基于多层次特征的 RoQ 隐蔽攻击无监督检测方法

赵静[1,2]，李俊[1,2]，龙春[1,2]，万巍[1,2]，魏金侠[1,2]，陈凯[1,2]
（1. 中国科学院计算机网络信息中心，北京 100083；2. 中国科学院大学计算机科学与技术学院，北京 100049）

摘 要：针对 RoQ 攻击隐藏在海量背景流量中难以识别，且现有样本稀少无法提供大规模学习数据的问题，提出了在极少先验知识条件下基于多层次特征的 RoQ 隐蔽攻击无监督检测方法。首先，考虑到大部分正常流量会对后续结果产生干扰，基于流特征，研究了半监督谱聚类的流量筛选方法，实现被筛除的流量中正常样本比例接近 100%。其次，为了找到隐蔽攻击特征与正常流量之间的微小差异且不依赖于攻击样本，基于时序包特征，构造了基于 n-Shapelet 子序列的无监督检测模型，使用具有明显辨识度的局部特征来辨别微小差异，从而实现 RoQ 隐蔽攻击的检测。实验结果表明，在只有少量学习样本的情况下，所提方法与现有方法相比具有较高的精确率和召回率，对规避攻击具有稳健性。

关键词：RoQ 隐蔽攻击；谱聚类；半监督聚类；Shapelet 子序列
中图分类号：TP18
文献标志码：A
DOI: 10.11959/j.issn.1000−436x.2022166

Unsupervised detection method of RoQ covert attacks based on multilayer features

ZHAO Jing[1,2], LI Jun[1,2], LONG Chun[1,2], WAN Wei[1,2], WEI Jinxia[1,2], CHEN Kai[1,2]
1. Computer Network Information Center, Chinese Academy of Sciences, Beijing 100083, China
2. School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China

Abstract: To solve the problems that RoQ covert attacks are hidden in overwhelming background traffic and difficult to identify, besides the existing samples are scarce and cannot provide large-scale learning data, an unsupervised detection method of RoQ covert attacks based on multilayer features was proposed under the condition of very little prior knowledge. First, considering that most normal flow might interfere with subsequent results, a classification method based on semi-supervised spectral clustering was studied by flow characteristics, so that the proportion of normal samples in the filtered traffic was close to 100%. Secondly, in order to distinguish the nuance between the hidden attack features and normal flow without relying on the attack samples, an unsupervised detection model based on the n-Shapelet subsequence was constructed by packet characteristics, and the subsequences with obvious difference were used, which enabled detection of RoQ convert attacks. Experimental results demonstrate that with only a small number of learning samples, the proposed method has higher precision and recall rate than existing methods, and is robust to evading attacks.

Keywords: RoQ converts attack, spectral clustering, semi-supervised clustering, Shapelet subsequence

收稿日期：2022−05−23；**修回日期：**2022−08−12
通信作者：龙春，longchun@cnic.cn
基金项目：国家自然科学基金资助项目（No.61672490）；中国科学院基金资助项目（No.CAS-WX2022GC-04）；中国科学院“青年创新促进会”基金资助项目（No.2022170）
Foundation Items: The National Natural Science Foundation of China (No.61672490), The Research Program of Chinese Academy of Sciences (No.CAS-WX2022GC-04), The Research Program of Youth Innovation Promotion Association of CAS (No.2022170)

0　引言

随着当前网络安全防护方法和能力的升级，拒绝服务（DoS, denial of service）等常见攻击已可以被轻松识别。然而，一类新的隐蔽攻击悄然而生，即降质（RoQ, reduction of quality）攻击。与传统 DoS 相比，RoQ 攻击的原则不再是使目标系统完全丧失对正常请求的服务能力，而是利用互联网广泛采用的确保网络公平性、稳定性的自适应机制，通过周期性地在短暂间隔内突发攻击数据包，降低目标主机的服务质量。这种攻击流量与正常流量几乎没有区别，很容易绕过常规的入侵检测系统。

具体分析，RoQ 攻击主要有以下特点。

1) 从流量特点上看，RoQ 攻击呈现出脉冲式的特征。在一个攻击周期中，只有大约$\frac{1}{5}$的时间存在攻击流量。因此，RoQ 攻击也称为脉冲式隐蔽攻击[1]。

2) 从攻击方式上看，RoQ 攻击广泛分布的攻击者在目标路由器的末端聚合攻击流量，使攻击流量的分布在整个网络中更加分散，平均流量更低。

3) 从攻击行为上看，RoQ 攻击可以使用从网络层到应用层的任何协议合法流量，与普通应用具有相同的行为特征，能够完全隐藏在海量网络流量中。

4) 从攻击目的上看，RoQ 攻击主要是为了使系统运行速度减慢，服务质量下降。因此，大部分网络管理员往往将此类情况归因于系统故障或网络线路故障，忽略了攻击的可能性而导致处理延后。

仅利用流特征不能完全准确地区分 RoQ 攻击流量和正常流量。但是，从包特征上看，异常流量整体分布的形态差异与正常流量相比具有一定的区分度（例如，将两段不同的流量从时域转换到频域，就能发现其中的不同）。因此，利用时序特征对这类攻击进行检测更合适。再者，真实环境中 RoQ 攻击样本稀少，且每次攻击都具有独立性、特殊性，不容易使用监督学习方法来学习隐蔽的攻击特征。因此，要求检测方法在零先验知识或极少先验知识的情况下发挥作用。综上，本文在完成大量攻击试验，并梳理 RoQ 攻击相关特征之后，提出了一种基于多层次特征的 RoQ 隐蔽攻击无监督检测方法，主要创新点如下。

1) 提出了基于多层次特征的 RoQ 隐蔽攻击检测的两阶段方法。根据目标在不同阶段使用不同层面的特征，整体上达到了较好的效果。

2) 为了筛除大部分正常流量，减少对后续结果的干扰，研究了基于正样本监督信息的半监督谱聚类方法。利用流特征，构造少量正样本监督信息以修正样本中偏离过大的离群点。同时构造标签损失算法来评价算法损失度以便调整参数，使筛除正样本的准确率接近 100%。

3) 考虑到 RoQ 攻击流量特征不明显，为了提高检测率，将经典 Shapelet 时间序列算法修改为多维算法，利用流量时序包特征构造了最具辨识度的 n-Shapelet 基序列空间。经过空间投影后的特征能够最大限度体现相似序列差别最大的局部特征，提高了 RoQ 攻击特征的识别率。

4) 考虑到 RoQ 攻击样本稀少无法训练检测模型但仍需实现高精度检测，构造了无监督分类优化模型。该模型能够根据检测结果多次迭代直至收敛到最优解，实现了无学习样本的高精度检测。

1　相关工作

一直以来，很多研究者着重研究 RoQ 攻击如何绕过检测系统，以期更好地解决 RoQ 攻击检测问题。Guirguis 等[2]最早提出了一种利用 TCP 协议的低速率拒绝服务攻击，这种攻击没有明显的异常特征，利用传输控制协议（TCP, transmission control protocol）中自适应机制造成信道阻塞，从而达到攻击目的。紧接着，Luo[3]介绍了一种短时间内阻塞链路的脉冲式拒绝服务攻击；Guirguis 等[1,4]提出了利用动态负载均衡机制的 RoQ 攻击和针对内容自适应机制的脉冲式攻击。近几年的主要研究包括利用 KeepAlive 机制占满服务器等待队列的攻击[5]、强效的 Shrew 变种 FB-Shrew 攻击[6]、利用物联网设备之间消息队列遥测传输（MQTT, message queuing telemetry trandport）协议漏洞的攻击[7]、隐蔽漏洞扫描等其他脉冲探测类攻击[8]。总体来讲，RoQ 攻击隐匿性高，缺乏已知样本，是一种极难发现的新型攻击。

现有针对 RoQ 隐蔽攻击的检测方法大致可以分为基于信号处理、基于统计分析、基于数据建模及基于人工智能技术这 4 种类型。

基于信号处理的检测方法是最常用的，其核心思想是将流量时域特征转换成频域，来增加正常流量和异常流量的区分度。Chen 等[9]采用了 2 个新的频域特征：傅里叶功率谱熵（FPSE, Fourier power spectral entropy）和小波功率谱熵（WPSE, wavelet power spectrum entropy），提高了检测率。Agrawal 等[10]将流量从时域变换到频域上，通过功率频谱分

布判断攻击。基于信号处理的检测方法能够从一定程度上检测出 RoQ 攻击，但是在检测深度伪装、特征不明显的攻击时效果不佳。

基于统计分析的检测方法根据流量偏离正常状态的特征行为来进行识别。吴志军等[11]根据低速率拒绝服务（LDoS, low-rate denial of service）攻击周期性的小信号特征构造出特征值估计矩阵来判断是否发生了攻击。Tang 等[12]提出了一种自适应指数加权移动平均算法，根据加权值的变化检测攻击。在之后的研究中又基于脉冲流量离散特性提出一种新的统计方法来进行检测[13]。为了进一步提高检测率，Tang 等[14]提出利用用户数据报协议（UDP, user datagram protocol）流量与传输控制协议（TCP, transmission control protocol）流量的比值（UTR, ratio of UDP traffic to TCP traffic）能够更准确区分攻击流量和正常流量。基于统计分析的检测方法在设立阈值的过程中需要大量的专业领域知识和工程技能，烦琐复杂，很难大规模使用。

基于数学建模的检测方法是利用数据方法对网络流量进行分析，区分出正常流量中的隐蔽 RoQ 攻击。如 Wu 等[15]研究了 RoQ 攻击对网络流量的多重分形特性的影响，提出了 MF-DFA（multifractal detrended fluctuation analysis）算法，计算正常情况和攻击情况下的网络流量 Hölder 指数的差值来判断是否发生了攻击。

