文章编号:1671-4598(2022)11-0265-07 DOI:10.16526/j.cnki.11—4762/tp.2022.11.039 中图分类号:TP393.4 文献标识码:A

基于历史数据分析的容器云安全风险评估方法

徐胜超
(广州华商学院 数据科学学院,广东 广州 511300)

摘要:容器云受到风险攻击会影响运维性能,无法有效保护内部存储的隐私数据安全;为了准确判断风险攻击类型,最大程度保证用户隐私数据安全,提出基于历史数据分析的容器云安全风险评估方法;根据云计算安全标准,对容器云风险等级进行分类;利用粗糙集算法挖掘容器云历史数据中的风险因素,获得风险因素归约集合;根据容器云的运行特点,通过德尔菲方法和决策隶属度矩阵计算安全风险权重;根据各风险间存在关联性,整合整体风险评估值,实现容器云安全风险评估;实验结果表明,该方法可以有效评估容器云安全风险,且评估结果较为准确,为用户保证隐私数据安全提供参考。

关键词:历史数据分析;数据挖掘;容器云安全;数据安全;风险评估

Method of Container Cloud Security Risk Assessment Based on Historical Data Analysis

XU Shengchao
(School of Date Science, Guangzhou Huashang College, Guangzhou 511300, China)

Abstract: The risk attack on the container cloud will affect the operation and maintenance performance, the internal stored privacy data security cannot effectively be protected. In order to timely intercept the risk attack, the container cloud security risk assessment method based on historical data analysis is proposed. The container cloud risk level is classified according to cloud computing security standards; The risk factors are excavated in container cloud historical data by using the rough set algorithm to obtain the reduced set of risk factors, according to the characteristics of container cloud, the security risk weight is calculated by the Delphi method and the decision membership matrix. Based on the correlation between each risk, the overall risk assessment value is integrated to realize the container cloud security risk assessment. The experimental results show that the proposed method can effectively evaluate the container cloud security risk, and the evaluation results are more accurate to ensure the security of user privacy data.

Keywords: historical data analysis; data mining; container cloud security; data security; risk assessment

0 引言

容器技术是近年来云计算行业发展中的新兴技术,容器虚拟化技术及其平台更是凭借自身部署快、移植性、轻量级、性能高、启动迅速等优势,被广泛应用于各大云服务项目中[1-5]。由于容器 Docker 的存在,容器技术极大地改变了云计算的发展,在容器云平台的一个单独的空间里,每个进程都有自己的文件系统、网络栈、进程组等,并且可以把 CPU、存储器等资源分配到这个独立的空间。在不同的容器中,过程是彼此独立的,这样就不会有任何的干涉和冲突,管理员可以在整个环境下配置并监视容器中的应用。容器云是云上的容器技术服务,但随着容器云存储的异质性数据不断增多,容器云会因数据不兼容而引发 CPU 数据异常、内存数据异常、网络数据异常等故障[6-9]。

国内外专家对容器云安全风险进行了研究。国内文献[10]提出基于粗糙集构建了容器云系统健康度评价模型,利用信息熵分割容器云安全监控数据的连续属性,建立一致性检查决策表,完成对容器云系统的安全评价。文献[11]基于灰色神经网络设计了云平台大数据安全风险评估方法,采用基于自治的元组划分方法分类云数据的安全风险信息;利用高斯密度谱提取风险信息特征;利用灰色神经网络构建信息风险分解模型;采用自适应差分改进方法评估云平台大数据的安全风险。国外研究文献[12]深入评估云计算的安全风险。分析云计算的复杂环境,提出了一个基于 Delphi 的云安全风险评估模型,用于识别、分析和评估云计算的安全风险。

基于以上研究成果,本文提出基于历史数据分析的容器云安全风险评估方法。根据云计算安全标准,将容器云的风险等级分为四类;根据风险等级,创新性地利用粗糙集算法挖掘和度量容器云历史数据中的三种风险因素对程序运行的攻击情况,并进行归约集合;基于风险因素度量结果,根据容器云的运行特点,设立隐私影响、脆弱性和威胁频度三个风险评估指标;通过德尔菲方法和决策隶属度矩阵计算安全风险权重。实验结果表明本文方法通过计

收稿日期:2022-06-24; **修回日期**:2022-08-29。
基金项目:广州华商学院校内导师制科研项目(2022HSDS07)。
作者简介:徐胜超(1980-),男,湖北武汉人,硕士,讲师,主要从事并行分布式处理软件方向的研究。
引用格式:徐胜超. 基于历史数据分析的容器云安全风险评估方法[J]. 计算机测量与控制,2022,30(11):265-271.

