关系, 从而刻画出威胁在不同设备、阶段间的发展趋势。

4 威胁微观传播层

威胁微观传播层负责刻画威胁利用原子攻击进行传播的路径。当前攻击图模型应用于分析高级安全威胁的传播路径时存在局限性, 即攻击模型单一, 无法分析社会工程攻击、网络扫描攻击等非技术漏洞攻击产生的威胁传播路径。针对这一问题, 本文利用本体论, 以技术漏洞利用模型为参照, 对社会工程攻击以及网络扫描攻击进行了细粒度分析, 构建囊括这 3 类攻击的混合原子攻击本体, 并通过本体推理构建威胁微观传播层, 提高刻画威胁传播路径的完备性。

4.1 本体建模

本体是概念及概念间关系的抽象描述, 当前网络安全领域的相关研究常通过构建原子攻击本体对原子攻击进行细粒度分析及建模。混合原子攻击本体相较于常见的原子攻击本体最大的区别在于增加了对社会工程攻击及网络扫描攻击的本体建模。以下首先对这两类攻击进行要素分析, 再对混合原子攻击本体进行介绍。

本文主要从攻击、脆弱性、脆弱性所在部位(组件)及攻击前提条件等方面对社会工程攻击及网络扫描展开分析, 结果如表 1 所示。

由表 1 可知, 社会工程攻击本质是利用人性弱点实施的攻击, 主要包括鱼叉(spear phishing)攻击、水坑(Waterholing)攻击等攻击技术。此类攻击前提是攻击者需要与受害者建立某种信任关系, 如控制受害者经常访问的网站、窃取有业务往来的邮箱账号等, 利用信任关系, 触发受害者好奇、轻信等人性弱点, 诱使其点击恶意链接、下载恶意邮件, 从而借助受害者的操作管理权限对相关设备实施攻击。

表 1 社会工程攻击及网络扫描攻击分析
Table 1 Analysis of social engineering attack and network scan attack

攻击	攻击子类	脆弱性	脆弱性存在部位(组件)	前提条件
社会工程攻击	鱼叉攻击、水坑攻击	人性弱点	操作人员	获取信任关系
网络扫描	服务扫描、端口扫描、漏洞扫描	—	—	控制扫描发起设备

网络扫描本质上是一种侦察行为, 主要通过利用一系列脚本远程检测网络或网络设备的某些状态, 如网络中存在的 IP, 设备开放的服务、端口等, 未实质性地破坏系统安全规则。本文根据扫描对象, 主要考虑两类扫描攻击, 即服务扫描、业务扫描。其中, 服务扫描是一种常用的扫描攻击, 用于探测目标节点可远程访问的服务; 业务扫描并不特指某种具体的扫描攻击, 而是复合多种扫描技术的抽象概念, 表示以业务关系导致的节点间特权为探测对象的扫描攻击, 如管理站对工作站的管理权限、工作站在服务器上的用户权限等。

根据以上分析, 构建混合原子攻击本体 HAOT, 如图 2 所示。HAOT 可表示为 $(C_H, I_H, R_H, L_H, T_H)$, 其中, 类集 C_H 由设备 (device)、组件 (component)、脆弱性 (vulnerability)、攻击 (attack) 4 个基础类及其子类组成。关系类型集 T_H 主要包括类间包含关系 subclass of、类与实例间的从属关系 type of、实例间关系以 OWL (web ontology language) 本体描述语言定义的对象属性格式 property(C_1, C_2) 描述, C_1 为对象属性的定义域,

C_2 为值域, 具体如表 2 所示。关系集 R_H 中, 类间包含关系如图 2 所示, 类与实例从属关系、实例间关系及实例集 I_A 需结合具体场景进行提取。规则集 L_H 以 SWRL (semantic Web rule language) 构建了技术漏洞利用、社会工程攻击以及网络扫描攻击等 3 类攻击行为模型, 以下分别举例说明。

例 1 以利用“CVE-2017-0290”漏洞发起的远程代码执行攻击为例, 其模型如下所示。

$\text{Device}(?a) \wedge \text{Software}(?b) \wedge \text{Vulnerability}(?c) \wedge \text{has} (?a, ?b) \wedge \text{exist} (?b, ?c) \wedge \text{sameAs} (?c, \text{CVE-2017-0290}) \wedge \text{AttackCodeExecution} (?d) \wedge \text{exploit} (?d, ?c) \wedge \text{Adversary} (?e) \wedge \text{hasAccess} (?e, ?b)$

$\rightarrow \text{launchAttack_CodeExecution_CVE-2017-0290} (?e, ?a) \wedge \text{hasCompromised} (?e, ?b) \wedge \text{hasPrivilegeRoot} (?e, ?a)$

例 2 以“hw-001”表示“轻信”的人性弱点实例, 利用该脆弱性发起的鱼叉攻击的模型如下。

$\text{Device} (?a) \wedge \text{Operator} (?b) \wedge \text{Vulnerability} (?c) \wedge \text{has} (?a, ?b) \wedge \text{exist} (?b, ?c) \wedge \text{sameAs} (?c, \text{hw-001}) \wedge \text{AttackPhishing} (?d) \wedge \text{exploit} (?d, ?c) \wedge \text{Adversary}$