考虑到隐蔽 RoQ 攻击的特征提取困难，一些研究者开始考虑使用人工智能技术。例如，Koay 等[16]基于熵特征集成了循环神经网络、多层感知网络、交替决策树 3 个分类器形成投票系统，共同判断流量异常。Tang 等[17]基于 SADBSCAN（self-adaptive density-based spatial clustering of applications with noise）算法提出了在多密度数据集中自适应识别聚类的解决方案。根据网络流量受到 RoQ 攻击的特点，对网络流量进行分组。然后使用余弦相似度来确定每个组中是否包含攻击数据。特征工程是基于人工智能技术的检测方法中重要的一环，为了提高攻击检测率，Tang 等[18]提取了 RoQ 攻击中包括信息熵在内的 28 个特征，并提出改进的 MF-Adaboost 算法，能够更准确地检测 RoQ 攻击。吴志军团队近几年着重研究了利用路由器队列管理机制漏洞的 RoQ 攻击的检测方法。2018 年，吴志军等[19]提取了路由器瞬时队列和平均队列作为数据特征，利用核主成分分析（KPCA, kernel principal component analysis）进行特征处理，并利用反向传播（BP, back propagation）神经网络进行训练和检测，效果较好。2020 年，该团队[20]提出了基于序列确认号（ACK, acknowledge number）、报文大小和队列长度的多特征融合的路由器降质攻击检测方法。每种特征分别输入 K 邻近（KNN, K-nearest neighbor）分类器得到综合的决策轮廓矩阵，并定义融合决策指标 D 作为检测 RoQ 攻击的依据，在一定程度上提高了检测率。利用路由器漏洞的攻击属于众多 RoQ 攻击的一种，在检测时比较容易提取路由器队列特征。而传输层和应用层的攻击因流量特征与正常流量区别很小，很难提取有效特征，隐蔽性很强。

综上，现有 RoQ 攻击检测方法大部分假阴性率较高，仅适用于小规模数据，或只能在仿真集中实现，在真实环境中应用有一定的局限性。目前，基于人工智能技术的检测方法研究进展较慢、实用性不强。大部分方法在训练模型时需要大量已知的训练样本支撑，这与缺少真实样本的现实情况相悖。其次，部分研究中的实验是基于 NS2 等模拟软件的，实验条件理想、数据单一，不具有说服力。此外，目前大部分研究没有考虑到 RoQ 攻击的伪装性，在海量流量中的识别性较差。

2 基础知识

2.1 谱聚类

谱聚类是一种聚类算法，相较于传统聚类算法，它对数据分布的适应性更强，聚类效果更好，计算量更小，在攻击检测中常常被用作聚类算法[21]。在对相似攻击行为的聚类方面，谱聚类比 K-means、（DBSCAN, density-based spatial clustering of applications with noise）等聚类算法轮廓系数更高，识别效果更好[22]。

谱聚类的主要思想是基于图谱分割理论，主要流程如下。

目标：输入 n 个数据点集，获得 k 个聚类。

1) 利用距离公式计算每个点集的相似度，形成相似矩阵。

2) 利用相似矩阵构建邻接矩阵 $\boldsymbol{N}$ 和度矩阵 $\boldsymbol{G}$ 。

3) 由步骤 2)得到的邻接矩阵 $\boldsymbol{N}$ 和度矩阵 $\boldsymbol{G}$ 计算得出拉普拉斯矩阵 $\boldsymbol{L}=\boldsymbol{G}^{\frac{1}{2}}\boldsymbol{N}\boldsymbol{G}^{\frac{1}{2}}$ 。

4) 选取拉普拉斯矩阵的前 k 个最大特征值对应的特征向量。

5) 标准化特征向量，形成 $k \times n$ 维特征矩阵。

6) 将特征向量的每一行作为一个 k 维的样本，共 n 个样本，使用 K-means 等聚类算法进行聚类。

2.2 Shapelet 时间子序列

2009 年，Keogh 等[23]在数据挖掘顶级会议上首次提出了时序数据中的 Shapelet 的概念。Shapelet 是时间序列的子序列，是相似时间序列中最不相同的一段序列，在区分相似度很高的序列方面稳健性更强，准确性更高。它可以较充分地说明各个类别之间的差异，使分类结果具有更强的可解释性。

Shapelet 子序列各概念定义如下。

定义 1　时间序列及子序列。时间序列 $T=(t_1,t_2,\cdots,t_n)$ 是按时间顺序采样的数据点构成的序列，$t_i(i\in 1,2,\cdots,n)$ 是任意实数。子序列 $S=(t_i,t_{i+1},\cdots,t_{i+k-1})$ 是从某一位置开始、长度为 k 的一段连续序列，$1\leqslant i\leqslant m-1+k$。

定义 2　时间序列的距离。将长度为 m 的 2 条时间序列 $\boldsymbol{A}=(a_1,a_2,\cdots,a_m)$ 和 $\boldsymbol{B}=(b_1,b_2,\cdots,b_m)$ 看作向量，它们之间的距离 $\mathrm{Dist}(\boldsymbol{A},\boldsymbol{B})=\|\boldsymbol{A}-\boldsymbol{B}\|_2=\left\{\sum_{i=1}^{m}|a_i-b_i|^2\right\}^{\frac{1}{2}}$。

定义 3　子序列和时间序列的距离。对于长度不同的子序列 S 和时间序列 T，距离定义为 S 与 T 中长度与 S 相同的子序列的距离的最小值，即 $\mathrm{SubseqDist}(S,T)=\min\left(\mathrm{Dist}(S,T_i)\right)$，$T_i$ 表示 T 中长度与 S 相同的所有子序列。

定义 4　信息增益。设数据集 D 被划分为数据子集 D_1 和 D_2，则其信息增益可表示为

$$\mathrm{IG}(D)=E(D)-\frac{n_1}{n}E(D_1)-\frac{n_2}{n}E(D_2) \tag{1}$$

其中，n、n_1 和 n_2 分别表示数据集 D、D_1 和 D_2 的大小。$E(D)$ 表示 D 的熵，表示为

$$E(D)=-\sum_{c\in\mathrm{class}\{D\}}p_c\lg p_c \tag{2}$$

其中，p_c 表示属于数据集 D 的元素 c 的概率。

定义 5　Shapelet 子序列。定义分裂点为一个二元组 $<S,\delta>$，由子序列 S 和距离阈值 δ 组成，根据 S 与数据集中每一条时间序列之间的距离是否大于 δ，将时间序列数据 D 分为 D_L 和 D_R，当信息增益最大时即 Shapelet，此时的距离阈值 $\delta=d_{\mathrm{osp}}$。

Shapelet 子序列在网络安全领域是一个较新的概念，但是已经在其他领域被用于时序异常检测和预测[24-25]。在现有的研究中，Shapelet 可以与图论[26]或神经网络[27]结合完成分类。

3 RoQ 隐蔽攻击无监督检测模型总体框架

本文提出的 RoQ 隐蔽攻击无监督检测模型包含 2 个阶段，分别使用了不同层面的流量特征。首先，利用流特征对流量进行初筛，筛除大量可能会干扰检测效果的正常流量，保留正常流量与异常流量相对均衡的样本。因此，要求筛选速度快、效率高，且筛除的正常流量的精确率接近 100%。其次，利用时序包特征，在没有先验知识的情况下，对筛选后的流量进行无监督精准分类，要求分类模型具有局部特征差异的辨识能力。RoQ 隐蔽攻击检测模型整体框架如图 1 所示。其中，网络流量通过数据平面开发套件（DPDK, data plane development kit）进行采集过滤，进入后续处理流程。

第一阶段为基于流的聚类检测算法，主要实现大量正常流量的筛除。该阶段的输入是去噪声、去冗余格式化后经过特征提取后的网络流特征，核心算法为改进的半监督谱聚类算法。为了提高正样本分类的精确度，利用极少数已知的攻击样本构造正样本强化监督信息。同时，为了最大化利用这些已知攻击样本，使谱聚类算法有最优的迭代次数限制，设计基于极少已知标签的算法调整机制，根据调整机制评价结果调整参数和聚类次数。

第二阶段为基于 n-Shapelet 子序列的无监督自学习算法，主要实现剩余流量的准确分类。该阶段的输入是按时间排列的数据包，称为时间包序列。首先，利用 Shapelet 子序列的特点构造 Shapelet 序列空间，该空间代表了样本间差距最大的局部特征。其次，将每一个时间包序列映射到 Shapelet 序列空间内，转换为 $\mathbb{S}$ 空间特征，使用聚类算法进行聚类。为了使分类结果更准确，构造无监督目标优化函数来调整算法参数。

4 基于流的聚类检测算法

通常，距离比较远的数据被认为是不同类的，而距离较近的数据则是同类的。但在一些特殊情况下，存在一些距离较大但属于不同类的、或者距离较小但属于同类的离群点需要被修正。因此，本节提出一种基于半监督谱聚类的正常流量筛除方法，尽可能保证筛除的正常流量的精确度接近 100%。在具体方法上包括 3 个部分：正样本强化监督信息