算容器云安全的隐私影响、脆弱性和威胁频度风险值，实现了精准风险评估，最大程度保证用户隐私数据安全。

1 容器云及其历史数据风险因素度量

1.1 容器云技术概述

容器云平台开展工作的时候，平台基本单位为容器，封装软件为平台运行环境。容器云平台主要侧重于容器的编排以及资源的部署、共享，结合容器技术与虚拟化技术，构建容器云平台的体系结构，如图 1 所示。从图 1 中可以看出，容器云平台的架构由下往上划分为设备层、共享资源层、集群服务层、应用层。设备层主要是指用户购买的服务器，路由器，集线器，物理存储等。共享资源层是以硬件资源库为基础，利用虚拟化技术将多个大型服务器设备按特定的类型进行分区。集群服务层次包括资源配置、项目发布、多个集群提供真实服务。应用层次主要是云客服端对容器云的访问。

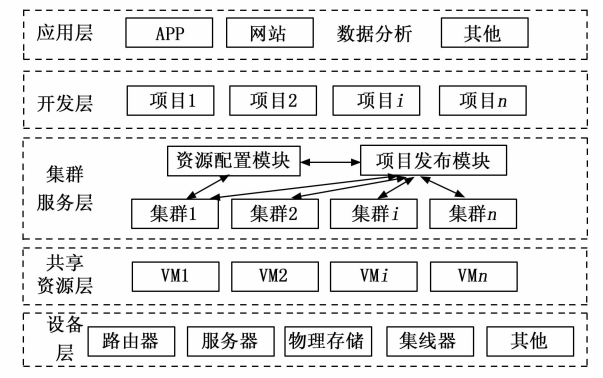

图 1 容器云平台的分层结构

集群服务层次是通过安装 Docker、Kubernetes 之类的容器平台软件来实现的，它可以一键式地安装 shell 脚本，并大量地配置集群。该系统是一种面向对象的人机交互系统，它可以让开发者将自己的应用程序以容器的形式发布到集群中，并为应用程序提供迁移、升级、扩充、回滚等服务。其中，资源配置模块的作用就是在容器云平台上进行资源分配，确保其安全性、稳定性、决策策略，以提高资源的使用效率和服务质量，并承担软件发布系统的开发和维护工作。

1.2 容器云资源的整合

在容器云类似是一种网络资源管理平台，其最重要的功能是资源整合。资源整合之前，需要对云资源进行虚拟化处理。将物理机的 CPU、内存、存储器等全部仿真出来，并将 Guest OS 的指令与下级硬件接口连接，然后捕获并处理这些对虚拟机敏感的权限命令。在此基础上，针对虚拟化的容器云资源按照图 2 表示流程进行整合处理。

如果要求容器云资源整合结果中不存在冗余资源数据，因此需要计算容器云资源中的任意两个数据之间的冗余度。若冗余度计算结果为 1，则表示判断的两个数据为冗余资源，需要删除其中一个。最终按照容器资源的存储空间顺序进行资源融合，并得出冗余度较低的云资源融合结果。

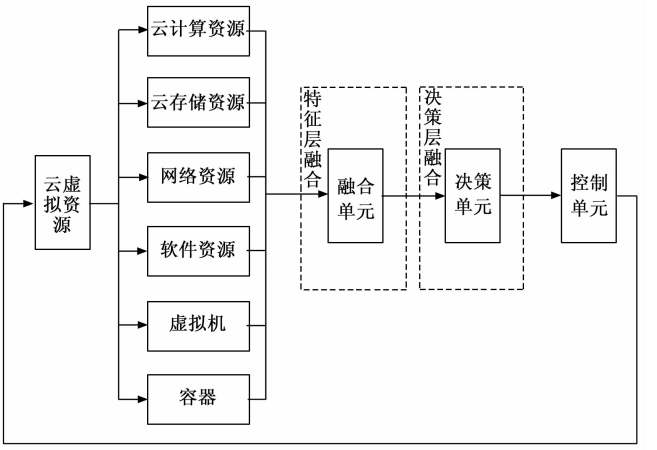

图 2 容器云平台资源整合处理流程图

容器云资源编排重组的实质是为计算节点指定一个容器，其目的是在较低的违约率条件下使用尽可能多的容器来实现最小的开销，从而实现最大的利润。排列重构大致可以分成两个阶段，第一步是缩减节点数目，将资源消耗较低的计算节点中的容器尽量转移到其他的节点，同时销毁闲置的计算节点，从而达到减少节点总量、降低运营费用的目的。其次，采用负载平衡策略，在两个节点间找到一对满足交换条件的容器，从而减少结点的碎片率，并实现负载平衡。最终以物理机为单位，将其内部的容器云资源根据 CPU 占用率从高到低进行排列，得出容器云资源序列的编排重组结果，由此完成容器云资源的部署任务。

类似于这种 Docker 这种容器云平台，资源管理模块还负责在容器云中的各个结点上对层次文件缓存进行同步，并通过 API 对 Docker 镜像中的元数据进行处理。为了在容器云中获得层次档案的缓存，必须在 Docker 引擎中增加一个界面，定期地将每个结点的层次缓存资源与物理机结点进行增量同步。

1.3 容器云资源整合的约束条件

为了管理好容器云内的物理资源，容器云资源部署的约束条件为：

$$vr_x^N \alpha_i^T \leqslant vr_{\text{Physics}}, x \in \{\text{cpu,ram,bw}\} \tag{1}$$

$$cr_x^S \beta_i^T \leqslant cr_{\text{fictitious}}, x \in \{\text{cpu,ram,bw}\} \tag{2}$$

$$cr_{\text{cpu}}^S \beta_i^T \leqslant vr_{\text{cpu}}^i \tag{3}$$

$$\forall j \in \{1,2,\cdots,U_i\}, \sum_{i=1}^{n} \delta_{i,j} = 1 \tag{4}$$

$$\forall j \in \{1,2,\cdots,H_i\}, \sum_{i=1}^{N} \sigma_{i,j} = 1 \tag{5}$$

式中，vr_x^N 表示物理机资源，cr_x^S 表示虚拟机资源，变量 α_i^T 和 β_i^T 均为向量，分别表示任意物理机的虚拟机和容器的部署情况，vr_{Physics} 和 $cr_{\text{fictitious}}$ 表示物理机和虚拟机的资源综合，具体包括 CPU、内存、带宽、三个部分，cr_{cpu}^S 和 vr_{cpu}^i 表示虚拟机和容器的 CPU 工作负载，$\delta_{i,j}$ 表示虚拟机数量，U_i 为虚拟机编号，$\sigma_{i,j}$ 表示容器数量，H_i 表示物理机编号。根据

