基于有效载荷的多级实时入侵检测系统框架

刘解放[1] 赵 斌[2] 周 宁[1]

(盐城工学院信息工程学院 盐城 224051)[1] (北京工业大学计算机学院 北京 100022)[2]

摘 要 网络入侵检测系统使用大量特征集来识别入侵,需要处理庞大的网络流量,目前大多数现有的系统缺乏实时异常检测能力。提出了一种基于有效载荷的多级实时入侵检测系统,它首先采用 *n*-gram 分析网络数据包有效载荷来构建特征模型,进行数据准备;其次采用3级迭代特征选择引擎进行特征子集选择,其中主成分分析用于数据的预处理,并结合累积能量、平行分析和碎石检验进行主成分选择;最后采用马氏距离图发现特征间及数据包间隐藏的相关性。马氏距离的差异性准则用来区分正常或攻击数据包。通过 DARPA 99 和 GATECH 数据集验证了该系统的有效性,用 Web 应用程序流量验证了其模型,用 F 值评估了其检测性能。与目前同类主流的两款入侵检测系统进行了对比试验,结果表明:该系统提高了检测精度,降低了误报率和计算复杂度。与中型企业网的真实场景相比,它具有 1.3 倍的高吞吐量。

关键词 入侵检测,数据预处理,N-gram,主成分分析,马氏距离图,迭代特征选择

中图法分类号 TP393 **文献标识码** A

Multilevel Real-time Payload-based Intrusion Detection System Framework

LIU Jie-fang[1] ZHAO Bin[2] ZHOU Ning[1]

(School of Information Engineering, Yancheng Institute of Technology, Yancheng 224051, China)[1]

(School of Computer Science, Beijing University of Technology, Beijing 100022, China)[2]

Abstract Intrusion detection systems use a lot of features sets to identify intrusions, so they need to deal with the huge network traffic. However, most of the existing systems lack real-time anomaly detection capability. This paper presented multilevel real-time payload-based intrusion detection system. It first uses *n*-gram to analyse network packet payload and build feature model for data preparation, and then uses 3-Level Iterative Feature Selection Engine for feature subset selection. Principal component analysis in 3LIFSEng is used for data preprocessing, and combining the cumulative energy, parallel analysis and gravel test, the principal component selection is made. Mahalanobis distance map is used to discover the hidden dependencies between packets and between features. Mahalanobis distance criteria is used to distinguish normal or attack data packets. DARPA 99 and GATECH datasets verify the system's validity. Web application traffic verifies its mode. F-value assesses its detection performance. Experimental results show that compared with the present mainstream two intrusion detection system, the system improves the detection accuracy and reduces the false positive rate and the computational complexity. Additionally, it has 1.3 time higher throughput in comparison with real scenario of medium sized enterprise network.

Keywords Intrusion detection, Data pre-processing, N-gram, Principal component analysis, Mahalanobis distance map, Iterative feature selection

1 引言

随着互联网的飞速发展,计算机安全已经成为一个至关重要的问题。入侵检测系统(IDS)是安全机制的一个重要组成部分。它的目标是通过检测和响应,提供针对恶意使用计算机系统的防御。然而,大多数现有 IDS 缺乏实时异常检测能力。

文献[1-5]给出了入侵检测系统的综述。综述指出,以往大多数针对 IDS 的研究不关注数据预处理技术。入侵检测算法被直接用于粗糙的网络数据中,不涉及流量特性选择。在实际应用中,数据预处理是最重要的阶段之一,它直接影响检测准确性和分类算法的能力。

在理想的情况下,入侵检测系统的目的是在早期阶段检测攻击,换句话说,尽量实时检测攻击,把攻击的影响降低到最小。因此,对于实时入侵检测,系统一旦被部署必须检测攻击。然而,在实际中,很难建立一个具有低误报率和高检测精

到稿日期:2013-06-21 返修日期:2013-10-14 本文受国家自然科学基金(61272500)资助。

刘解放(1982—),男,硕士,讲师,CCF 会员,主要研究领域为网络安全、数据挖掘等,E-mail:l.jiefang@gmail.com;赵 斌(1979—),男,博士生,讲师,CCF 会员,主要研究领域为网络安全与测试、物联网技术、信息取证等;周 宁(1972—),男,硕士,副教授,主要研究领域为网络安全、数据挖掘等。

度的系统。在一般情况下，IDS处理庞大的数据量，其中包含无关的、冗余的特征，造成训练和测试过程的缓慢、较高的资源消耗以及较差的检测率。因此，选择重要且合适的特征来表现网络流量的行为模式并从网络流量数据中明确区分正常或异常活动，是建立一个实时IDS所面临的关键挑战之一。