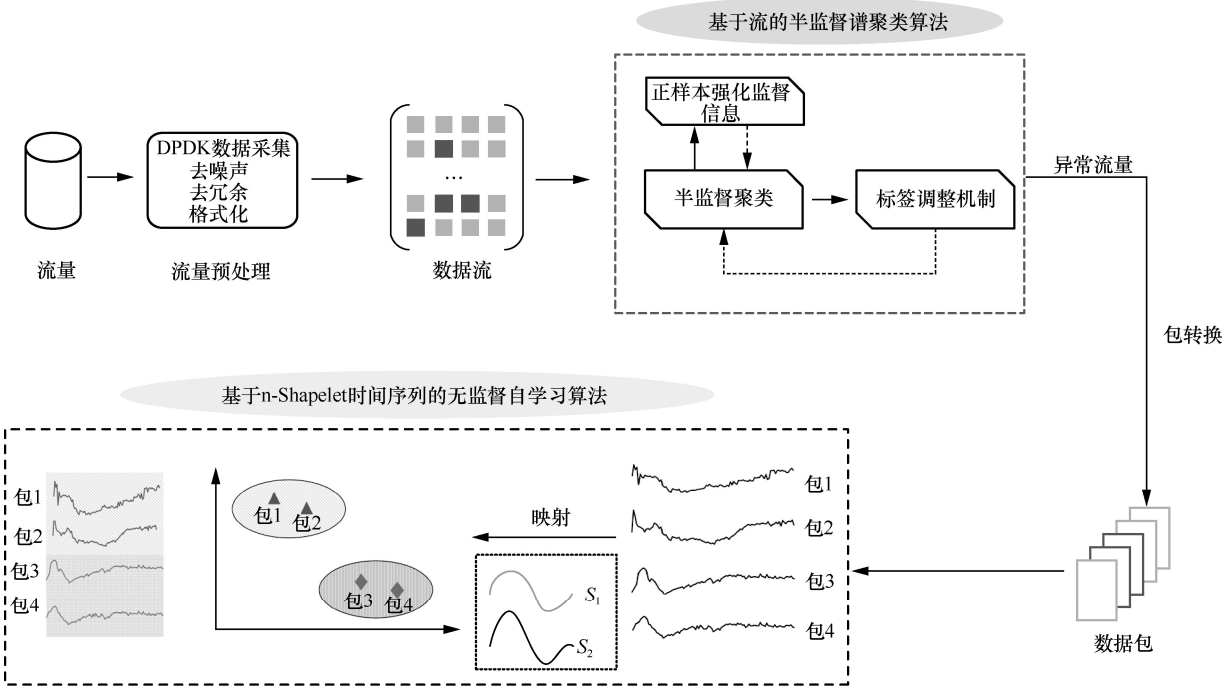


图 1　RoQ 隐蔽攻击检测模型整体框架

构造主要实现去除正样本侧的离群点；半监督谱聚类算法主要实现数据流的聚类；基于极少已知标签的算法调整机制主要实现聚类结果的合理评价。

4.1　正样本强化监督信息构造

构造正样本强化监督信息对，包括距离很大的正样本信息对、正样本与其他类型样本距离很小的信息对。具体方法如下。

1) 关注的正常类别。初始化监督信息限制数目 M，集合 A 为空，利用简单专家知识规则在 Q_{int} 中确定一定数量的正样本，加入集合 S 中；初始化当前监督信息数目 $m=0$。

2) 当 $m \leqslant \dfrac{M}{2}$ 时，选择距离集合 A 最远的点 x，即 $x \mid d(x,A)=\max(d_{y\in A}(x,y))$。询问 x 是否属于正样本，若属于，则随机抽取集合 A 中的一点 y，构建成对的约束监督信息集合 1，$m=m+1$。

3) 当 $\dfrac{M}{2}<m \leqslant M$ 时，选择距离集合 A 最近的点 x，即 $x \mid d(x,A)=\min(d_{y\in A}(x,y))$。询问 x 是否属于正样本，若不属于，则随机抽取集合 A 中的一点 y，构建成对的约束监督信息集合 2，$m=m+1$。

重复步骤 2)和步骤 3)，完成正样本监督信息集构造。监督信息集包括 2 个集合，集合 1 中都为正样本，但彼此距离较远；集合 2 中一个为正样本，另一个为其他类型样本，但距离很近。

4.2　半监督谱聚类算法

本文提出的改进后的半监督谱聚类算法具有对正样本修正的能力，具体算法 R 如下。

1) 计算待分类数据集中两点之间欧氏距离，记为 $\text{dist}_{ij}=\sqrt{(x_i-x_j)^2}$。形成距离矩阵 $\boldsymbol{D}$。

2) 基于正样本监督信息，修正正样本侧离群点。若 (x_i,x_j) 属于集合 1，则 $\text{dist}_{ij}=0$；若 (x_i,x_j) 属于集合 2，则 $\text{dist}_{ij}=\max(\max(\boldsymbol{D}_i),\max(\boldsymbol{D}_j))$。

3) 构建对角矩阵 $\boldsymbol{S}$，$\boldsymbol{S}_{ii}=\sum_{j=1}^{n}\text{dist}_{ij}$，标准化后拉普拉斯矩阵为 $\boldsymbol{P}=\boldsymbol{S}^{-\frac{1}{2}}(\boldsymbol{D}-\boldsymbol{S})\boldsymbol{S}^{\frac{1}{2}}$。

4) 选取 $\boldsymbol{P}$ 矩阵最小 e 个特征值对应的特征向量组成特征矩阵 $\boldsymbol{F}$，对 $\boldsymbol{F}$ 进行标准化后再聚类，得到聚类结果 $Q=\{q_0,q_1,\cdots,q_l\}$，q_0 为正样本的类别。

4.3　基于极少已知标签的算法调整机制

基于半监督谱聚类算法通过监督信息消除了正样本侧的离群点，提高了分类的精度。在聚类过程中，数据集带有极少已知的标签信息。为了提高这部分信息的利用程度，得到更精确的聚类分配结果，使算法结果可评价，收敛速度可控，本文提出基于极少已知标签的算法调整机制。

首先，定义 $p(y_i,q)$ 为样本 y_i 与类别 q 的相似度，也可以看作样本 y_i 属于类别 q 的概率。

$$p(y_i,q)=\frac{(1+\|y_i-\mu_{j=q}\|^2)}{\sum_{j=1}^{k}(1+\|y_i-\mu_j\|^2)} \tag{3}$$

算法调整机制具体方法如下。

1) 对数据集使用算法 R 进行首次聚类，得到聚类结果 $Q=\{q_0,q_1,\cdots,q_l\}$，统计每类结果中标签数据量最大的那一类 Maxnum($Q[a_i\in q_i]$)，作为该类的标签 c。对于已知标签的部分样本 $y=[l_1,l_2,\cdots,l_n]$，经过聚类后被划分为 $y'=[l,l_r,l_k,\cdots,l_h]$，$l_*$ 代表不同的类别，*为任意整数。

2) 使用标记列表 A' 标记正样本点的划分正确与否，标记列表 A' 中元素 a_i 的含义为

$$a_i=\begin{cases}1, y'_i\neq c\\ 0, y'_i=c\end{cases} \tag{4}$$

则已知标签的正样本聚类损失为

$$L_1=-\lambda\sum_{i=1}^{n}a_i\log p(y_i,c) \tag{5}$$

3) 使用标记列表 B' 标记负样本点的划分正确与否，标记列表 B' 中元素 b_i 的含义为

$$b_i=\begin{cases}1, z'_i=c\\ 0, z'_i\neq c\end{cases} \tag{6}$$

则已知标签的负样本聚类损失为

$$L_2=\sigma\sum_{i=1}^{m}b_i\log p(z_i,c) \tag{7}$$

则整体损失为

$$L=L_1+L_2=-\lambda\sum_{i=1}^{n}a_i\log p(y_i,c)+\sigma\sum_{i=1}^{m}b_i\log p(z_i,c) \tag{8}$$

5　基于 n-Shapelet 子序列的无监督自学习算法

第一阶段的初步分类旨在精准地筛除大部分正常样本，第二阶段将完成更准确的分类。不同于第一阶段基于流特征的方法，第二阶段将剩余的流还原成数据包，按照时间顺序排列形成时间序列，一条流表示为

$$\boldsymbol{T}=(t_1,t_2,\cdots,t_l)=\begin{pmatrix}t_{11}&t_{21}&\cdots&t_{l1}\\ t_{12}&t_{22}&\cdots&t_{l2}\\ \vdots&\vdots&\ddots&\vdots\\ t_{1n}&t_{2n}&\cdots&t_{ln}\end{pmatrix} \tag{9}$$

其中，l 表示时间序列的长度，n 表示每一个数据包的特征维度。

5.1　Shapelet 空间特征映射

为了找到局部更具有区分度的特征，本节利用 Shapelet 子序列完成特征提取。在待检测的流量中寻找到时序包序列中的若干个最优 Shapelet，构成 $\mathbb{S}$ 空间。将时间包序列投影到新空间，转换为新特征再进行检测。

根据 Shapelet 定义[23]，n 维包序列的 Shapelet 子序列表示为

$$\boldsymbol{S}=(s_1,s_2,\cdots,s_k)=\begin{pmatrix}s_{11}&s_{21}&\cdots&s_{k1}\\ s_{12}&s_{22}&\cdots&s_{k2}\\ \vdots&\vdots&\ddots&\vdots\\ s_{1n}&s_{2n}&\cdots&s_{kn}\end{pmatrix} \tag{10}$$

其中，$k(k<l)$ 为 Shapelet 子序列的长度，子序列的每个元素保持 n 个维度，用于表示序列短时间内（即长度 k 内）在所有维度上的综合变化特征。选取若干个 Shapelet 子序列，构造 $\mathbb{S}$ 空间为 $\mathbb{S}=\{\boldsymbol{S}_1,\boldsymbol{S}_2,\cdots,\boldsymbol{S}_v\}$。

定义时间序列 $\boldsymbol{T}$ 与子序列 $\boldsymbol{S}$ 的距离为

$$\begin{aligned}\mathrm{Dist}(\boldsymbol{T},\boldsymbol{S})&=\min_{p=0,1,\cdots,l-k}d_p(\boldsymbol{T},\boldsymbol{S})\\ &=\min_{p=0,1,\cdots,l-k}\sum_{i=1}^{k}\sum_{j=1}^{n}(t_{(p+i)j}-s_{ij})^2\end{aligned} \tag{11}$$

为了方便后续的最小化求解，距离函数连续化为

$$D(\boldsymbol{T},\boldsymbol{S})=\frac{\sum_{p=0}^{l-k}d_p(\boldsymbol{T},\boldsymbol{S})e^{\beta d_p(\boldsymbol{T},\boldsymbol{S})}}{\sum_{p=0}^{l-k}e^{\beta d_p(\boldsymbol{T},\boldsymbol{S})}} \tag{12}$$

根据距离公式，包序列集合 $\mathbb{T}=\{\boldsymbol{T}_1,\boldsymbol{T}_2,\cdots,\boldsymbol{T}_u\}$ 在 $\mathbb{S}=\{\boldsymbol{S}_1,\boldsymbol{S}_2,\cdots,\boldsymbol{S}_v\}$ 上的投影矩阵为

$$\boldsymbol{X}=\begin{pmatrix}D_{11}&D_{21}&\cdots&D_{u1}\\ D_{12}&D_{22}&\cdots&D_{u2}\\ \vdots&\vdots&\ddots&\vdots\\ D_{1v}&D_{2v}&\cdots&D_{uv}\end{pmatrix} \tag{13}$$

其中，$D_{uv}=D(\boldsymbol{T}_u,\boldsymbol{S}_v)$。

至此，原始包序列各特征投影到新空间形成新的特征矩阵 $\boldsymbol{X}$，新特征是在不同 Shapelet 子序列上的投影，具有最佳辨识度，可以使用聚类算法对 $\boldsymbol{X}$ 进行聚类。