公式（1～2）可以看出，部署在物理机上虚拟机的资源之和不能高于物理机能提供的资源，部署在虚拟机上的容器资源之和不能高于虚拟机资源的最大值，容器的工作负载不能高于虚拟机，每台虚拟机只能被部署在一台物理机上，同时每个容器只能被部署在一台虚拟机上。

通过上面的分析可以看出，容器云是将原有应用程序镜像虚拟化，如果程序本身携带病毒，病毒很容易通过应用程序入侵容器云程序，对 CPU、内存、网络带宽数据进行攻击，导致数据出现异常，严重干扰容器云的运维，甚至危害容器云内部存储的隐私数据的安全。因此，为了准确判断出风险攻击类型，最大程度保证用户隐私数据安全，需要对容器云进行安全风险评估。

1.4　容器云的数据风险因素

容器云的数据中的风险因素有 CPU 异常数据、内存异常数据、网络异常数据。假设决策系统 DS 称 $S=(U,A,\{V_a\},a)$ 代表云安全内知识系统（该知识系统基于《云计算服务安全能力要求》），其中 S 代表论域，$a:U\rightarrow V_a$ 代表单射 A 代表历史数据风险属性集合，a 代表风险等级，V_a 代表风险因素 $a\in A$ 域，U 是 V_a 内唯一参数值。

若 A 是通过容器云历史数据风险因素集合 C 与结论属性集合 D 所构成的，那么 C、D 满足 $C\cap D=\varphi$。在决策系统内，U 集合中的所有元素都存在对应的规则，其中，规则前件通过 C 确定，规则后件通过 D 确定。

假设不可分辨关系：决策系统 $S=(U,C\cup D)$，用 $B\subseteq C$ 描述属性子集，二元关系 $IND(B,D)=\{(x,y)\in U\times U:f(x,a)=f(y,a)\ \forall a\in B\}$ 则代表 S 不可分辨。

相对于风险因素等级判定指标 $S(U,A)$，设置历史数据安全属性 $B\subseteq A$，属性 $X\subseteq U$，以此可得到具体公式为：

$$BX=\{x\in U\mid [x]_{IND(B)}\in X\} \tag{6}$$

$$\overline{B}X=\{x\in U\mid [x]_{IND(B)}\cap X\neq\varphi\} \tag{7}$$

其中：式（6）表示为下近似，式（7）表示为上近似，而下近似 BX 则表示全部 X 子集内原子集并集，且包括 X 最小复合集。

假设 $X\subseteq U$ 上近似与下近似，把 U 进行划分，明确边界区域 $BND(X)$、正向区域 $POS(X)$ 以及负向区域 $NEG(X)$，可以得到具体公式为：

$$POS(X)=B(X) \tag{8}$$

$$NEG(X)=U-\overline{B}(X) \tag{9}$$

$$BND(X)=\overline{B}(X)-B(X) \tag{10}$$

属于 $POS(X)$ 的元素 x，同样也属于 X，而任何属于 $NEG(X)$ 的元素，则不属于 x，不过一定属于 X 补集，在一个元素 x 属于 $BND(X)$ 时，那么没有办法确认它属于 X，还是属于 X 补集。即 $\overline{B}X=POS(X)\cup BND(X)$。若 $BND(X)=\varphi$，这样就可以说明 X 代表精确集，相反，X 为粗糙集。

假设数据属性 B 与 $R\subseteq U$ 间的互相依赖，能够使用属性间依赖函数相互衡量[13]，具体公式为：

$$y_R(B)=\frac{card(POS_R(B))}{card(U)} \tag{11}$$

$$POS_R(B)=\bigcup_{X\in U/IND(B)}RX \tag{12}$$

式中，$card(\cdot)$ 代表集合基数，$POS_R(B)$ 代表属性集 R 处于 $U/IND(B)$ 内正区域。

在属性 α 加入 R，计算分类 $U/IND(B)$ 的重要程度为：

$$SGF(a,R,B)=y_R(B)-y_{R-\{a\}}(B) \tag{13}$$

属性 α 依赖于 B 以及 R，所以处于不同背景，属性有可能会不同。若定义 D 代表决策属性，那么 $SGF(\alpha,R,D)$ 可反映：把 α 添加至属性 R 内后，改变了 R 和 D 间依赖程度，体现属性 α 重要性。

若数据的冗余属性[14]相对于属性 D 以及 R，则属性 $\alpha\in R$，若 $POS_R(D)=POS_{R-\{a\}}(D)$，那么 α 在 R 内则为冗余，反之，α 在 R 内相对于 D 是必要的。

通过对容器云安全属性进行归约[15]，可以使每个条件属性存在互相关联，通常为 $C'\subset C$，其中 C' 内的云安全属性能够确认结论属性 D 取值[16-17]。