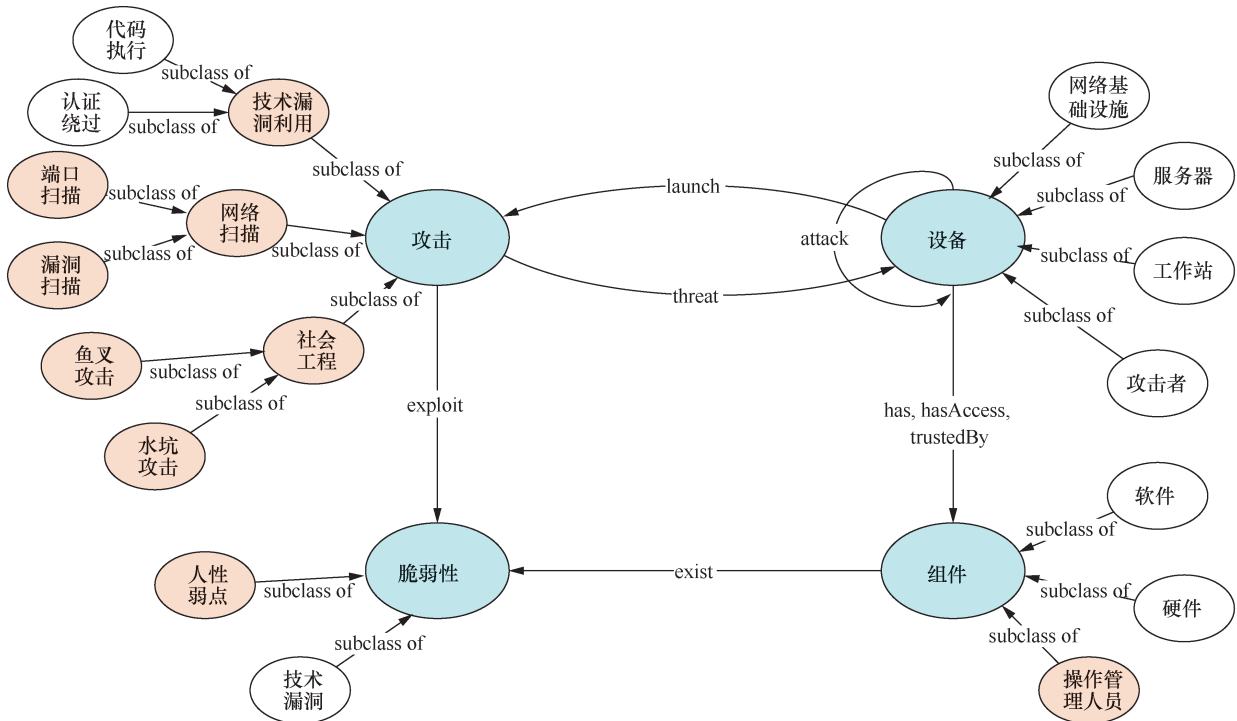


图 2 混合原子攻击本体示意
Figure 2 Hybrid atomic attack ontology

$(?e) \wedge \text{trustedBy}(?e, ?b)$
 $\rightarrow \text{launchAttack_Phishing_hw-001}(?e, ?a)$
 $\wedge \text{hasCompromised}(?e, ?b) \wedge \text{hasPrivilegeUser}(?e, ?a)$

例 3 在完全控制扫描发起设备的条件下发起服务扫描, 探测开放服务的模型如下所示, 由于不同的扫描发起源获取的结果可能不同, 因此 launchAttack 的首个参数为源设备实例以示区分。

$\text{Device}(?a) \wedge \text{Service}(?b) \wedge \text{Device}(?c) \wedge$
 $\text{has}(?a, ?b) \wedge \text{hasAccess}(?c, ?b) \wedge \text{AttackServiceScan}(?d) \wedge \text{Adversary}(?e) \wedge \text{hasPrivilegeRoot}(?e, ?c)$
 $\rightarrow \text{launchAttack_ServiceScan_null}(?c, ?a) \wedge \text{hasAccess}(?e, ?b)$

以上规则模型的条件部分称为规则体 (body), 结果部分称为规则头 (head)。

4.2 攻击成功概率计算

攻击成功概率的量化是威胁度计算的基础, 本体建模包括技术漏洞利用、社会工程攻击以及网络扫描攻击 3 种攻击行为, 下面针对这 3 类攻击的发生概率分别进行量化计算。

(1) 技术漏洞攻击

此类攻击的成功率常用的量化计算方式是基于漏洞的 CVSS 评分, 综合漏洞可用性、影响等因素进行量化。CVSS3.1 评分标准中漏洞的基本得分公式如下。

$$\text{Base} = \begin{cases} 0, & \text{Impact} \leq 0 \\ \text{Roundup}(\text{Min}[(\text{Impact} + \text{Exp}), 10]), & \text{其他} \end{cases}$$

函数 Roundup () 对输入数值向上舍入保留至小数点后一位, Impact、Exp 分别为漏洞的影响得分及漏洞的可用性得分, 计算公式如下。

$$\begin{cases} \text{Impact} = 6.42 \cdot [1 - (1 - C) \cdot (1 - I) \cdot (1 - A)] \\ \text{Exp} = 8.22 \cdot \text{AV} \cdot \text{AC} \cdot \text{PR} \cdot \text{UI} \end{cases}$$

由于高级安全威胁通常将攻击隐蔽性作为追求, 而漏洞利用的代码成熟度越高, 越能够支持或发展出各种代码隐藏技术, 从而满足攻击隐蔽性需求, 因此将漏洞评分公式优化如下。

ModifiedScore=

Roundup(Base · ExploitCodeMaturity)

漏洞评分越高, 利用该漏洞的攻击成功概率越高, 由于评分值域为 [0, 10], 因此技术漏洞攻击成功概率计算公式给出如下。

$$\text{pr} = 0.1 \cdot \text{ModifiedScore}$$

(2) 其他攻击

社会工程攻击及网络扫描攻击均以技术漏洞利用攻击作为参照, 进行量化。其中, 社会工程攻击利用人性弱点发起, 相较于技术漏洞, 人性弱点需通过广泛的背景情报分析进行挖掘, 且触发及影响具有较大的不确定性, 因而成功率相对较低, 对标低危技术漏洞得分, pr