虽然文献[6-10]中提出从数据包报头中判别特征的方法，但是数据包有效载荷的方法没有明确界定。从基于有效载荷的异常检测研究综述中发现，*n*-gram[11]和libAnomaly[12]是两种常用方法。但是它们的缺点是必须使用大规模的特征集，所以它们无法提供充足且正确的流量判别能力，从而导致高的误报率。此外，由于在网络会话中需要更深度的搜索来寻找巨大有效载荷特征，因此基于有效载荷攻击的检测计算开销较大。这个挑战激发了我们的研究工作，亦即使用合适的特征子集建立一个基于有效载荷的实时入侵检测系统。

本文要解决的相关问题包括：特征集质量和维数灾难。主要工作包括以下5部分。

1)提出了一个3级迭代特征选择引擎进行特征子集选择。第1级利用主成分分析(PCA)技术对原始数据集进行分析，依据每个成分保留的方差，检查和评价多维特征空间中各个成分的重要性。第2级利用数学解决方案即累积能量和平行分析以及非数学的解决方案即碎石检验来分别独立地确定重要的主成分数量。第3级进行特征细化，以及产生和验证正常训练模型。

2)提出使用马氏距离图(MDM)来识别数据包有效载荷的模式。MDM对提取特征间及网络数据包有效载荷间隐藏的相关性有很好的前景。它也部分地捕捉有效载荷的结构信息。这些相关性和结构信息有助于提高检测性能和降低误报率。

3)提出了一个基于有效载荷的多级实时入侵检测系统(MlRePIDS)，它可以实时地检测网络上基于有效载荷的攻击。作为MlRePIDS的主要组成部分，3级迭代特征选择引擎(3LIFSEng)和MDM提高了对网络数据流中攻击包的高效检测，降低了误报率。

4)采用F值作为度量标准，以评估MlRePIDS的性能。使用F值的原因是正常和异常实例的数量分布并不均匀，因此，如果使用误报率(FPR)、漏报率(FNR)、入侵检测率(TPR)、正常检测率(TNR)指标评估系统的性能，系统偏向于达到的精度超过99%。然而，F值是基于精度和召回率(也称查全率)的，与训练和测试样本大小无关。

5)通过DARPA 99[13]和GATECH数据集[14]测试了MlRePIDS的有效性，并将之与目前同类主流的IDS-PAYL[15]和McPAD[16]进行了检测性能(F值)和计算复杂度的对比。实验结果表明，MlRePIDS提高了检测精度，降低了误报率和计算复杂度，可以适合中型企业网的真实场景。本文第2节分析最新的相关研究工作；第3节讨论MlRePIDS的框架；第4节给出实验结果和分析；第5节是性能对比分析；最后给出结论。

2 相关研究

本节详细叙述了最近基于有效载荷的IDS的研究。由于越来越多的攻击采用纯粹的数据包有效载荷，因此研究人员最近更热衷于用基于有效载荷的方法来建立IDS检测模型。文献[15-18]中提出了几种典型高效的基于有效载荷的IDS。

以前异常检测研究的工作是基于对应用层有效载荷的简单统计来构建Web应用程序的正常轮廓。最新的研究采用了*n*-gram分析网络数据包有效载荷来构建特征模型。

Wang和Stolfo提出了PAYL[15]，它使用1-gram构建了基于网络数据包有效载荷的一个字节频率分布模型。数据包有效载荷的预处理使用1个字节的滑动窗口创建特征向量，它包含有效载荷中256个可能的1-gram的相对频数。采用简化的马氏距离来比较违反模型的新进入的数据流，整体检测率接近60%，误报率小于1%。虽然PAYL的方法在显示异常字节分布时是有效的，但是它也有一些缺点，例如它不能抵挡模拟攻击[16]和对于异常检测考虑完整载荷，这在高速和高带宽网络中就是一个重大问题。

Bolzoni等人提出了POSEIDON[19]，即一种双层入侵检测模型。POSEIDON结合了SOM(Self Organizing Map)和PAYL。SOM用于处理数据包的有效载荷，PAYL用作检测的基础。SOM的目的是对于一个给定的目标地址和端口识别类似的有效载荷。SOM可以提高检测精度。然而SOM和PAYL都需要单独进行训练，因此很难达到较高的精度。对于SOM，神经元的个数依赖于网络的大小，因此计算负荷会随着神经元个数增加而成倍增加。