容器云安全数据属性的归约条件：在决策系统 $S=(U,C\cup D)$ 内，集合 C 归一代表 C 的非空子集 C'，具体满足如下所示：

1）$IND(C',D)=IND(C,D)$。

2）不存在 $C''\subset C'$，令 $IND(C'',D)=IND(C,D)$。

则 C 的全部归约集合可以标记成 $RED(C)$。

以上集合即为容器云的历史数据中的风险因素归约集合，该集合中包含了风险因素 CPU 异常数据、内存异常数据、网络异常数据对程序运行的攻击类型和攻击结果。在此基础上，建立容器云安全风险评估模型。

1.5　容器云可信环境

容器云可信环境由管理节点和计算节点共同构成，公式如下所示：

$$En=\{M,N_1,N_2,\ldots,N_n\} \tag{14}$$

其中：E 表示可信容器云环境，M 则表示可信容器云其中的管理节点，$N_i(i=1,2,\ldots,n)$ 表示可信容器云中的计算节点。

可信容器云中的管理节点 M 代表容器云整体环境的安全管理中心，能够管理容器云环境中全部的计算节点，主要由系统管理、安全管理以及审计管理三个部分共同构成。

而可信容器云中的计算节点 N 在整体容器云环境中，负责进行业务处理，其上运行容器引擎和容器实例，具体公式如下：

$$N=\{D,C_1,C_2,\ldots,C_n\} \tag{15}$$

其中：N 就是可信容器云的计算节点，D 则表示可信容器云上运行的容器引擎，能够管理容器实例，$C_i(i=1,2,\ldots,n)$ 就是容器引擎所管理的容器实例。

2　容器云安全风险评估模型

由于容器云是对应用程序镜像虚拟化，因此受到应用程序中病毒和外来侵入病毒的双面威胁，且容器云内存在了大量异质性数据，因此程序运行较为脆弱，数据中包含了重要隐私信息。根据上节利用粗糙集算法根据容器云历

史数据归纳计算出的风险因素 CPU 异常数据、内存异常数据、网络异常数据对程序运行的攻击方式、攻击位置和攻击结果，将安全风险评估的一级指标设为隐私影响、威胁频度、脆弱性[18-20]。首先计算指标权重。

2.1 对容器云中的镜像文件进行扫描

在容器云的平台上装备“扫描容器”，对容器云内的仓库镜像、节点镜像进行获取，并通过储存在扫描容器内部的镜像文件扫描器对镜像文件中的软件包进行分离，以 CVE 安全漏洞库、Webshell 库和病毒木马库为基础，深度扫描软件包，对其中存在的漏洞以及安全风险进行发掘，逐层对其进行敏感扫描。扫描内容主要包括：镜像中的软件包和文件。

对镜像进行扫描后，使其分离为对应的层和软件包，再对软件包进行检查，对于镜像中的文件，要对其进行分层提取，同时对分层后的文件逐层进行检测。其中主要应用历史数据分析法、模糊哈希、YARA 规则以及机器学习等方式。

镜像扫描的流程如下所述：

1）对容器进行扫描，从中提取容器镜像，并将其保存为压缩包，格式为 tar；

2）用拆包器将属于 tar 格式的镜像拆分成镜像层；

3）对 manifest. json 文件中的镜像层进行提取；

4）通过 layer _ id 对数据文件进行解析；

5）对可疑文件通过恶意病毒和 WebShell 进行检测打分；

6）对镜像中的软件包版本进行解析；

7）将解析结果与 CVC 漏洞进行匹配，发掘其中的安全风险；

8）分析镜像历史行为以及证书文件；

9）对漏洞信息进行综合打分；

10）将镜像扫描报告输出。

2.2 容器云安全风险指标权重

通过德尔菲方法获取安全指标的初步权重。再通过决策隶属度矩阵计算安全风险最终权重结果。具体分为以下几个步骤。

1）组建评估专家小组。

选取容器云、风险评估、社会管理等领域的专家若干，组建专家小组，每组包含三个领域的专家至少一名。

2）设计调查问卷。

依据相关材料设计调查问卷，调查问卷中全部为有助于确定指标权重的信息。

3）专家小组评估权值。

每个专家小组成员独立地依据调查问卷评估指标权重值。

4）获得结果。

重复步骤 2)、3)，直至专家小组的评估结果趋于一致，将此评估结果作为指标的初步权重值集合。为了避免初步权重值集合具有主观性误差，考虑容器云安全脆弱性、威胁频度以及隐私影响因素间存在的关联性，因此结合马尔科夫链[21]计算上节得出的初步权重值集合，构建决策矩阵，进一步对所有指标实行归一化处理，提高指标权重的客观性。

设评分影响隐私的风险因素为 j，可以得到隐私影响指标权重具体公式为：

$$p_{jk}=\frac{E_k}{\sum_{k=1}^{E}E_{jk}} \tag{16}$$

式中，E 代表评估指标个数，k 为隶属度系数。则隐私影响指标权重向量为 $\bar{\omega}=(p_{jk1},\ p_{jk2},\ \cdots,\ p_{jkn_1})$，其中 n_1 代表隐私影响评价指标内的元素个数。

通过公式（16）计算出脆弱性因素 f、威胁频度因素 t 的指标权重分别为 p_{jfk}、p_{jtk}。

则威胁频度指标权重向量为 $\mu=(p_{tk1},\ p_{tk2},\ \cdots,\ p_{tkn_2})$，其中 n_2 代表威胁频度评价指标内的元素个数。