表 2 混合原子攻击本体主要对象属性
Table 2 Main object properties of HAOT

对象属性名	定义域	值域	含义
has	Device	Component	设备实例 Indv _{Dev} 拥有组件实例 Indv _{Cmpt}
hasAccess	Device	Component	设备实例 Indv _{Dev} 拥有组件实例 Indv _{Cmpt} 的访问权限
exist	Component	Vulnerability	组件实例 Indv _{Cmpt} 存在脆弱性实例 Indv _{Optr}
exploit	Attack	Vulnerability	攻击实例 Indv _{Att} 利用脆弱性实例 Indv _{Optr}
hasPrivilegeUser	Device	Device	设备实例 Indv _{Dev} 拥有组件实例 Indv _{Cmpt} 的用户权限
hasPrivilegeRoot	Device	Device	设备实例 Indv _{Dev} 拥有组件实例 Indv _{Cmpt} 的完全控制权限
hasCompromised	Device	Component	设备实例 Indv _{Dev} 损害了组件实例 Indv _{Cmpt}
trustedBy	Device	Operator	设备实例 Indv _{Dev} 受操作管理人员实例 Indv _{Optr} 信任
launchAttack_{name}_{vul}	Device	Device	设备实例 Indv _{Dev} 向组件实例 Indv _{Cmpt} 发起攻击

注: “launch Attack”命名格式中, name 为攻击名称, vul 为脆弱性, 技术漏洞以 cve 编号格式进行标识, 人员弱点采取自编号模式, 以“hw_code”格式标识, 当攻击属于网络扫描类时, vul 以“null”标识。

设置为 0.4；而网络扫描攻击不依赖漏洞发起，pr 设置为 1。

4.3 传播层构建

利用混合原子攻击本体，笔者进一步提出微观传播层的构建方法，框架如图 3 所示。

第一步，信息获取，主要获取目标系统场景知识及攻击知识这两方面信息，场景知识包括拓扑结构、业务关系、节点软硬件配置、操作管理人员以及存在的技术漏洞、人员弱点评估数据等；攻击知识包括攻击模式、难度以及攻击影响等，可通过检索 NVD 漏洞库以及 CAPEC 攻击模式库等公共数据库以及社会工程及网络扫描相关研究成果获取。

第二步，知识库构建，该部分利用混合原子攻击本体定义的类及关系类型，从搜集的信息中，提取实例及关系，并生成推理规则，从而补全本体知识，构建完整的本体知识库。

第三步，Mi 生成，将知识库与既设初始条件作为输入，利用语义推理引擎进行攻击推理，并通过可视化处理，生成模型 TL-TAM 的威胁微观传播层。具体生成算法如算法 2 所示。

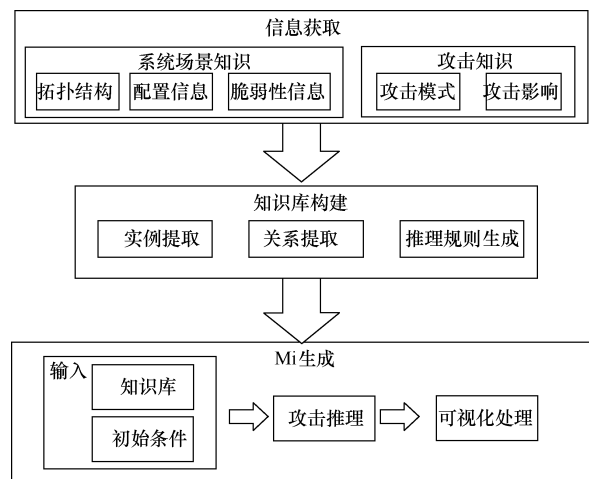


图 3 基于混合原子攻击本体的微观传播层构建框架
Figure 3 HAOT based Mi construction framework

算法 2 威胁微观传播层生成算法

输入 混合原子攻击本体 HAOT；攻击者实例 att；初始条件集 P_0

输出 威胁微观传播层 $Mi=(P, A, E, Pr)$

- 1) Initialize $P, A, E, S, Pr = \emptyset$;
- 2) AttackReason(att, P_0 , HAOT); //利

用推理引擎执行多步攻击连续推理

3) $S = \text{inferredFacts}()$; //提取推理结果

4) $A = \text{getAttacksProperties}(S)$; //提取原子攻击集

5) $P = P_0 \cup \text{getPrivilegesProperties}(S)$; //提取权限集

6) for each $a \in A$

7) Calculatingpr(a); //计算原子攻击成功概率

8) $Pr \leftarrow \text{pr}(a)$;

9) $r = \text{getRules}(a)$;

10) $P_{pre} = \text{Mapping}(\text{Explanation}(a), r.body)$; //识别原子攻击前置权限

11) $P_{post} = \text{Mapping}(P, r.head)$; //识别原子攻击后置权限

12) for each $p \in P_{pre}$

13) $E = \text{CreateDirectedEdge}(p, a) \cup E$; //创建有向边

14) end for

15) for each $p \in P_{post}$

16) $E = \text{CreateDirectedEdge}(a, p) \cup E$; //创建有向边

17) end for

18) end for

19) Return $Mi = (P, A, E, Pr)$

以上算法核心为步骤 2)~步骤 5)，利用推理引擎对系统中可能发生的原子攻击进行推理并从推理结果中提取原子攻击集及权限集；在此基础上，步骤 6)~步骤 18)，识别原子攻击与权限的前后置关系并构建有向边，其中，原子攻击的前置权限通过匹配推理规则体 (body) 与引擎提供的推理解释 (explanation) 进行确定，而后置权限则通过匹配推理规则头 (head) 与权限集进行提取。需要注意的是，近年来 Hermit 等本体推理引擎能够自动将推理获取的事实作为新的推理条件执行多步连续推理，因此不再需要依靠逐轮更新攻击条件的形式实现推理进程的持续推进。并且推理引擎为每条推理获取的事实提供了详细的推理解释，极大地方便了节点前后置关系的识别。