5.2　无监督分类优化模型

$\mathbb{S}$ 空间的 Shapelet 子序列可以最大程度地学到

各个时间序列的最大区分局部特征，因此，需保证各个 Shapelet 子序列的相似度最低。定义 Shapelet 相似度矩阵 $\boldsymbol{H}\in\mathbb{R}^{v\times v}$，表示为

$$\boldsymbol{H}=\begin{pmatrix} H_{11} & H_{21} & \cdots & H_{v1} \\ H_{12} & H_{22} & \cdots & H_{v2} \\ \vdots & \vdots & \ddots & \vdots \\ H_{1v} & H_{2v} & \cdots & H_{vv} \end{pmatrix} \tag{14}$$

其中，$H_{ij}=\mathrm{Dist}(\boldsymbol{S}_i,\boldsymbol{S}_j)=\sum_{a=1}^{k}\sum_{b=1}^{n}(s_{ab}^{i}-s_{ab}^{j})^2$。

基于此，构造分类优化函数为

$$\min_{\boldsymbol{W},\mathbb{S},\boldsymbol{Y}}\frac{1}{2}\left\|\boldsymbol{W}^{\mathrm{T}}\boldsymbol{X}(\mathbb{S})-\boldsymbol{Y}\right\|_{\mathrm{F}}^{2}+\frac{\lambda_1}{2}\left\|\boldsymbol{H}(\mathbb{S})\right\|_{\mathrm{F}}^{2}+\frac{\lambda_2}{2}\left\|\boldsymbol{W}\right\|_{\mathrm{F}}^{2} \tag{15}$$

其中，$\boldsymbol{W}\in\mathbb{R}^{c\times v}$ 表示分类器的分类矩阵，$\boldsymbol{Y}$ 表示分类结果，第一次分类中 $\boldsymbol{Y}$ 将由普通的聚类得到；$\|\boldsymbol{W}^{\mathrm{T}}\boldsymbol{X}(\mathbb{S})-\boldsymbol{Y}\|_{\mathrm{F}}^{2}$ 表示分类结果的最小平方误差；$\|\boldsymbol{H}(\mathbb{S})\|_{\mathrm{F}}^{2}$ 表示 $\mathbb{S}$ 空间的每个 Shapelet 子序列相关性最小；$\|\boldsymbol{W}\|_{\mathrm{F}}^{2}$ 表示惩罚系数；λ_1 和 λ_2 为权重参数。

$\boldsymbol{X}$ 经过聚类之后得到分类结果 $\boldsymbol{Y}$，并根据分类优化函数对参数 $\boldsymbol{W}$ 和 $\mathbb{S}$ 进行迭代优化，迭代过程如下。

1) $\boldsymbol{Y}$ 的迭代

根据分类优化函数中的第 1 项

$$F(\boldsymbol{Y})=\frac{1}{2}\|\boldsymbol{W}^{\mathrm{T}}\boldsymbol{X}(\mathbb{S})-\boldsymbol{Y}\|_{\mathrm{F}}^{2} \tag{16}$$

对 $\boldsymbol{Y}$ 求导并令其为 0，有

$$\boldsymbol{Y}=\boldsymbol{W}^{\mathrm{T}}\boldsymbol{X}(\mathbb{S}) \tag{17}$$

故 $\boldsymbol{Y}_{t+1}=\boldsymbol{W}_t^{\mathrm{T}}\boldsymbol{X}(\mathbb{S}_t)$。

2) $\boldsymbol{W}$ 的迭代

根据分类优化函数中的第 1 项和第 3 项

$$F(\boldsymbol{W})=\frac{1}{2}\|\boldsymbol{W}^{\mathrm{T}}\boldsymbol{X}(\mathbb{S})-\boldsymbol{Y}\|_{\mathrm{F}}^{2}+\frac{\lambda_2}{2}\|\boldsymbol{W}\|_{\mathrm{F}}^{2} \tag{18}$$

对 $\boldsymbol{W}$ 求导并令其为 0，有

$$\boldsymbol{W}_{t+1}=\left[\boldsymbol{X}(\mathbb{S}_t)\boldsymbol{X}^{\mathrm{T}}(\mathbb{S}_{\mathrm{t}})+\lambda_2\right]^{-1}\left[\boldsymbol{X}(\mathbb{S}_t)\boldsymbol{Y}_{\mathrm{t+1}}^{\mathrm{T}}\right] \tag{19}$$

3) $\mathbb{S}$ 的迭代

根据分类优化函数中的第 1 项和第 2 项

$$F(\mathbb{S})=\frac{1}{2}\|\boldsymbol{W}^{\mathrm{T}}\boldsymbol{X}(\mathbb{S})-\boldsymbol{Y}\|_{\mathrm{F}}^{2}+\frac{\lambda_1}{2}\|\boldsymbol{H}(\mathbb{S})\|_{\mathrm{F}}^{2} \tag{20}$$

采取梯度下降法：$\mathbb{S}_{t+1}=\mathbb{S}_t-\eta\nabla F(\mathbb{S}_t)$，其中，$\eta$ 是给定的学习率，$\nabla F(\mathbb{S}_t)$ 是与 $\mathbb{S}_t$ 维度相同的矩阵集。

考虑 $\mathbb{S}=\{\boldsymbol{S}_1,\boldsymbol{S}_2,\cdots,\boldsymbol{S}_v\}$，其中，每个元素都是一个 k 列 n 行的矩阵，而 $\nabla F(\mathbb{S})$ 是一个含有 v 个元素的有序集，每个元素是一个 k 列 n 行的矩阵。因此，以每个元素为单位计算 $\nabla F(\mathbb{S})$ 的结果，以第 v 个矩阵的第 k 列 n 行的元素 $\frac{\partial F(\mathbb{S})}{\partial s_{kn}^{(v)}}$ 为例，设

$$\boldsymbol{W}=\begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1c} \\ w_{21} & w_{22} & \cdots & w_{2c} \\ \vdots & \vdots & \ddots & \vdots \\ w_{v1} & w_{v2} & \cdots & w_{vc} \end{pmatrix} \tag{21}$$

$$\boldsymbol{Y}=\begin{pmatrix} y_{11} & y_{21} & \cdots & y_{u1} \\ y_{12} & y_{22} & \cdots & y_{u2} \\ \vdots & \vdots & \ddots & \vdots \\ y_{1c} & y_{2c} & \cdots & y_{uc} \end{pmatrix} \tag{22}$$

其中，c 为类别数目。

将 $\boldsymbol{W}$、$\boldsymbol{Y}$、$\boldsymbol{X}$、$\boldsymbol{H}$ 代入 $F(\mathbb{S})$，展开得

$$\begin{aligned} F(\mathbb{S})=&\frac{1}{2}\sum_{1,2,\cdots,u}\sum_{1,2,\cdots,c}\Big(\sum_{1,2,\cdots,v}w_{vc}D_{uv}-y_{uc}\Big)^2+\\ &\frac{\lambda_1}{2}\sum_{1,2,\cdots,v}\sum_{1,2,\cdots,v}H_{v_1v_2}^{2} \end{aligned} \tag{23}$$

故

$$\begin{aligned} &\frac{\partial F(\mathbb{S})}{\partial s_{kn}^{(v)}}=\\ &\frac{\partial}{2\partial s_{kn}^{(v)}}\left[\sum_{1,2,\cdots,u}\sum_{1,2,\cdots,c}\left(\sum_{1,2,\cdots,v}w_{vc}D_{uv}-y_{uc}{}^{2}\right)\right]+\\ &\frac{\lambda_1\partial}{2\partial s_{kn}^{(v)}}\left[\sum_{1,2,\cdots,v}\sum_{1,2,\cdots,v}H_{v_1v_2}^{2}\right]=\\ &\sum_{1,2,\cdots,u}\sum_{1,2,\cdots,c}\frac{\partial}{2\partial s_{kn}^{(v)}}\left[\frac{1}{2}(w_{vc}D_{uv})^2\right]+\\ &\lambda_1\frac{\partial}{\partial s_{kn}^{(v)}}\left[\sum_{i=1}^{v-1}H_{iv}^{2}\right]=\\ &\sum_{1,2,\cdots,u}\sum_{1,2,\cdots,c}w_{vc}^{2}D_{uv}\frac{\partial D_{uv}}{\partial s_{kn}^{(v)}}+\lambda_1\sum_{i=1}^{v-1}H_{iv}\frac{\partial H_{iv}}{\partial s_{kn}^{(v)}} \end{aligned} \tag{24}$$

因此，只需求出每个 $\frac{\partial D_{uv}}{\partial s_{kn}^{(v)}}$ 与 $\frac{\partial H_{iv}}{\partial s_{kn}^{(v)}}$ 即可。

关于 $\frac{\partial D_{uv}}{\partial s_{kn}^{(v)}}$，已知

$$D_{uv}=D(\boldsymbol{T}_u,\boldsymbol{S}_v)=\frac{\sum_{p=0}^{l-k}d_p(\boldsymbol{T}_u,\boldsymbol{S}_v)e^{\beta d_p(\boldsymbol{T}_u,\boldsymbol{S}_v)}}{\sum_{p=0}^{l-k}e^{\beta d_p(\boldsymbol{T}_u,\boldsymbol{S}_v)}} \tag{25}$$

记 $D_{uv}=\frac{E_1}{E_2}$，则 $\frac{\partial D_{uv}}{\partial s_{kn}^{(v)}}=\frac{1}{E_2^2}(E_2\frac{\partial E_1}{\partial s_{kn}^{(v)}}-E_1\frac{\partial E_2}{\partial s_{kn}^{(v)}})$。