则脆弱度指标权重向量为 $\theta=(p_{fk1},\ p_{fk2},\ \cdots,\ p_{fkn2})$，其中 n_3 代表威胁频度评价指标内的元素个数[22]。

那么三项指标权重合集公式为：

$$R=\eta+\theta+\bar{\omega} \tag{17}$$

根据指标权重合集 R 构建决策矩阵 $\boldsymbol{Q}$，具体公式为：

$$\boldsymbol{Q}=\begin{bmatrix}q_{11}&q_{12}&q_{13}&q_{14}\\q_{21}&q_{22}&q_{23}&q_{24}\\q_{31}&q_{32}&q_{33}&q_{34}\\q_{41}&q_{42}&q_{43}&q_{44}\end{bmatrix} \tag{18}$$

式中，q 代表决策权重。

进一步对 $\boldsymbol{Q}$ 内的每一行实行归一化处理，可以得到公式为：

$$\gamma_{ij}=\frac{q_{ij}}{\sum_{j=1}^{n}q_{ij}} \tag{19}$$

最终获得评估指标最终权重。

2.3 实现容器云安全风险评估

基于容器云安全风险评估指标最终权重，对容器云的历史数据中的风险因素归约集合 $\boldsymbol{RED}(\boldsymbol{C})$ 建立风险值评判矩阵[23-24]。

$$\boldsymbol{RED}(\boldsymbol{C})_{\gamma}=\begin{bmatrix}C_{\gamma_{11}}&C_{\gamma_{12}}&C_{\gamma_{13}}&C_{\gamma_{14}}\\C_{\gamma_{21}}&C_{\gamma_{22}}&C_{\gamma_{23}}&C_{\gamma_{24}}\\C_{\gamma_{31}}&C_{\gamma_{32}}&C_{\gamma_{33}}&C_{\gamma_{34}}\\C_{\gamma_{41}}&C_{\gamma_{42}}&C_{\gamma_{43}}&C_{\gamma_{44}}\end{bmatrix} \tag{20}$$

在不损失信息的前提下，采用最简单的表达方式，集合管理评判属性。

核心数据分辨矩阵：在容器云安全系统 S 内，关于属性集 $RED(C)_{\gamma}$ 分辨矩阵 $M(C)_{\gamma}=(M_{i,j})_{n\times n}$，具体公式：

$$M_{i,j}=\begin{cases}\varphi & (x_i,x_j\in RED(C)_{\gamma}\text{ 的同一等价类})\\ \{c\in C:f(x_i,c)\} & (x_i,x_j\in RED(C)_{\gamma}\text{ 的不同等价类})\end{cases} \tag{21}$$

式中，$M(C)=(M_{i,j})_{n\times n}$ 表示区分指标权重项 x_i，x_j 完整

信息[25-26]。

计算容器云的整体风险值：

$$S_\gamma = \sum_{i=1}^{Q} M(C)_{\gamma i} \tag{22}$$

通过上述步骤，实现容器云安全风险评估。

3　实验与性能分析

3.1　实验环境

为了验证基于历史数据分析的容器云安全风险评估方法在实际应用中是否能够达到合格标准，选择在容器云 Kubernetes 平台 Slave 采集的四组数据作为实验数据。第一组数据为 600 个基本训练数据；第二组数据在第一组基础上插入 100 条 CPU 异常数据；第三组数据在第二组数据基础上插入 100 条内存异常数据；第四组数据在第三组数据基础上插入 100 条网络异常数据。为了模拟容器云风险攻击，在容器中运行 Tensorflow 卷积神经网络 Demo 模拟 CPU 风险攻击、内存攻击、网络攻击对容器云网络安全风险进行评估测试。测试前对数据进行预处理：

1）将攻击数据矩阵格式转换为二维数据；

2）分离输出分类为 OneHot 编码；

3）对构造出来的二维数据进行归一化处理。

Tensorflow 卷积神经网络 Demo 测试模型采用最经典的 LeNet-5 结构。以 20×12（每行 12 个特征）的卷积层格式输入数据集进行测试。

设置风险评估指标，引入安全度 ν，并且设置 $\nu \in [0,1]$。以此作为风险值的评估条件，对标准风险等级进行细化，将四级风险进一步划分为五级风险，具体指标如表 1 所示。

表 1　风险值评估指标

编号	风险评估等级	隐私影响	威胁频度	脆弱性
1	无风险	0～0.2	0～0.2	0～0.2
2	少量风险	0.21～0.4	0.21～0.4	0.21～0.4
3	一般风险	0.41～0.6	0.41～0.6	0.41～0.6
4	较大风险	0.61～0.8	0.61～0.8	0.61～0.8
5	极大风险	0.81～1	0.81～1	0.81～1

3.2　实验结果与分析

为了验证容器云安全隐私影响情况，通过重复性的 CPU 风险攻击、内存攻击、网络攻击，评估搭建的容器云网络隐私影响风险结果，具体如图 3 所示。

通过观察图 3 能够看出，CPU 风险攻击在实验次数为 40 次时，出现大幅度上升，这是因为 CPU 风险攻击导致搭建的容器云网络平台运行出现卡顿，影响运行速度，其隐私影响风险值处于 0～0.6 之间，说明容器云网络存在一般风险，有一定危险性。而针对内存攻击、网络攻击，由于并未入侵成功，因此一直保持一个平稳的状态，并未对容器云安全造成严重影响，其风险值处于 0～0.2 之间，说明容器云网络无风险，容器云网络数据安全。