以上算法的时间开销主要由攻击推理造成，而推理过程的时间复杂度取决于推理引擎性能，与目标系统规模相关，假设系统节点总数为 N ，推理引擎性能以函数 $R(N^k)$ 表示， $k=1,2,\dots$ ，则算法时间复杂度为 $O(R(N^k))$ 。

5 威胁发展预测

威胁发展趋势预测以威胁微观传播层为基础，威胁宏观发展层为框架，以被攻陷的目标设备为输入，判定当前威胁发展状态并根据相邻目标设备威胁度，预测下一步可能选取的目标设备提取相应威胁传播路径，算法如下。

算法 3 威胁发展预测算法

输入 被攻陷设备 d ；双层威胁分析模型 TL-TAM=(Ma, Mi)

输出 威胁发展状态 s ；下一步目标设备 d' ；威胁传播路径集 AP

1) Initialize $s=\text{null}$, $d'=\text{null}$, $D_c=\emptyset$, $dp=\emptyset$, $ap=\emptyset$;

2) $s=s(d)$;

3) $D_c = \text{NeighborHost}(d, \text{Ma})$;

4) $d' = \text{argmax}(\text{dot}(d): d \in D_c)$; //以威胁度最高的邻接设备作为下一步攻击目标

5) $\text{AP} \leftarrow s.\text{AP}_{d'}$; //提取由设备 d' 指向各位置核心资产的威胁传播路径

6) $dp \leftarrow d, d'$;

7) $ap = f^{-1}(dp)$; //提取由设备 d 发起，设备 d' 上的攻击场景

8) for each $ap' \in \text{AP}$

9) if $\text{post}(ap) < \text{pre}(ap')$ //判断设备 d' 上的攻击场景获取的权限能否支撑威胁进一步传播

10) create UnknownAttack^0 ; //创建未知攻击

11) $\text{pre}(a^0) = \text{post}(ap)$; $\text{post}(a^0) = \text{pre}(ap')$;

12) $ap' = (ap \rightarrow a^0 \rightarrow ap')$; //链接威胁传播路径

13) else

14) $ap' = (ap \rightarrow ap')$;

15) end if

16) end for

17) Return s , d' , AP

上述步骤 9)~步骤 15) 在威胁传播路径预测中考虑了未知攻击的影响，当被攻陷设备与被选取作为下一步攻击目标的设备间的攻击路径 ap 获取的权限不能支持由被选取设备指向核心资产的威胁传播路径 ap' 时，预测两段路径间存在未知攻击，未知攻击的前置权限为 $\text{post}(ap)$ ，后置权限为 $\text{pre}(ap')$ 。

6 实验

6.1 实验环境

为验证模型性能，以典型的企业信息系统作为实验场景设计实验，实验环境如图 4 所示。该信息系统由 4 个子网组成，每个子网均部署防火墙及入侵检测系统以实现访问控制及入侵检测。其中，DMZ 区域部署 Web 服务器及邮件服务器，可提供对外服务；子网 1 部署 2 台工作站及 1 台文件服务器，其中，工作站 1 拥有文件服务器用户账号，文件服务器存储了企业关键办公信息，防火墙仅允许工作站访问邮件服务器；子网 2 部署数据服务器，为 Web 服务器提供数据服务，存储重要业务数据，防火墙仅允许 Web 服务器与管理站访问数据服务器；子网 3 部署管理站，负责数据服务器管理，具有控制能力。系统中脆弱性包括技术漏洞及操作管理人员存在的弱点，具体如表 4 所示。

6.2 模型构建

实验使用 Protege5.5.0 构建混合原子攻击本体，依据该本体定义的类，从场景知识及攻击知识中提取实例，如表 3 所示。依据关系类型，提取实例间关系如表 4 所示，表 4 中仅展示业务访问关系，其余关系通过表 3 中列间对应关系进行描述。为使表达简便，对象属性中实例均以简称代替，如 Web Server 1 写作 web_1。构建攻击者实例 $\text{type of (Adversary, att)}$ ，初始条件设置如表 5 所示。

Protege5.5.0 集成了 HermiT1.4.3 推理引擎，可支持多步连续推理。首先，在 SWRLTab 中，为脆弱性实例以及服务扫描、业务关系扫描等两类扫描攻击构建本体推理规则，然后，使用推理引擎执行攻击推理，提取原子攻击集 A 、权限集 P 并通过解析 HermiT1.4.3 提供的推理解释对攻