其中，

$$\frac{\partial E_1}{\partial s_{kn}^{(v)}}=\frac{\partial}{\partial s_{kn}^{(v)}}\left(\sum_{p=0}^{l-k}d_p(\boldsymbol{T}_u,\boldsymbol{S}_v)e^{\beta d_p(\boldsymbol{T}_u,\boldsymbol{S}_v)}\right)=$$
$$\sum_{p=0}^{l-k}\left(e^{\beta d_p(\boldsymbol{T}_u,\boldsymbol{S}_v)}\frac{\partial}{\partial s_{kn}^{(v)}}d_p(\boldsymbol{T}_u,\boldsymbol{S}_u)+d_p(\boldsymbol{T}_u,\boldsymbol{S}_u)\right.$$
$$\left.\frac{\partial}{\partial s_{kn}^{(v)}}e^{\beta d_p(\boldsymbol{T}_u,\boldsymbol{S}_v)}\right)=\sum_{p=0}^{l-k}2e^{\beta d_p(\boldsymbol{T}_u,\boldsymbol{S}_v)}$$
$$[1+\beta d_p(\boldsymbol{T}_u,\boldsymbol{S}_v)]\left(s_{kn}^{(v)}-t_{(p+k)n}^{(u)}\right) \tag{26}$$

$$\frac{\partial E_2}{\partial s_{kn}^{(v)}}=\sum_{p=0}^{l-k}\beta e^{\beta d_p(\boldsymbol{T}_u,\boldsymbol{S}_v)}\frac{\partial}{\partial s_{kn}^{(v)}}d_p(\boldsymbol{T}_u,\boldsymbol{S}_u)=$$
$$\sum_{p=0}^{l-k}2\beta e^{\beta d_p(\boldsymbol{T}_u,\boldsymbol{S}_v)}\left(s_{kn}^{(v)}-t_{(p+k)n}^{(u)}\right) \tag{27}$$

关于$\frac{\partial H_{iv}}{\partial s_{kn}^{(v)}}$，由式(23)得出

$$\frac{\partial H_{iv}}{\partial s_{kn}^{(v)}}=\frac{\partial}{\partial s_{kn}^{(v)}}\left[\sum_{a=1}^{k}\sum_{b=1}^{n}\left(s_{ab}^{(i)}-s_{ab}^{(v)}\right)^2\right]=$$
$$\frac{\partial}{\partial s_{kn}^{(v)}}\left(s_{kn}^{(i)}-s_{kn}^{(v)}\right)^2=2\left(s_{kn}^{(i)}-s_{kn}^{(v)}\right) \tag{28}$$

根据以上计算得到的$\frac{\partial D_{uv}}{\partial s_{kn}^{(v)}},\frac{\partial H_{iv}}{\partial s_{kn}^{(v)}},\frac{\partial F(\mathbb{S})}{\partial s_{kn}^{(v)}}$，最终完成$\mathbb{S}$的迭代更新。

5.3　时间复杂度评估

在无监督学习和推理过程中，经过参数的初始化后，目标函数需要多次迭代直至收敛。每次迭代过程都包括 2 个矩阵($\boldsymbol{X},\boldsymbol{H}$)的计算阶段以及 3 个参数矩阵($\boldsymbol{Y},\boldsymbol{W},\boldsymbol{S}$)的更新阶段。根据 5.2 节的定义，包序列数据集$\mathbb{T}=\{T_1,T_2,\cdots,T_u\}$，一条时间序列$T$长度为$l$，每个包有$n$维特征；$\mathbb{S}$空间 Shapelet 子序列长度为$k$，大小为$v$；聚类类别为$K$。那么，在矩阵$\boldsymbol{X},\boldsymbol{H}$的计算阶段，根据式(13)和式(14)可得计算$\boldsymbol{X}$的时间复杂度为$O(uvkln)$，计算$\boldsymbol{H}$的时间复杂度为$O(v^2kn)$。在参数矩阵($\boldsymbol{Y},\boldsymbol{W},\boldsymbol{S}$)的更新阶段，根据式(16)可得更新参数矩阵$\boldsymbol{Y}$的时间复杂度为$O(Kuv)$，根据式(18)可得更新参数矩阵$\boldsymbol{W}$的时间复杂度为$O(v^2u+vuK+v^2K+v^3)$，根据式(24)~式(28)可得更新参数矩阵的时间复杂度为$O(uvlk^2n^2K+v^2kn)$。因此，无监督聚类算法的总体时间复杂度为$O\left(I(uvkln+v^2kn+Kuv+v^2u+v^2K+v^3+uvlk^2n^2K)\right)$，$I$是迭代次数，又因为参数中$k\ll l,v\ll u,K\ll u$，因此，时间复杂度可表示为$O\left(I(uln^2)\right)$。

6　实验分析

6.1　实验环境及数据说明

由于目前开源数据集中没有 TCP 拥塞控制等攻击样本，且在真实网络数据中也较难捕捉到这类攻击，因此，实验搭建了简单的模拟环境，模拟部分攻击来采集攻击数据。实验环境配置如表 1 所示。

为了从不同角度验证本文方法的合理性，实验基于模拟场景使用开源数据集，模拟攻击流量和真实网络流量构造了多种数据集。攻击类型覆盖了利用 HTTP 自适应机制的攻击、新型隐蔽攻击、逃避攻击等多种组合。

本文实验中使用的部分开源数据集来自 ISCX-SlowDoS-2016[5]、CIC-DDoS-2019[28]、CIRA-CIC-DoHBrw-2020[29]，详细描述如表 2 所示。ISCX-SlowDoS-2016 数据集包含了应用层各种 DoS 攻击，包含大量的应用层的小容量隐蔽攻击。CIC-DDoS-2019 数据集中包含了 WebDoS、Port Scan 等多源的 DDoS 变种隐蔽攻击。CIRA-CIC-DoHBrw-2020 中的恶意流量包含了针对网站的隐蔽攻击样本。这 3 个数据集均提供了原始的包捕获文件（.pcap 文件），数据完整。

6.2　特征提取与参数设置

6.2.1　流量特征提取

本文方法在实验过程中需要提取 2 个层面的

表 1　　**实验环境配置**

设备	配置
攻击主机	Windows 10 系统，处理器 Intel(R) Core (TM)，i7-6700，CPU @ 3.40 GHz，8 GB RAM
正常主机	Windows 10 系统，处理器 Intel(R) Core (TM)，i7-6498DU，CPU @ 2.50GHz，8 GB RAM
捕获主机	Windows10 系统，处理器 Intel(R) Core (TM)，i7-9700，CPU @ 3.00 GHz，8 GB RAM，配置 Wireshark 流量捕获软件
目标服务器	Apache，Windows Server 7，处理器 Intel(R) Core (TM)，i7-6700，CPU @ 3.40GHz，8 GB RAM
图形工作站	Windows 10 系统，处理器 Intel(R) Xeon(R) Gold 5218，CPU @ 2.30 GHz（2 颗），内存 128 GB，显卡 NVIDIA Quadro RTX 5000

表 2　本文实验使用的数据集详情

序号	数据集	异常/正常比例		异常/正常比例		协议	描述
		包	流	包	流		
1	Slowheaders	1:100	1:40	6728507	3950	HTTP	ISCX-SlowDoS-2016 数据集中 Slowheaders 攻击流量与 CIRA-CIC-DoHBrw-2020 数据集正常流量的混合
2	WebDoS	1:35	1:10	6852052	4931	HTTP	CIC-DDoS-2019 数据集中 WebDoS 攻击流量与 CIRA-CIC-DoHBrw-2020 数据集正常流量的混合
3	DoS slowloris	1:20	1:18	9072873	8808	HTTP	ISCX-SlowDoS-2016 数据集中 Slowbody2 攻击流量与 CIRA-CIC-DoHBrw-2020 数据集正常流量的混合
4	TCP-Congestion DoS	1:129	1:50	6711428	8433	TCP	实验环境中利用 TCP 拥塞控制机制的攻击流量和真实网络正常流量的混合
5	Router LDoS	1:100	1:30	8741118	4785	IP	实验环境中利用路由器队列机制的攻击流量和真实网络正常流量的混合
6	混合数据 1	1:100	1:100	7162332	10188	全栈	1、2、3 的混合流量
7	混合数据 2	1:100	1:100	7852130	8795	全栈	1、2、3、4、5 的混合流量
8	隐蔽攻击 1[30]	1:1000	1:1000	420000	3500	TCP	侧信道攻击

特征。首先是基于流特征的初步筛选，需要在原始流量中提取一段时间内流量的宏观特征。其次是基于时间包序列的精确分类，需要将剩余流量还原成按照时间排列的包序列，提取每一个包的具体特征。流特征和包特征描述分别如表 3 和表 4 所示。

为了能够直观看出多层次特征使用的意义，本文选取部分具有代表性的数值化特征绘制了对比

表 3　流特征描述

特征名称	特征描述
数据包总量	一段时间内数据包总数量
包数量最大值	一定时间间隔形成流中包数量的最大值
包数量最小值	一定时间间隔形成流中包数量的最小值
包数量差值	一定时间间隔形成流中包数量的最大值与最小值的差值
频域	一段时间内数据包数量在频域上的转换
源 IP 地址信息熵	一段时间内数据包的源 IP 地址的信息熵
源端口信息熵	一段时间内数据包的源端口的信息熵
目的端口信息熵	一段时间内数据包的目的端口的信息熵
包长度信息熵	一段时间内数据包长度的信息熵
时间信息熵	一段时间内数据包之间时间间隔信息熵
IP 分离率	源 IP 和目的 IP 信息熵特征的分离率
端口分离率	源端口和目的端口信息熵特征的分离率

表 4　包特征描述

特征名称	特征描述
源 IP	一个数据包内源 IP 地址
目的 IP	一个数据包内目的 IP 地址
目的端口	一个数据包内目的端口
协议	一个数据包内协议
包长度	一个数据包的长度
TCP 窗口	TCP 窗口大小
序列号	TCP 序列号
前序包差值	与前一个数据包大小的差值

曲线。本文方法第一阶段的正常流量与攻击流量的部分流特征对比如图 4 所示，从图 4 中可以看出，流特征在正常流量和攻击流量中有明显差异，能够使用聚类算法完成大部分流量的分类。

本文方法第 2 个阶段的正常流量与攻击流量的部分包特征对比如图 5 所示。从图 5(a)和图 5(b)中可以看出，正常流量和攻击流量的独立的包特征差别较小，很难用可视化的图分辨。从图 5(c)中可以看出，正常流量和攻击流量的包的前序包特征有较明显的差别。因此，在这一阶段提取时间序列特征对流量进行分类比较有效。

6.2.2　参数选择

本文方法 2 个阶段的算法需要确定超参。在第一阶段中，聚类类别数 K 与拉普拉斯矩阵的最小特征向量个数 e 取值一致，均可以根据实验数据集本身的标签获得。监督信息的数量 M 则通过正样本的聚类纯度 P 来确定。