同样方式评估容器云安全威胁频度风险，具体如图 4 所示。

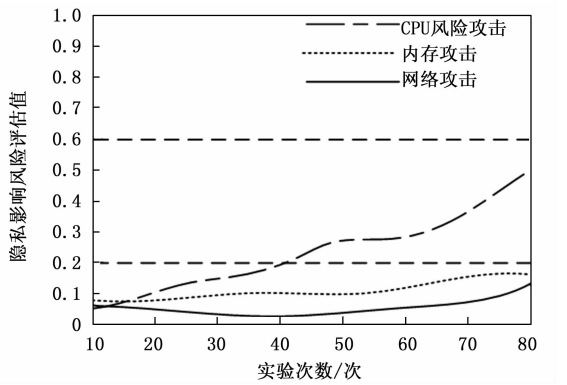

图 3　隐私影响风险评估

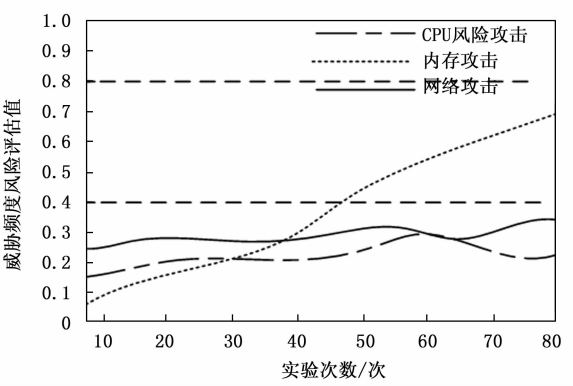

图 4　威胁频度风险评估

通过观察图 4 能够看出，内存攻击在实验次数为 40 次时，出现大幅度上升的情况，这主要是因为在实验次数为 30 次之前的容器云忽略了内存入侵安全，并未实行有效处理，因此对整体网络造成了累积性的伤害，使其威胁频度风险上升，整体曲线处于 0～0.8 之间，存在较高风险，其容器云安全性较差。而 CPU 风险攻击和网络攻击则对容器云安全影响较低，这是因为 CPU 风险攻击和网络攻击虽然导致容器云运行速度降低，但并不会使容器云数据丢失，其风险值为 0～0.4 之间，说明容器云安全风险较小。

同样操作下评估容器云的脆弱性风险，具体结果如图 5 所示。

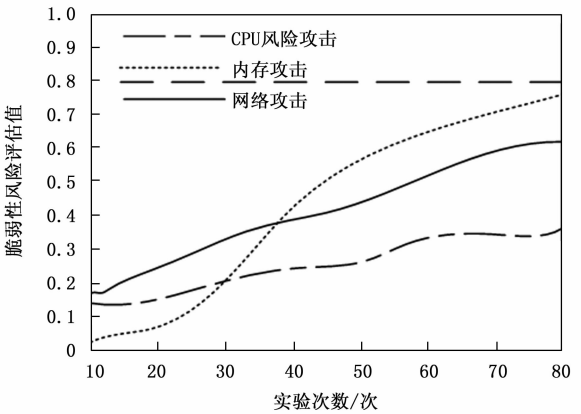

图 5　脆弱性风险评估

通过观察图 5 能够看出，三种攻击对于容器云脆弱性影响均较大，且二者都是随着实验次数的增加，其容器云网络的脆弱程度提高，当实验次数为 80 次时，其风险值整体曲线处于 0～0.8 以内，说明容器云安全性较差，存在较大风险。

因为容器云的各风险之间存在关联性，所以要计算所有风险数值，本文将 3 种风险评估结果的风险数值进行整合处理，获得整体风险评估具体结果如图 6 所示。

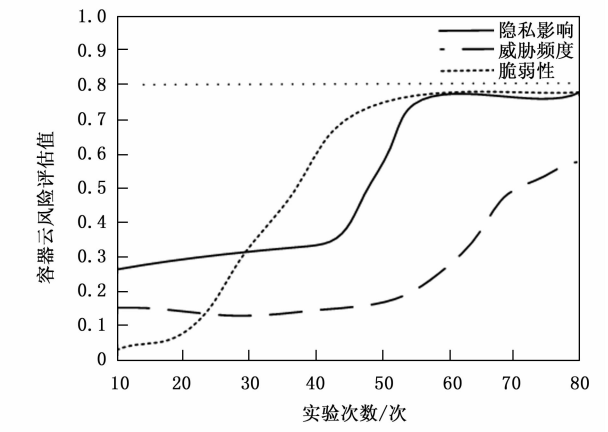

图 6 三种风险整合后的整体风险评估结果

观察图 6 能够看出，经过重复性实验，且并未对三种攻击行为实行有效的后续补救措施，因此，容器云整体风险评估值较高，均处于 0～0.8 之间，说明容器云网络环境存在较大风险，符合实验设置的行为，其评估结果精度较高。

综合上述分析，基于历史数据分析的容器云安全风险评估方法能够有效实现容器云安全风险评估，且评估结果较为准确，这主要是因为基于历史数据分析的容器云安全风险评估方法，通过分析历史数据，生成丰富的经验，从而有效提高容器云安全风险评估精度。

4 结束语

本文提出一种基于历史数据分析的容器云安全风险评估方法，通过挖掘出历史中影响容器云安全性能的风险因素，计算整体容器云安全的脆弱性、隐私影响以及威胁频度风险值，将三种风险进行整合，即可实现容器云安全整体的风险评估。

参考文献：

[1] 龚 坤，武永卫，陈 康. 容器云多维资源利用率均衡调度研究 [J]. 计算机应用研究，2020，37 (4)：1102－1106.