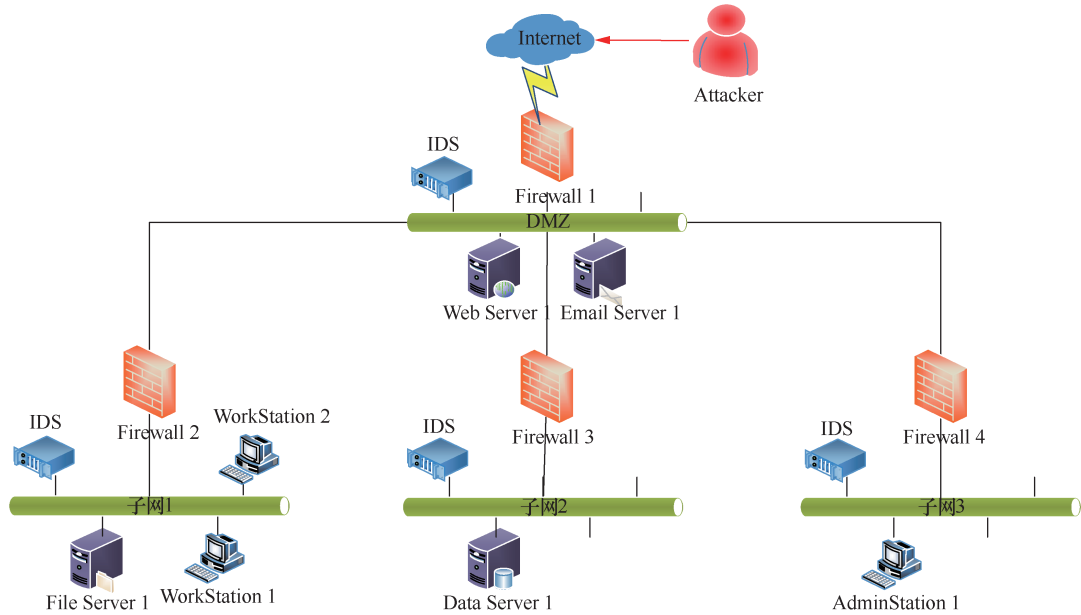


图 4 实验环境
Figure 4 Experiment environment

表 3 实例提取
Table 3 Extracted individuals

Device	Component	Vulnerability	ModifiedScore	Attack
Firewall 4	Management Software(MS)	CVE-2019-1642 (恶意脚本注入)	6.1	ScriptInjection
Admin Station 1	Google Chrome	CVE-2018-6116 (越界访问内存)	6.5	CodeExecution
Web Server 1	Apache Tomcat Native(ATN)	CVE-2018-8019 (过期证书认证)	7.4	CertAbuse
Email Server 1	Apache James JMX(JMX)	CVE-2017-12628 (本地权限提升)	7.8	PriElevation
	Apache James Server(AJS)	CVE-2015-7611 (操作系统命令注入)	8.1	CommandInjection
WorkStation 1	Microsoft outlook	CVE-2013-3870 (远程权限获取)	9.1	CodeExecution
	Operator	hw-001 (安全意识薄弱)	4.0	Waterholing
WorkStation 2	MsMpEng	CVE-2017-0290 (远程权限获取)	9.3	CodeExecution
File Server 1	Microsoft Windows	CVE-2019-0543 (本地权限提升)	7.8	PriElevation
Data Server 1	S-CMS	CVE-2019-6805 (SQL 命令注入)	9.8	SQLInjection

表 4 业务访问关系提取
Table 4 Extracted relations

From Individual	To Individual	Object Properties
Web Server	S-CMS	hasAccess(web_1,DS_1_S-CMS)
	Management Software(MS)	hasAccess(web_1,FW_4_MS)
Email Server	Operator	trustedBy(ES_1,WS_1_operator)
	Management Software(MS)	hasAccess(ES_1,FW_4_MS)
WorkStation 1	File Server	hasPrivilegeUser(WS_1,FS_1)
	WorkStation 2	hasAccess(WS_1,WS_2_MsMpEng)
WorkStation 2	File Server	hasAccess(WS_2,FS_1_windows)
	WorkStation 1	hasPrivilegeUser(WS_2,WS_1)
Admin Station	Data Server	hasPrivilegeRoot(AS_1,DS_1)
Firewall 4	Admin Station	hasAccess(FW_4,AS_1_chrome)

表 5 初始条件
Table 5 Initial conditions

From Individual	To Individual	Object Properties
att	Apache Tomcat Native(ATN)	hasAccess(att, web_1_ATN)
	Apache James Server(AJS)	hasAccess(att, ES_1_AJS)

击动作与权限的前后置关系进行识别，构建有向边集 E ，最后，计算单步攻击成功概率。以漏洞 CVE-2018-8019 为例对本体推理过程进行说明，首先本体推理规则构建如下。

$\text{Device}(?a) \wedge \text{Software}(?b) \wedge \text{Vulnerability}(?c) \wedge \text{has}(?a, ?b) \wedge \text{exist}(?b, ?c) \wedge \text{sameAs}(?c, \text{CVE-2018-8019}) \wedge \text{AttackCert Abuse}(?d) \wedge \text{exploit}(?d, ?c) \wedge \text{Adversary}(?e) \wedge \text{hasAccess}(?e, ?b)$

$\rightarrow \text{launch_Attack_CertAbuse_CVE-2018-8019}(?e, ?a) \wedge \text{hasCompromised}(?e, ?b) \wedge \text{hasPrivilegeRoot}(?e, ?a)$

执行推理，攻击者实例 att 新增攻击关系“launchAttack_CertAbuse_CVE-2018-8019 (att, web_1)”，表明攻击发生，该关系的推理解释如图 5 所示，红线标注为推理使用的本体推理规则，规则中绿框标注为前置权限模式，根据模式中权限类型及参数检索匹配推理解释条目中符合该模式的权限实例，图中条目 8 符合模式要求，提取为该攻击的前置权限，后置权限以同样的方式识别权限模式后在攻击者实例 att 新增的权限实例中检索匹配。根据 4.2 节计算方法，CVE-2018-8019 漏洞评分为 7.4，则该攻击成功概率计算得 0.74。