监督信息数选择如表 5 所示。从表 5 中可以看出，

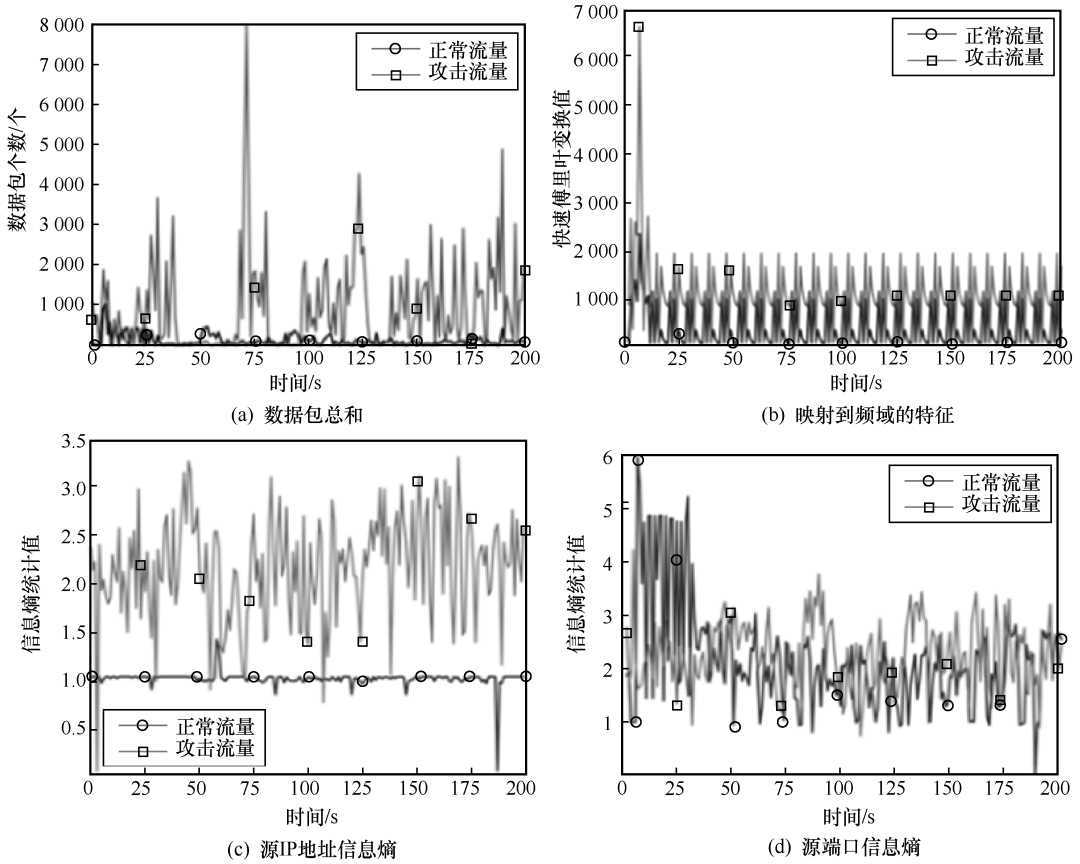


图 4　正常流量与异常流量的部分流特征对比

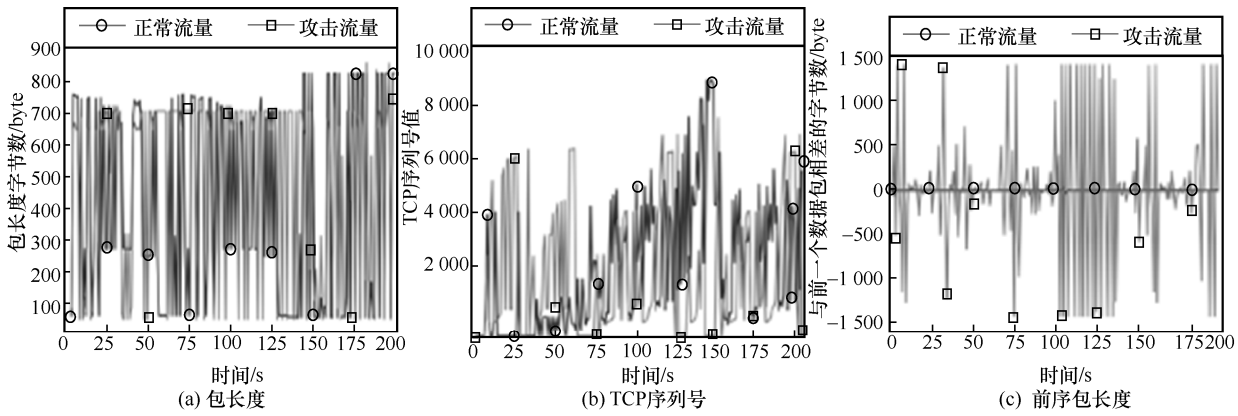


图 5　正常流量与异常流量的部分包特征对比

有 0.5%的监督信息比无监督信息精确率提高了 4.4%。为了使实验结果有明显区分度，选择 0.5%为增长间隔。当监督信息占比进一步增加至 1%时，精确率提升至 99.7%。从表 5 后两行可以看出，随着监督信息的继续增加，精确率并没有明显提升，当监督信息占比增长至 2%时，精确率只比 1%时提升了 0.1%，回报率低。因此，监督信息占比 1%为最佳平衡点，可在尽量少的监督信息下获得尽量高的精确度。

第二阶段中，$\mathbb{S}$ 空间的大小即 Shapelet 子序列的长度 k 和 v - $\mathbb{S}$ 空间中 Shapelet 的个数 v 为待确定的参数，通过聚类纯度 P 、兰德系数（RI, Rand index）及 F1 值确定。

表 5　　监督信息数选择

监督信息在总数据中的占比	聚类纯度 P
0%（M=0）	94.1%
0.5%（M=400）	98.5%
1%（M=800）	99.7%
1.5%（M=1 200）	99.6%
2%（M=1 600）	99.8%

在 TCP-Congestion Dos 数据集中，数据的长度范围为 100～4 000，参考 Zhang 等[31]实验中的参数选择，本文实验分别选取最短序列长度的 5%、8%、10%、15%、20%进行对比。需要说明的是，经过前期的实验，当 Shapelet 子序列长度 k 选取 5%以下时，信息量过少会影响区分度，因此最短序列长度从 5%开始。为了方便实验，选择 Shapelet 的个数为 2，不同 Shaplet 长度的性能表现如表 6 所示。从表 6 中可以看出，在 k 从 5%增长到 20%的过程中，性能呈现先上升后下降的趋势，可以推断，在 TCP-Congestion Dos 数据集中，最具辨识度的 k 在 5%~20%，也说明在这个区间内存在局部最优解。进一步分析，k 在 10%~15%时 P、RI 和 F1 值均较高，因此最优的 Shapelet 子序列长度应在 10%~15%。为了计算方便，后续实验选取最短序列长度的 10%为 k 的最优值。

表 6　　不同 Shapelet 长度的性能表现

k	P	RI	F1
5%	88.73%	81.29%	84.57%
8%	93.75%	84.67%	87.77%
10%	93.26%	89.00%	91.60%
15%	92.71%	88.00%	90.82%
20%	83.94%	77.36%	77.76%

$\mathbb{S}$ 空间中 Shapelet 的个数 v 的选择同样重要，v 过大会影响计算速度，过小又会使区分度不够明显。因此，实验分别选取 v 为 2、3、4、5、6、7、8，在计算效率可接受的情况下选取最优值。

不同 Shapelet 个数的性能表现如表 7 所示，从表 7 中可以看出，随着 v 的增加，聚类纯度 P 总体呈上升趋势，在 RI 和 F1 值上的表现也相对较好。综合考虑性能和计算速度，v =3 是本文实验的局部最优取值。同时需要说明的是，经过实验发现，当 v =8 时，所用时间是 v =3 时的 2 倍，因此，不再考虑 v 超过 8 的情况。综上，Shapelet 个数 v 最佳取值为 3。

表 7　　不同 Shapelet 个数的性能表现

v	P	RI	F1
2	93.26%	89.00%	91.60%
3	**94.39%**	**91.33%**	**93.43%**
4	93.89%	90.83%	93.06%
5	**94.74%**	90.00%	92.31%
6	93.91%	91.00%	93.20%
7	90.25%	80.24%	82.71%
8	**94.82%**	91.00%	93.13%

至此，本文方法的超参分别确定为：监督信息占比为 1%，Shapelet 长度为最短序列长度的 10%，Shapelet 空间子序列个数为 3。

6.3　实验结果分析

6.3.1　检测性能评估

为了验证本文方法的先进性，实验选取了部分具有代表性的无监督异常/攻击检测方法进行对比。这些方法主要分为以下三类：第一类为经典的基于原始流量的经典无监督聚类方法——K-means；第二类为专门针对隐蔽攻击的检测方法，包括自动编码器（Autoencoder）[32]、Whisper[33]模型；第三类为基于 Shapelet 的无监督检测方法，如 u-Shapelet[23]等。

由于 K-means 和 u-Shapelet 没有针对样本不平衡的处理，本文中的数据集正负样本数量差距大，直接使用会影响后续结果的计算，因此这 2 种方法使用的是经过本文方法第一阶段处理筛除大量正常流量后的数据。具体的对比结果如表 8~表 10 所示。

首先，从表 8~表 10 的第一列和第二列中可以看出，K-means 对于 RoQ 隐蔽攻击基本起不到聚类的作用，平均聚类纯度在 50%左右，相当于随机归类。Autoencoder 对于 Slowheaders、DoS slowloris

等特征较明显的攻击检测效果较好。同样地，在混合数据 1 上的表现也比较好，这是因为混合数据 1 中含有大量 Slowheaders、DoS slowloris 攻击数据。除此之外，Autoencoder 在其他数据集上表现一般。

其次，从表 8~表 10 的后三列中可以看出，Whisper、u-Shapelet 和本文方法各具优势。Whisper 最主要的特点是将时域数据转换成频域数据，因为攻击者不能轻易干扰频域特征，所以即使是手段复杂隐蔽的攻击在转换为频域特征后也具有一定的识别率。从实验结果上来看，在攻击比较隐蔽的混合数据 2 和隐蔽攻击 1 数据集中，Whisper 的聚类纯度最高；在频域特征最明显的 DoS slowloris 数据集中，Whisper 的 F1 值最高。这说明该方法有较强的隐蔽攻击识别能力，并且可以看出频域特征是该算法所依赖的一个重要特征。

u-Shapelet 只适合一维时间序列，因此在实验时只选取了包大小随时间的变化特征。从实验结果来看，在 5 种方法中，u-Shapelet 在 Router LDoS 数据集中聚类纯度最高，在 TCP-Congestion DoS 中 RI 和 F1 值最高，在其他数据集上的表现也相对均衡。这表明使用 Shapelet 子序列辨识流量中的变化是可行的。