[2] 李 磊，薛 洋，吕念玲，等. 基于李雅普诺夫优化的容器云队列在线任务和资源调度设计 [J]. 计算机应用，2019，39 (2)：494－500.

[3] TAN B，MA H，MEI Y. Novel Genetic Algorithm with Dual Chromosome Representation for Resource Allocation in Container-Based Clouds [C] // Novel Genetic Algorithm with Dual Chromosome Representation for Resource Allocation in Container-based Clouds 2019.

[4] VALDEZ M G，JJM Gue rv ós. A container-based cloud-native architecture for the reproducible execution of multi-population optimization algorithms [J]. Future Generation Computer Systems，2021，116 (1)：234－252.

[5] SHI T，MA H，CHEN G. Energy-Aware Container Consolidation Based on PSO in Cloud Data Centers [C] // IEEE Congress on Evolutionary Computation，IEEE，2018.

[6] KAFHALI S E，MIR I E，HANINI M. Security Threats，Defense Mechanisms，Challenges，and Future Directions in Cloud Computing [J]. Archives of Computational Methods in Engineering，2021：1－24.

[7] CROCKFORD N. An Introduction to Risk Manageme-nt . Woodhead-Faulkner [R]. 1986.

[8] 李 旭，荣梓景，阮晓曦. 关系决策系统中相对不可区分和区分关系的约简 [J]. 计算机应用，2019，39 (10)：2852－2858.

[9] XMA E，FRB C，MSD E. Measuring satisfaction and power in influence based decision systems [J]. Knowledge-Based Systems，2019，174 (C)：144－159.

[10] 张可颖，龙士工，吕尚青，等. 基于粗糙集的容器云系统健康度评价建模 [J]. 计算机技术与发展，2020，30 (4)：69－74.

[11] 耿文莉，高梦瑜. 基于灰色神经网络的云平台大数据安全风险评估 [J]. 科学技术与工程，2021，21 (28)：11932－11937.

[12] YOUSSEF A. A delphi-based security risk assessment model for cloud computing in enterprises [J]. Journal of Theoretical and Applied Information Technology，2020，98 (1)：151－162.

[13] 马莉莉，刘江平. 基于数据挖掘的光纤通信网络异常数据检测研究 [J]. 应用光学，2020，41 (6)：1305－1310.

[14] 张大伟，叶蓓蓓. 面向冗余数据消除的多维异质网络数据传输控制方法 [J]. 中国电子科学研究院学报，2019，14 (5)：519－523，552.

[15] 张逸然，陈 龙，安向哲，等. 面向 GPU 计算平台的归约算法的性能优化研究 [J]. 计算机科学，2019，46 (2)：306－314.

[16] YU Q，REN J，ZHANG J，et al. An Immunology-Inspired Network Security Architecture [J]. IEEE Wireless Communications，2020，PP (99)：1－6.

[17] 冯译萱，张月霞. 一种时序有向网络中的链路预测方法 [J]. 计算机工程与应用，2019，55 (21)：151－157.

[18] ZHOU F，DU Y，YUAN Y，et al. The Cross-networks Impact Analysis and Assessment in Multilayer Interdependent Networks：A Case Study of Critical Infrastructures [J]. International Journal of Modern Physics C，2019，30 (7)：2087－A1－5.

[19] 郭曙杰，李志华，蔺凯青. 云环境下基于模糊隶属度的虚拟机放置算法 [J]. 计算机应用，2020，40 (5)：1374－1381.

[20] 宋 悦，杨乃定，张延禄，等. 基于云模型的研发网络脆弱性评价研究 [J]. 科技管理研究，2019，39 (6)：49－54.

[21] RYU J，PARK J，LEE J，et al. Community-based diffusion scheme using Markov chain and spectral clustering for mobile

social networks [J]. Wireless Networks, 2019, 25 (2): 875 - 887.
[22] 李　国，李静雯，王　静，等. 基于威胁状态的新型机载网络安全风险评估改进模型 [J]. 现代电子技术，2019，42 (2)：41 - 45.
[23] 马峻岩，张　特，王　瑾. 基于任务转移概率的感知节点异常运行状态检测方法 [J]. 北京邮电大学学报，2019，42 (3)：37 - 42.
[24] 杨小冈，李维鹏，马玛双. 概率框架下多特征显著性检测算法 [J]. 电子学报，2019，47 (11)：2378 - 2385.
[25] 岳少博，王清河，王晓春，等. 基于云技术的网络节点关联性风险评估仿真 [J]. 计算机仿真，2020，37 (8)：247 - 251.
[26] 文家朝，杨鸿章. 基于数值模拟的网络安全风险量化参数优化分析 [J]. 科学技术与工程，2019，19 (7)：183 - 188.