微观传播层 $M_i = (P, A, E, Pr)$ 构建如图 6 所示，其中 $P = \{p_i\}$ ， $A = \{a_j\}$ ， $i, j \in N$ 且 $i \in [0, 17]$ ， $j \in [0, 18]$ ，其中，蓝、绿、棕色分别为技术漏洞

利用、社会工程攻击及网络扫描攻击产生的威胁传播路径。

以数据服务器的业务数据及文件服务器的机密文件作为实验系统核心资产，利用算法 1，基于威胁传播路径、计算系统设备的威胁度，构建威胁发展层如图 7 所示。图中圆形为目标设备，连接线为业务访问关系，各设备的威胁度以及攻击行为框架如矩形框中所示，实连接线刻画了威胁由低威胁度设备、阶段向高威胁度发展的轨迹。假设企业规定威胁度超过 1 时为高危状态，0.7 至 1 为中危状态，不超过 0.7 为低危状态，分别以绿、黄、红三色标注低、中、高 3 种威胁发展阶段。

6.3 实验分析

当系统告警显示某个设备被攻陷时，利用算法 3 可判定威胁的当前发展状态并预测威胁的下一步攻击目标及相应传播路径。表 6 展示了按威胁提升原则，不同设备被攻陷情况下的威胁发展趋势预测结果。

其中，工作站 2 (WS_2) 被攻陷时，与其存在业务访问关系的邻接设备中，文件服务器威胁度最高，然而攻陷工作站 2 后获取的文件服务器访问权限无可利用的漏洞，考虑到高级安全威胁可能具备未知攻击能力，因此，预测工作站 2 与文件服务器间存在未知攻击 a_0 : launchAttack_UnknownAttack_0day(att, FS_1)，攻击的前后置权限分别为 $\text{pre}(a_0) = \text{hasAccess}(att, FS_1)$ ， $\text{post}(a_0) =$