为了模拟有效载荷的结构，Wang等人提出了ANAGRAM[20]。$n(n\geqslant 1)$被用于在256^n维特征空间中提取字节序列信息。监督学习通过将正常和攻击数据包的*n*-gram存储到两个独立的布隆过滤器中来构建正常和攻击的流量模型。然而，根据Perdisci等人的研究[16]，布隆过滤器在高带宽或高速率网络中不能工作，因为ANAGRAM把*n*-gram存储在布隆过滤器中，并基于在检测阶段未察觉的和恶意的*n*-gram的数量生成一个评分。然而，由于维数灾难和计算的难题，它很难建立精准的模型。Perdisci等人后来提出了一个名为McPAD的IDS[16]，它通过使用一个2字节的滑动窗口涵盖所有集合创建了2-gram。这就产生了高维特征空间($256^2=65536$)。因为每个字节在0～255范围内有个值，$n=2$。然后通过聚类算法对该特征空间的维数进行降维。然而，他们并没有给出聚类个数和分类器选择的准则。Rieck和Laskov也提出了一个模型[21]，它是从连接负载中提取高阶*n*-gram语言特征，通过使用向量相似度技术(如核函数、距离函数)来比较连接负载中的高阶*n*-gram和词。

然而，所有上述讨论的方法只考虑有效载荷的特征，而没有考虑特征间及数据包有效载荷间的相关性。这将导致高误报率。

最近，很多学者专注于高速网络检测技术的研究。网络组件使用DPI(Deep Packet Inspection)[22,23]技术作为分析器来检测所有层上的攻击。为了识别和阻止恶意攻击，研究人员正在探索每个进入的数据包的报头和有效载荷。文献[24]中，研究人员使用DPI技术开发了网络入侵检测和预防系统。DPI搜索整个报头和有效载荷用于模式匹配，使用大量的规则数据库比较进入的数据包。不幸的是当传统的方法(如确定性有限自动机)用于快速的正则表达式扫描时，内存消耗过高。由于规则的复杂性和DPI基于软件的实施，基于DPI数据包处理速度的解决方案是非常有限的。比如，SNORT有4000多个规则，在正常的环境下处理链路的速率

仅达 250Mbps[24]。这种速率不能满足中等速度的访问或边界网络需求。

目前，基于有效载荷的 IDS 通常存在 3 个主要问题：高误报率、高维数据的复杂性和协议的动态构造。这是因为基于有效载荷的 IDS 使用了大量特征来区分网络中的正常和异常数据包。特征集中冗余的、不相关的特征的出现导致了高误报率，这限制了基于有效载荷的 IDS 的在线网络流的实时处理。

本文中 MlRePIDS 使用了 PCA[25]，它是一个有效的方法，通过提供一个线性映射，把 n 维特征空间降至 m 维特征空间，提高了检测性能，更适合于实时应用。在实践中，减少特征间的复杂关系。从原始特征空间中丢弃无关的、冗余的、不重要的特征对于 IDS 的实时应用是重要的。一方面，这不仅减少流量，而且减少处理时间；另一方面还将提高检测率。

PCA 尽管已被应用到基于报头的入侵检测[26-28]领域，但实现的合理特征减少了。然而，为有效载荷特征选择使用 PCA 来对数据进行预处理的工作还没有发现。Nwanze 等人[29]使用基于 PCA 的数据挖掘技术讨论数据包有效载荷的建模。然而，他们忽略了 PCA 的主要思想，没有把原始数据投影到一个较低维的特征空间。此外，他们没有考虑特征之间的相关性。

本文研究的目的是降低特征空间的维数和误报率。实现途经是将高效的数据包有效载荷进行预处理并将数据表示为合适的格式，以便用于实时入侵检测。

3 基于有效载荷的多级实时 IDS

本节详细阐述我们研究的具体内容。首先提出 MlRePIDS 的框架，然后讨论框架中的模块：数据准备模块、n-gram 文本分类模块、3 级迭代特征选择引擎、轮廓生成器和数据流分类器。

3.1 MlRePIDS 的框架

完整的框架有 4 个阶段，包括数据准备、数据预处理、模型生成和异常检测，如图 1 所示。

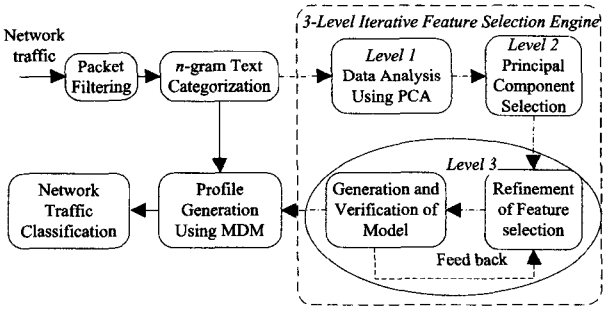


图 1 基于有效载荷的实时入侵检测系统的框架

第一阶段，数据准备和 n-gram 文本分类[30]。对于数据准备，根据进入网络数据的应用和负载长度的类型进行过滤。n-gram 文本分类把网络数据包有效载荷转换成一系列的特征向量。这些特征向量描述了最初高维特征空间的进入流量模式。