(上接第 250 页)

慢，导致出现较大误差；环境高度最高为 B 点废墟 42 米，最低为 A 点水域 39，平均误差 0.425；光照系数平均误差 5 lx；生命特征正确检测 4 次，在 A 点检测到生命时，向用户中心返回了 A 点经纬度：23.248 69，113.864 036，提示生命所在位置。根据以上指标，机器人能完成一般信息采集任务。采集信息如表 1 所示。

表 1　机器人环境信息采集表

坐标	空气(PPM)		高度/m		光照系数/lx		生命特征	
	现场	云端	现场	云端	现场	云端	现场	云端
A	35	30	40	39	1 500	1470	有	有
B	38	37	40.5	42	1 460	1 480	无	无
C	18	21	39	39.2	1 100	1 150	无	无
D	14	14	40	41	1 250	1 300	无	无

5　结束语

在灾区环境中，多传感器信息融合是机器人感知周围环境信息技术中一项行之有效、必不可缺的关键技术。

全地形信息采集机器人，在混合式融合架构下充分利用了多传感器技术，以实现对多感应器的探测信息的合理支配与利用，将多感应器的空隙及时间上的多余或相互利用信息加以综合。

实验结果表明，机器人凭借多种传感器数据融合，能够提高机器人定位精度，通过扩展卡尔曼滤波算法，能够校正崎岖环境下累积的误差，从而使得机器人得到最优信息，助力灾后救援。

参考文献：

[1] 白雪梅. 水域智能应急救援装备发展现状 [J]. 中国船检，2020 (8)：56 - 58.
[2] 周世鹏. 基于 ARM 的水面机器人控制系统研究 [D]. 杭州：杭州电子科技大学，2018.
[3] 杨健玮，龚椿彭，姜逸川，等. 水上救援机器人的设计 [J]. 机电工程技术，2020，49 (3)：28 - 29.
[4] 冯迎宾，张　婧. 水面救援机器人控制系统设计及控制算法研究 [J]. 光电技术应用，2020，35 (5)：60 - 65.
[5] 付瑞玲，王　宁，杜志强. sss 基于多传感器信息融合的火灾报警器设计 [J]. 计算机测量与控制，2018，26，232 (1)：206 - 208，212.
[6] 李培源. 基于物联网技术的输电线路状态智能传感与监测系统研究 [D]. 重庆：重庆大学，2020.
[7] 许承宇，徐绍凯. 基于视觉延时补偿的无人机室内实时导航系统 [J/OL]. 动力学与控制学报：1 - 8 [2021 - 07 - 10]. http://kns.cnki.net/kcms/detail/43.1409.o3.20210624.1203.012.html.
[8] 司书斌，赵大伟，徐婉莹，等. 视觉—惯性导航定位技术研究进展 [J]. 中国图象图形学报，2021，26 (6)：1470 - 1482.
[9] 袁千贺，田　昕，沈斯杰. 基于多传感器融合的移动机器人定位 [J]. 计算机系统应用，2022，31 (3)：136 - 142.
[10] 李　帆，王茂森，戴劲松. 某康复训练运动信息采集系统 [J]. 兵工自动化，2019，38 (6)：24 - 28.
[11] 杨晓丽，刘博涛，赵　哲，等. 基于视觉与激光雷达数据融合的前车识别研究 [J]. 汽车实用技术，2020，45 (18)：22 - 24.
[12] 刘英旋，万　腾. 轮式田间信息采集机器人开发与控制系统设计 [J]. 农机化研究，2022，44 (1)：124 - 129.
[13] 谢　晖，王书涵，陈相全，等. 基于 LoRa 技术的通用型环境监测节点低功耗设计 [J]. 计算机测量与控制，2022，30 (7)：41 - 48.
[14] 赵一兵，刘昌华，郑　震，等. 基于多传感信息融合的智能车辆定位方法 [J]. 汽车工程学报，2021，11 (1)：1 - 10.
[15] 赵文龙，高建烨，何　涛. 融合多传感器的自主 AGV 定位研究 [J]. 现代制造工程，2021 (10)：85 - 90，113.
[16] 张书亮，谭向全，吴清文. 基于多传感器融合技术的室内移动机器人定位研究 [J]. 传感器与微系统，2021，40 (8)：53 - 56.
[17] 李昌明，梅　莉，秦东兴. 基于扩展卡尔曼滤波的轮式移动机器人定位技术 [J]. 中国测试，2011，37 (6)：76 - 79.
[18] 叶　润，刘　鹏，张凌浩，等. 基于多传感器数据融合的无人机 GPS 欺骗检测研究 [J]. 计算机测量与控制，2020，28 (12)：12 - 16.
[19] 谭光兴，符丹丹，丁　颖，等. 基于扩展卡尔曼滤波的汽车行驶状态估计 [J]. 广西科技大学学报，2020，31 (1)：18 - 24，44.
[20] 张书亮. 基于多传感器融合的室内移动机器人定位与导航研究 [D]. 北京：中国科学院大学 (中国科学院长春光学精密机械与物理研究所)，2021.
[21] 黄俊杰，李海滨，贾翠玲. 基于卡尔曼滤波的 UWB 与里程计融合定位方法 [J/OL]. 机床与液压：1 - 11 [2021 - 07 - 10]. http://kns.cnki.net/kcms/detail/44.1259.TH.20210624.0948.002.html.
[22] 唐俊淮. 移动机器人同时定位与地图构建研究 [D]. 杭州：浙江工业大学，2014.
[23] 蒋金豹. 基于 GPS/惯导数据融合的移动机器人自主导航控制研究 [D]. 芜湖：安徽工程大学，2020.