本文方法在 4 个数据集上具有最优的聚类纯度，在 5 个数据集上具有最优的 RI 和 F1 值。对比

表 8　不同方法在不同数据集上聚类纯度 ***P*** 的性能表现

数据集	K-means	Autoencoder	Whisper	u-Shapelet	本文方法
Slowheaders	50.00%	85.44%	92.98%	90.79%	**95.11%**
Slowbody2	50.00%	74.23%	93.67%	92.84%	**94.51%**
DoS slowloris	59.56%	82.42%	94.45%	92.80%	94.18%
TCP-Congestion DoS	58.82%	78.47%	90.88%	94.36%	**94.39%**
Router LDoS	50.55%	68.04%	86.67%	**89.51%**	88.71%
混合数据 1	60.19%	83.78%	91.78%	92.10%	**92.74%**
混合数据 2	55.45%	66.17%	**88.87%**	88.24%	86.96%
隐蔽攻击 1	39.36%	76.73%	**87.49%**	83.33%	87.25%

表 9　不同方法在不同数据集上兰德系数 **RI** 的性能表现

数据集	K-means	Autoencoder	Whisper	u-Shapelet	本文方法
Slowheaders	45.45%	86.18%	**92.52%**	89.77%	92.24%
Slowbody2	45.45%	75.36%	91.87%	91.70%	**93.41%**
DoS slowloris	57.27%	84.07%	**94.23%**	87.64%	91.73%
TCP-Congestion DoS	55.00%	73.16%	87.82%	**93.32%**	91.33%
Router LDoS	45.91%	65.73%	85.95%	87.59%	**89.09%**
混合数据 1	55.45%	85.23%	90.98%	93.25%	**93.64%**
混合数据 2	50.45%	65.09%	83.55%	69.09%	**84.09%**
隐蔽攻击 1	36.36%	73.14%	85.39%	70.91%	**86.20%**

表 10　不同方法在不同数据集上 **F1** 值的性能表现

数据集	K-means	Autoencoder	Whisper	u-Shapelet	本文方法
Slowheaders	50.00%	87.66%	**93.16%**	90.61%	92.70%
Slowbody2	47.37%	78.81%	92.45%	92.35%	**93.92%**
DoS slowloris	61.83%	86.04%	**94.72%**	88.09%	92.26%
TCP-Congestion DoS	58.33%	73.99%	88.55%	**93.84%**	93.43%
Router LDoS	47.92%	69.05%	87.20%	88.50%	**90.16%**
混合数据 1	56.52%	86.972%	91.72%	93.93%	**94.26%**
混合数据 2	52.83%	69.72%	84.11%	63.83%	**95.11%**
隐蔽攻击 1	42.53%	74.73%	86.44%	68.63%	**87.37%**

Whisper，本文方法在时序特征的分辨上更细致，因此在检测时序特征明显的隐蔽攻击时表现更好。对比 u-Shapelet，本文方法使用了流量中更多的有效特征，检测性能更好。综上，本文方法在针对 RoQ 隐蔽攻击的检测时最具优势。

谱聚类算法筛除了大部分正常流量，仅留下比较难以识别的少量流量使进入第二阶段。修改后的基于正样本的半监督谱聚类算法进一步提高了筛选精度。为了验证本文提出的半监督谱聚类的优势，实验分别选取了经典的聚类方法 K-means、围绕中心点的划分（PAM, partitioning around medoid）及 DBSCAN 替换谱聚类进行实验对比，并依旧保留了原本的半监督机制，分别记为方法 A、方法 B 和方法 C。实验数据选择混合数据 1，结果如表 11 所示。从表 11 中可以看出，方法 A 方法、B 方法、C 的性能逐渐提升，这是由于 PAM 在获取新的聚类中心时策略较 K-means 更佳。基于密度聚类的方法 C 的召回率能达到 94.17%，说明了 RoQ 攻击流量在全流量中的分布也具有密度稀疏的特点。综合来看，本文方法总体性能表现优异。

表 11　　不同聚类方法的性能表现

方法	精确率	RI	召回率	F1
方法 A	88.98%	88.86%	90.86%	89.90%
方法 B	89.15%	89.09%	91.08%	90.11%
方法 C	90.69%	91.55%	94.17%	92.40%
本文方法	**92.74%**	**93.64%**	**95.83%**	**94.26%**

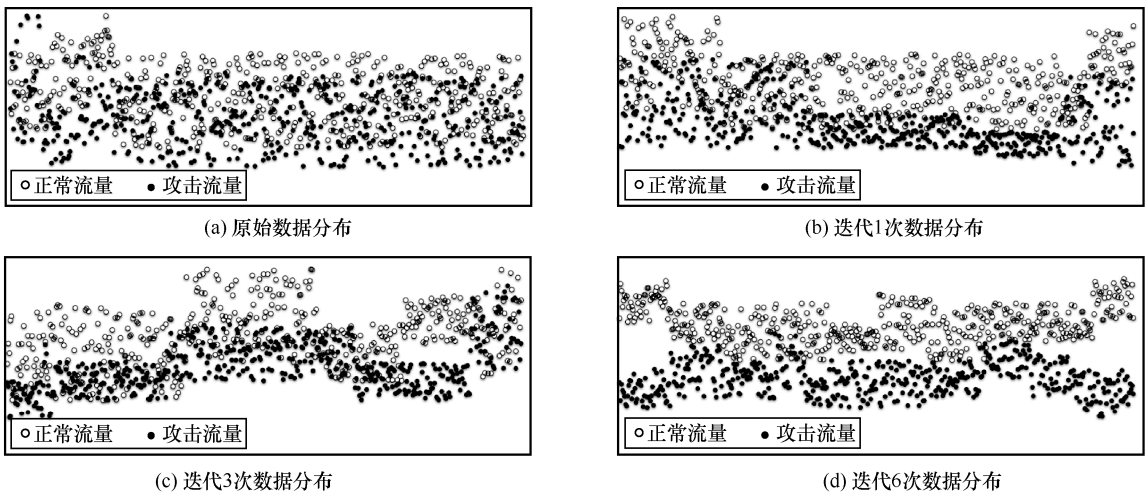
为了更加形象地表示本文方法的学习过程，实验在 Slowheaders 数据集上将分类过程进行了数值可视化，如图 6 所示。从图 6 可以看出，随着迭代次数的增多，分类效果越明显，当迭代次数达到 6 次时，已经可以很明显地区分出正常流量和攻击流量。

6.3.2　稳健性评估

本节将验证本文方法的稳健性。实验假设攻击者知道恶意流量检测的存在，可以构造规避攻击，即通过注入各种良性流量将恶意流量伪装成正常流量来逃避检测。

在实验中，选择 5 种恶意流量模式，并混入不同比例的良性流量，恶性流量和良性流量的比例在 1：1 到 1：8 之间。表 12 显示了不同方法在不同比例（恶意流量/良性流量）时对 5 种攻击的（AUC, area under curve）值。

从表 12 中可知，K-means 的稳健性较差。在 TCP-Congestion DoS 攻击采取不同注入策略时，AUC 值最多降低了 0.694，在大部分攻击使用不同策略伪装后检测性能不稳定，部分 AUC 值甚至低于 0.5。对比 K-means，Autoencoder 受规避攻击的影响较小。在 Slowheaders 伪装攻击上的 AUC 值最多降低了 0.229，推测这与自动编码器本身在特征方面的处理优势有关。u-Shapelet 从检测结果来看，当良性和恶意流量为 1:1 时，AUC 值较高，能够达到 0.879。当良性和恶意流量比例逐渐增加时，在 WebDoS、Slowbody2 和 Router LDoS 这 3 种规避攻击检测中有比较明显的降低，最多降低了 0.311；但对于另外两类规避类攻击稳健性较好，可以推测 Shapelet 子序列起到了关键作用。对比上述 3 种方法，本文方法在 Router LDoS 规避攻击中 AUC 值最多下降了 0.136，在 Slowheaders 规避攻击和 WebDoS 规避攻击中保持了较高且较稳定 AUC 值。以上结果说明本文方法稳健性较强，且本文方法



(a) 原始数据分布　(b) 迭代1次数据分布

(c) 迭代3次数据分布　(d) 迭代6次数据分布

图 6　分类过程可视化

表 12　不同方法在不同比例（恶意流量/良性流量）时对 5 种攻击的 AUC 值

攻击	比例	本文方法	K-means	Autoencoder	u-Shapelet
Slowheaders+TLS	1:1	0.912	0.781	0.771	0.870
	1:2	0.875	0.625	0.840	0.770
	1:4	0.930	0.521	0.763	0.780
	1:8	0.890	0.623	0.611	0.758
Slowbody2+TLS	1:1	0.903	0.710	0.840	0.810
	1:2	0.925	0.520	0.700	0.823
	1:4	0.842	0.421	0.860	0.782
	1:8	0.894	0.628	0.611	0.691
WebDoS+TLS	1:1	0.940	0.500	0.861	0.851
	1:2	0.952	0.880	0.755	0.720
	1:4	0.891	0.785	0.880	0.761
	1:8	0.910	0.341	0.810	0.650
TCP-Congettion DoS+TLS	1:1	0.897	0.823	0.779	0.879
	1:2	0.914	0.575	0.659	0.875
	1:4	0.880	0.129	0.762	0.840
	1:8	0.814	0.620	0.830	0.852
Router LDoS+TLS	1:1	0.896	0.680	0.681	0.871
	1:2	0.850	0.700	0.700	0.813
	1:4	0.760	0.355	0.813	0.660
	1:8	0.793	0.400	0.790	0.560

的第一阶段基于半监督谱聚类的正常流量筛除起到了重要的作用。

图 7 为不同方法的 AUC 平均值对比，从图 7 中可以看到，本文方法的 AUC 平均值最高，K-means 最低。在 WebDoS 攻击检测中本文方法分类能力最突出。