图 5 前置权限识别示意

Figure 5 Identification of pre-privileges to atomic attack

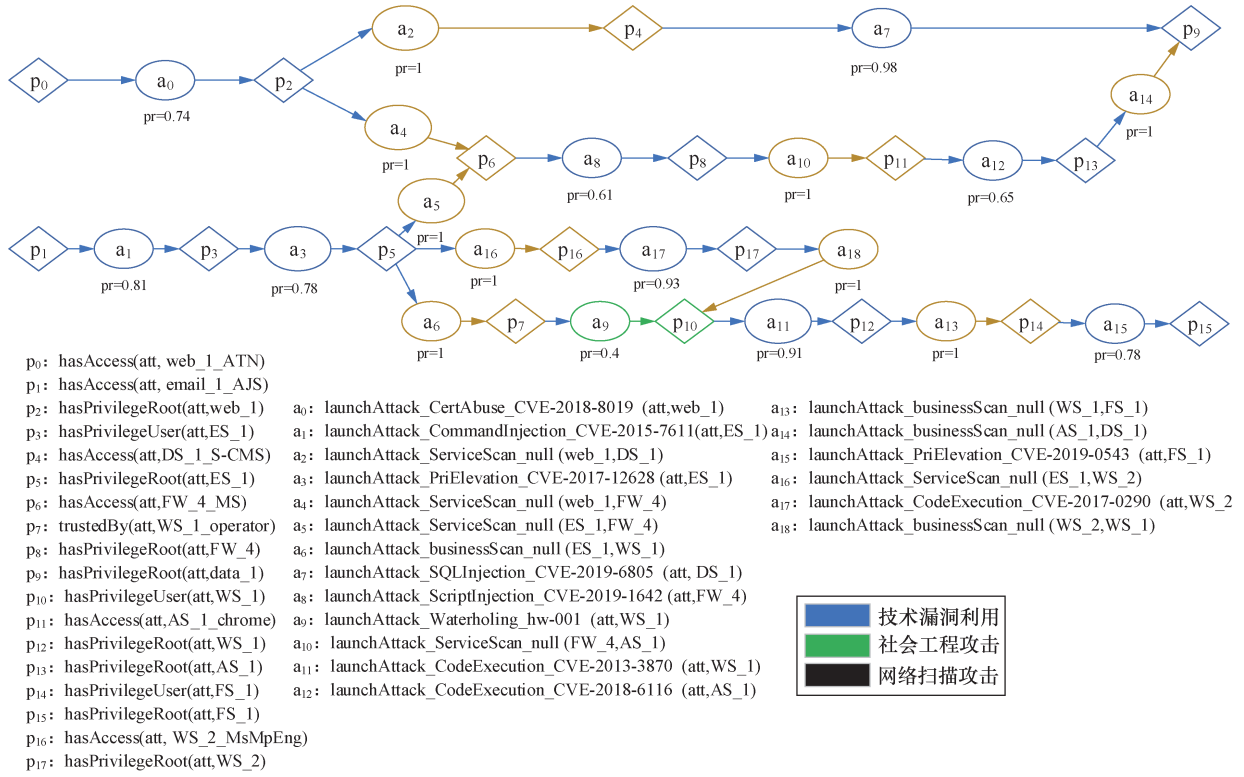


图 6 实验环境威胁传播路径
Figure 6 Micro trans ferlayer constructed by experiment

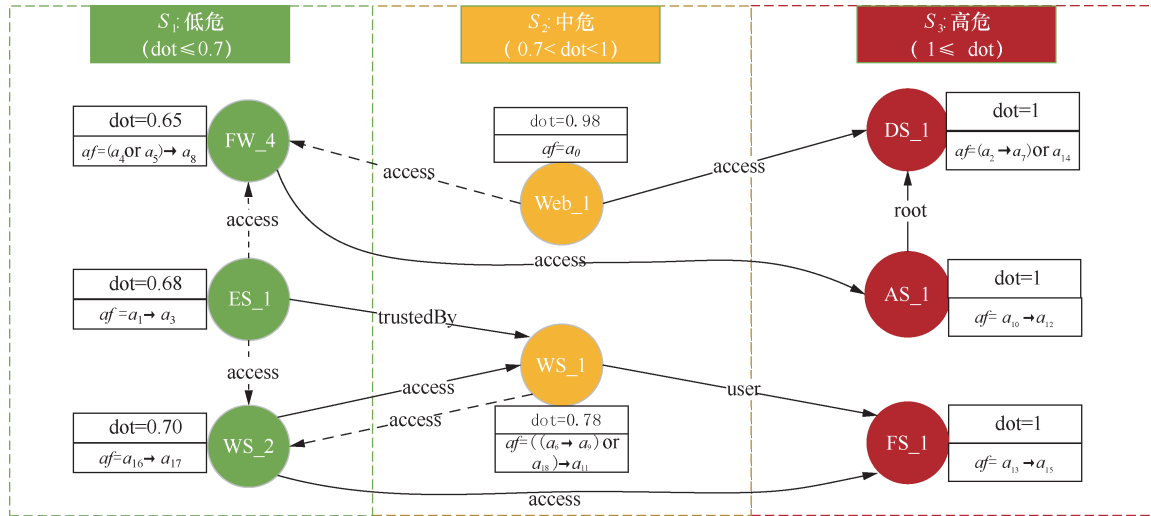


图 7 威胁宏观发展层示意
Figure 7 Sample of threat development prediction

hasPrivilegeRoot(att, FS_1)。

当邮件服务器 (ES_1) 被攻陷时, 威胁向下一步目标设备工作站 1 (WS_1) 进行转移时, 涉及社会工程攻击, 需要使用水坑攻击 a_9 。

6.4 对比分析

为验证 TL-TAM 模型分析刻画威胁发展传播的

性能, 以下将本模型分别与 Chen 等^[4]、冷强等^[14]以及樊雷等^[17]提出的模型进行对比, 结果如表 7 所示。

其中, Chen 等^[4]提出的 APT 宏观分析模型基于杀伤链划分阶段, 模型的各阶段可反复出现在针对不同目标节点的攻击过程中, 因而无法有效刻画威胁由低到高的发展趋势。而 TL-TAM 模型

表 6 威胁发展趋势预测
Table 6 Prediction of threat development trend

被攻陷设备 d	威胁发展现状 s	下步目标 d'	威胁传播路径 ap'	目标资产预测
web_1	中危	DS_1	$a_2 \rightarrow a_7$	数据服务器的业务数据
ES_1	低危	WS_1	$a_6 \rightarrow a_9 \rightarrow a_{11} \rightarrow a_{13} \rightarrow a_{15}$	文件服务器的机密文件
FW_4	低危	AS_1	$a_{10} \rightarrow a_{12} \rightarrow a_{14}$	数据服务器的业务数据
AS_1	高危	DS_1	a_{14}	数据服务器的业务数据
DS_1	高危	—	—	数据服务器的业务数据
WS_1	中危	FS_1	$a_{13} \rightarrow a_{15}$	文件服务器的机密文件
WS_2	低危	FS_1	a_0	文件服务器的机密文件
FS_1	高危	—	—	文件服务器的机密文件

表 7 对比分析
Table 7 Comparison analysis

模型	发展趋势分析	网络扫描攻击	社会工程攻击	未知攻击	路径预测
文献[4]	×	×	×	×	×
文献[14]	—	×	×	×	√
文献[17]	√	×	×	×	×
本文	√	√	√	√	√

的威胁宏观发展层以威胁度为阶段划分的依据，能够直观反映相关设备被攻陷时威胁的发展现状，并刻画了威胁由低威胁度设备向高威胁度设备的发展动向，能够有效支持结合发展趋势的威胁传播路径预测。

冷强等^[14]提出的基于攻击图的动态威胁分析模型，虽然对网络服务关系产生的威胁传播进行了分析，但未涉及社会工程攻击及网络扫描攻击。以本文实验环境中的传播路径 $a_1 \rightarrow a_3 \rightarrow a_6 \rightarrow a_9 \rightarrow a_{11} \rightarrow a_{13} \rightarrow a_{15}$ 为例，利用文献[14]方法生成结果为 $a_1 \rightarrow a_3$ ，路径出现缺失，由此可见，TL-TAM 模型的威胁微观传播层对威胁传播路径刻画准确度更高。

分层式模型 APT-HRM^[17]虽然同样利用分层式结构分析刻画了威胁在不同层次的发展特征，但其底层利用攻击树组合攻击方案的形式描述攻击路径，无法结合具体系统场景实现攻击预测。而 TL-TAM 模型不仅能够支持攻击预测，并且能够对未知攻击进行一定程度的预测分析。

7 结束语

本文针对现有威胁分析模型无法同时兼顾 APT 等高级安全威胁的宏观发展趋势及微观传播

路径，导致无法有效适用于动态威胁跟踪分析的问题，提出了一种双层威胁分析模型 TL-TAM。模型首先基于混合原子本体，建模了社会工程攻击、网络扫描攻击等以往攻击图模型欠缺的攻击行为，并利用本体推理引擎实现攻击的自动推理，构建威胁微观传播层，提高了威胁传播路径刻画的完备性。在此基础上，设计了设备威胁度计算方法，基于设备威胁度构建威胁宏观发展层，通过划分威胁发展阶段直观展示不同设备被攻陷时威胁的发展状态，并通过由低到高链接不同威胁度的设备，刻画威胁在设备间的发展走向，为结合趋势分析的威胁发展预测提供了框架。最后，本文提出了基于双层模型的威胁发展预测算法，结合威胁在设备层面的发展动向及在攻击行为层面的传播路径，对给定系统设备被攻陷后，威胁可能选取的目标设备、传播路径以及目标资产进行了预测分析。

下一步，将在持续做好模型优化的基础上，基于双层威胁分析模型，对高级安全威胁的动态跟踪分析方法展开更深入的研究。

参考文献：

- [1] 赵志岩, 纪小默. 智能化网络安全威胁感知融合模型研究[J]. 信息安全, 2020, 20(4): 87-93.