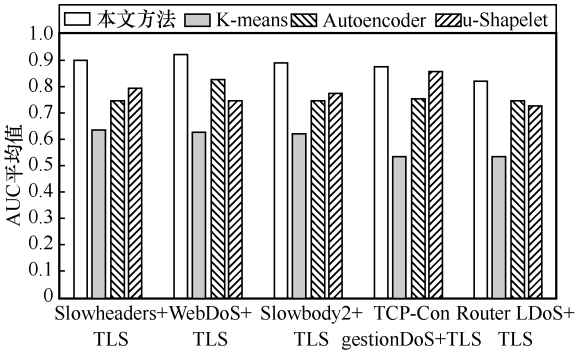


图 7　不同方法的 AUC 平均值对比

7　结束语

本文针对 RoQ 攻击特征不明显，高质量学习样本稀少等问题，提出了一种基于多层次特征的无监督隐蔽攻击检测方法。首先研究了基于半监督谱聚类算法，利用第一层流特征实现了正常流量高精度筛除，减少了对后续结果的干扰。其次，基于第二层时序包特征构造了基于 n-Shapelet 子序列的无监督攻击检测模型，实现了 RoQ 攻击的高检测率。实验结果表明，相较于现有方法，本文方法能够保持较高的检测性能，且对规避攻击具有稳健性。但是，本文方法的无监督学习模型中有 3 个超参，这 3 个超参的迭代计算增加了整体方法的复杂度。因此，未来将研究无监督学习模型，使其达到更快的收敛速度。

参考文献：

[1] GUIRGUIS M, THARP J, BESTAVROS A, et al. Assessment of vulnerability of content adaptation mechanisms to RoQ attacks[C]//Proceedings of the 8th International Conference on Networks. Piscataway: IEEE Press, 2009: 445-450.

[2] GUIRGUIS M, BESTAVROS A, MATTA I. Exploiting the transients of adaptation for RoQ attacks on Internet resources[C]//Proceedings of the 12th IEEE International Conference on Network Protocols. Piscataway: IEEE Press, 2004: 184-195.

[3] LUO X P, CHANG R K C. On a new class of pulsing denial-of-service attacks and the defense[C]//Proceedings of the NDSS Symposium 2005. Piscataway: IEEE Press, 2005: 1-19.

[4] GUIRGUIS M, BESTAVROS A, MATTA I, et al. Reduction of quality (RoQ) attacks on dynamic load balancers: vulnerability assessment and design tradeoffs[C]//Proceedings of the 26th IEEE International Conference

on Computer Communications. Piscataway: IEEE Press, 2007: 857-865.

[5] JAZI H H, GONZALEZ H, STAKHANOVA N, et al. Detecting HTTP-based application layer DoS attacks on Web servers in the presence of sampling[J]. Computer Networks, 2017, 121: 25-36.

[6] YUE M, WANG M X, WU Z J. Low-high burst: a double potency varying-RTT based full-buffer shrew attack model[J]. IEEE Transactions on Dependable and Secure Computing, 2019, 18(5): 2285-2300.

[7] VACCARI I, AIELLO M, CAMBIASO E. SlowITe, a novel denial of service attack affecting MQTT[J]. Sensors, 2020, 20(10): 2932.

[8] MERGET R, SOMOROVSKY J, AVIRAM N, et al. Scalable scanning and automatic classification of TLS padding oracle vulnerabilities[C]//Proceedings of the 28th USENIX Conference on Security Symposium. Berkeley: USENIX Association, 2019: 1029-1046.

[9] CHEN Y, HWANG K. Collaborative detection and filtering of shrew DDoS attacks using spectral analysis[J]. Journal of Parallel and Distributed Computing, 2006, 66(9): 1137-1151.

[10] AGRAWAL N, TAPASWI S. Low rate cloud DDoS attack defense method based on power spectral density analysis[J]. Information Processing Letters, 2018, 138: 44-50.

[11] 吴志军, 裴宝崧. 基于小信号检测模型的 LDoS 攻击检测方法的研究[J]. 电子学报, 2011, 39(6): 1456-1460.
WU Z J, PEI B S. The detection of LDoS attack based on the model of small signal[J]. Acta Electronica Sinica, 2011, 39(6): 1456-1460.

[12] TANG D, CHEN K, CHEN X S, et al. A new detection method based on AEWMA algorithm for LDoS attacks[J]. Journal of Networks, 1969, 9(11): 2981.

[13] TANG D, DAI R, TANG L, et al. Low-rate DoS attack detection based on two-step cluster analysis[C]//Information and Communications Security. Berlin: Springer, 2018: 92-104.

[14] TANG D, DAI R, TANG L, et al. Low-rate DoS attack detection based on two-step cluster analysis and UTR analysis[J]. Human-Centric Computing and Information Sciences, 2020, 10(1): 1-20.

[15] WU Z J, ZHANG L Y, YUE M. Low-rate DoS attacks detection based on network multifractal[J]. IEEE Transactions on Dependable and Secure Computing, 2016, 13(5): 559-567.

[16] KOAY A, CHEN A, WELCH I, et al. A new multi classifier system using entropy-based features in DDoS attack detection[C]//Proceedings of 2018 International Conference on Information Networking (ICOIN). Piscataway: IEEE Press, 2018: 162-167.

[17] TANG D, ZHANG S Q, CHEN J W, et al. The detection of low-rate DoS attacks using the SADBSCAN algorithm[J]. Information Sciences, 2021, 565: 229-247.

[18] TANG D, TANG L, DAI R, et al. MF-Adaboost: LDoS attack detection based on multi-features and improved Adaboost[J]. Future Generation Computer Systems, 2020, 106: 347-359.

[19] 吴志军, 刘亮, 岳猛. 基于 ANN 与 KPCA 的 LDoS 攻击检测方法[J]. 通信学报, 2018, 39(5): 11-22.
WU Z J, LIU L, YUE M. Detection method of LDoS attacks based on combination of ANN & KPCA[J]. Journal on Communications, 2018, 39(5): 11-22.

[20] LIU L, WANG H Y, WU Z J, et al. The detection method of low-rate DoS attack based on multi-feature fusion[J]. Digital Communications and Networks, 2020, 6(4): 504-513.

[21] WANG X, QIAN B Y, DAVIDSON I. On constrained spectral clustering and its applications[J]. Data Mining and Knowledge Discovery, 2014, 28(1): 1-30.

[22] CHEN F, YU R, LIU W M. Internet of things attack group identification model combined with spectral clustering[C]//Proceedings of 2021 IEEE 21st International Conference on Communication Technology. Piscataway: IEEE Press, 2021: 778-782.

[23] YE L X, KEOGH E. Time series shapelets: a new primitive for data mining[C]//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2009: 947-956.

[24] ZAKARIA J, MUEEN A, KEOGH E. Clustering time series using unsupervised-shapelets[C]//Proceedings of 2012 IEEE 12th International Conference on Data Mining. Piscataway: IEEE Press, 2012: 785-794.

[25] HILLS J, LINES J, BARANAUSKAS E, et al. Classification of time series by shapelet transformation[J]. Data Mining and Knowledge Discovery, 2014, 28(4): 851-881.

[26] HU W J, YANG Y, CHENG Z Q, et al. Time-series event prediction with evolutionary state graph[C]//Proceedings of the 14th ACM International Conference on Web Search and Data Mining. New York: ACM Press, 2021: 580-588.

[27] MEDICO R, RUYSSINCK J, DESCHRIJVER D, et al. Learning multivariate shapelets with multi-layer neural networks for interpretable time-series classification[J]. Advances in Data Analysis and Classification, 2021, 15(4): 911-936.

[28] SHARAFALDIN I, LASHKARI A H, HAKAK S, et al. Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy[C]//Proceedings of 2019 International Carnahan Conference on Security Technology (ICCST). Piscataway: IEEE Press, 2019: 1-8.

[29] MONTAZERISHATOORI M, DAVIDSON L, KAUR G, et al. Detection of DoH tunnels using time-series classification of encrypted traffic[C]//Proceedings of 2020 IEEE International Conference on Dependable, Autonomic and Secure Computing, International Conference on Pervasive Intelligence and Computing, International Conference on Cloud and Big Data Computing, International Conference on Cyber Science and Technology Congress. Piscataway: IEEE Press, 2020: 63-70.

[30] FENG X W, FU C P, LI Q, et al. Off-path TCP exploits of the mixed IPID assignment[C]//Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2020: 1323-1335.

[31] ZHANG Q, WU J, ZHANG P, et al. Salient subsequence learning for time series clustering[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 41(9): 2193-2207.

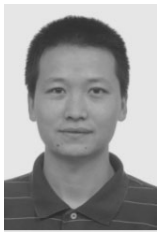
[32] HINDY H, ATKINSON R, TACHTATZIS C, et al. Utilising deep learning techniques for effective zero-day attack detection[J]. Electronics, 2020, 9(10): 1684.

[33] FU C P, LI Q, SHEN M, et al. Realtime robust malicious traffic detection via frequency domain analysis[C]//Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2021: 3431-3446.

[作者简介]

赵静（1987− ），女，甘肃武威人，博士，中国科学院计算机网络信息中心高级工程师，主要研究方向为网络空间安全、信息安全、计算机网络等。

李俊（1968− ），男，安徽桐城人，博士，中国科学院计算机网络信息中心副总工程师，主要研究方向为互联网体系结构、人工智能和大数据应用、互联网安全等。



龙春（1979− ），男，湖北广水人，博士，中国科学院计算机网络信息中心正高级工程师，主要研究方向为智能动态网络安全保障、安全大数据挖掘与分析、云计算与移动互联网安全事件管控等。

万巍（1982− ），男，湖北孝感人，博士，中国科学院计算机网络信息中心高级工程师，主要研究方向为基于人工智能的网络安全异常检测、安全大数据分析等。

魏金侠（1987− ），女，河北秦皇岛人，博士，中国科学院计算机网络信息中心高级工程师，主要研究方向为网络安全大数据分析、网络安全威胁智能检测、基于人工智能的高隐蔽性大规模复杂网络攻击等。

陈凯（1997− ），男，山东淄博人，中国科学院计算机网络信息中心硕士生，主要研究方向为网络空间安全、网络入侵检测。