- ZHAO Z Y, JI X M. Research on the intelligent fusion model of network security situation awareness[J]. Netinfo Security, 2020, 20(4): 87-93.
- [2] 金辉, 张红旗, 张传富, 等. 复杂网络中基于 QRD 的主动防御决策方法研究[J]. 信息网络安全, 2020, 20(5): 72-82.
- JIN H, ZHANG H Q, ZHANG C F, et al. Research on active defense decision-making method based on QRD in complex network[J]. Netinfo Security, 2020, 20(5): 72-82.
- [3] LI M, HUANG W, WANG Y, et al. The study of APT attack stage model[C]//2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS). 2016.
- [4] CHEN P, DESMET L, HUYGENS C. A study on advanced persistent threats[C]//IFIP International Conference on Communications and Multimedia Security. 2014: 63-72.
- [5] USSATH M, JAEGER D, CHENG F, et al. Advanced persistent threats: behind the scenes[C]//2016 Annual Conference on Information Science and Systems (CISS). 2016: 181-186.
- [6] 贺诗洁, 黄文培. APT 攻击详解与检测技术[J]. 计算机应用, 2018, 38(S2): 170-173.
- HE S J, HUANG W P. APT attacks details and detection technology[J]. Journal of Computer Applications, 2018, 38(S2): 170-173.
- [7] HUTCHINS E M, CLOPPERT M J, AMIN R M. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains[J]. Leading Issues in Information Warfare & Security Research, 2011, 1(1): 80.
- [8] SWILER L P, PHILLIPS C. A graph-based system for network-vulnerability analysis[R]. 1998.
- [9] 吴迪, 连一峰, 陈恺, 等. 一种基于攻击图的安全威胁识别和分析方法[J]. 计算机学报, 2012, 35(9): 1938-1950.
- WU D, LIAN Y F, CHEN K, et al. A security threats identification and analysis method based on attack graph[J]. Chinese Journal of Computers, 2012, 35(9): 1938-1950.
- [10] WU S, ZHANG Y, CAO W, et al. Network security assessment using a semantic reasoning and graph based approach[J]. Computers & Electrical Engineering, 2017: 96-109.
- [11] WANG L, LIU A, JAJODIA S. Using attack graphs for correlating, hypothesizing, and predicting intrusion alerts[J]. Computer communications, 2006, 29(15): 2917-2933.
- [12] AHAMADINEJAD S H, JALILI S, ABADI M. A hybrid model for correlating alerts of known and unknown attack scenarios and updating attack graphs[J]. Computer Networks, 2011, 55(9): 2221-2240.
- [13] 刘威歆, 郑康锋, 武斌, 等. 基于攻击图的多源告警关联分析方法[J]. 通信学报, 2015, 36(9): 135-144.
- LIU W X, ZHENG K F, WU B, et al. Alert processing based on attack graph and multi-source analyzing[J]. Journal on Communications, 2015, 36(9): 135-144.
- [14] 杨英杰, 冷强, 常德显, 等. 基于属性攻击图的网络动态威胁分析技术研究[J]. 电子与信息学报, 2019, 41(8): 1838-1846.
- YANG Y J, LENG Q, CHANG D X, et al. Research on network dynamic threat analysis technology based on attribute attack graph[J]. Technology Based on Attribute Attack Graph Journal of Electronics & Information Technology, 2019, 41(8): 1838-1846.
- [15] 杨英杰, 冷强, 潘瑞萱, 等. 基于属性攻击图的动态威胁跟踪与量化分析技术研究[J]. 电子与信息学报, 2019, 41(9): 2172-2179.
- YANG Y J, LENG Q, PAN R X, et al. Research on dynamic threat tracking and quantitative analysis[J]. Technology Based on Attribute Attack Graph Journal of Electronics & Information Technology, 2019, 41(9): 2172-2179.
- [16] 胡浩, 叶润国, 张红旗, 等. 基于攻击预测的网络安全态势量化方法[J]. 通信学报, 2017, 38(10): 122-134.
- HU H, YE R G, ZHANG H Q, et al. Quantitative method for network security situation based on attack prediction[J]. Journal on Communications, 2017, 38(10): 122-134.
- [17] 樊雷, 余江明, 雷英杰. 面向 APT 攻击的分层表示模型[J]. 计算机工程, 2018, 44(8): 155-160.
- FAN L, YU J M, LEI Y J. Hierarchical representation model for APT attack[J]. Computer Engineering, 2018, 44(8): 155-160.

[作者简介]



孙澄 (1991-), 男, 江苏常州人, 信息工程大学硕士生, 主要研究方向为 APT 检测跟踪。



胡浩 (1989-), 男, 安徽池州人, 博士, 信息工程大学讲师, 主要研究方向为网络态势感知。



杨英杰 (1971-), 男, 河南郑州人, 博士, 信息工程大学教授, 主要研究方向为信息安全。



张红旗 (1962-), 男, 河北遵化人, 信息工程大学教授、博士生导师, 主要研究方向为网络安全、移动目标防御、等级保护和信息安全管理。