第二阶段，3LIFSEng 用于主成分选择。每一级执行一个特定的任务。第 1 级，PCA 技术[25]分析网络流量。第 2 级，使用 3 种方法进行特征子集选择。第 3 级，提炼最佳特征子集和评估特征子集的识别能力。MDM[18,33,34]用来在选定的特征间捕捉更复杂的非线性相关性，并表示网络流量轮廓。

第三阶段，利用 3LIFSEng 的输出建立一个正常的流量轮廓。创建一个 MDM 作为一个正常网络流量的正常轮廓，它是用于最后阶段的新进入网络数据流的分类。

第四阶段，用马氏距离测量新进入网络数据包的轮廓和预先形成的正常轮廓间的差异性。根据新进入网络数据包的轮廓与正常轮廓的偏移量，数据包被分类为正常或攻击。下面的小节中给出了每个模块的详细描述。

3.2 框架模块

本节对 MlRePIDS 的所有模块的技术细节进行详细的描述。

3.2.1 数据准备模块

数据准备是该框架的第一阶段，在此要准备不同的数据集。我们使用 Wireshark[35] 把网络数据分成各种类型。Wireshark 是一个流量分析器，能够基于服务类型、目的地址、负载长度、网络流量方向来分离网络数据。网络数据可以来自真实的网络或收集的 tcpdump 文件。准备好的数据集可在 MlRePIDS 的下一阶段使用。

3.2.2 n-gram 文本分类模块

n-gram 文本分类负责有效载荷的特征分析和构建。使用 n-gram 文本分类技术($n=1$)从数据包有效载荷中提取原始特征，并转换成一系列的特征向量。每个有效载荷用 256 维特征空间的特征向量表示。相对频率用式(1)表示。

$$f_i=\frac{O_i}{\sum_{j=1}^{256}O_j} \tag{1}$$

O_i 是第 i 个 n-gram 的发生率，f_i 是第 i 个 n-gram 的相对频率。相对频率的总值如式(2)所示。

$$\sum_{i=1}^{256}f_i=1 \tag{2}$$

因此，一个数据包有效载荷由一个相对频率向量 $q=[f_1 f_2\cdots f_{256}]^{\mathrm{T}}$ 表示；在一个 256 维的特征空间，它表示网络有效载荷中的一个模式。T 代表矩阵的转置。

3.2.3 3 级迭代特征选择引擎

3LIFSEng 包括第 1 级 PCA 分析数据、第 2 级主成分选择、第 3 级特征选择细化及模型的生成和验证。

第 1 级，使用 PCA 分析原始数据集；作为一个线性的数学系统，PCA 是在基于特征向量的多元分析的基础上开发的。它试图通过转换一组观测数据集到一个新的正交坐标系中，从而高效地表示数据，这时数据最大程度地降低了相关性。包含较重要变元的坐标轴较多地贡献了数据表示。最初的几个最有贡献的坐标轴通常用于构建新的低维特征空间，这样可以更高效地表示数据。

PCA 应用于网络数据集 $Q=[q_1 q_2\cdots q_m]$，其中 m 是观测数据集个数，每个观察数据集 $q_i(1\leqslant i\leqslant m)$由一个 256 维特征向量 $q_i=[f_1^i f_2^i\cdots f_{256}^i]^{\mathrm{T}}$ 表示。首先，为使 PCA 正确地工作，数据集中所有观察值都要进行零均值的标准化。零均值数据集如式(3)所示。

$$Q_{zh}=\begin{bmatrix}q_1-\bar{q}\\ q_2-\bar{q}\\ \vdots\\ q_m-\bar{q}\end{bmatrix}^{\mathrm{T}} \tag{3}$$

$\bar{q}=\frac{1}{m}\sum_{i=1}^{m}q_i$，通过分析样本协方差矩阵 C_Q 得到主成分，

数据集中的 C_Q 如式(4)所示。

$$C_Q=\frac{1}{m-1}Q_{sh}Q_{sh}^{T} \tag{4}$$

使用特征分解，协方差矩阵 C_Q 被分解为矩阵 W 和对角矩阵 λ。它们满足条件 $\lambda W=C_Q W$。矩阵 W 的列代表协方差矩阵 C_Q 的特征向量(称为主成分)。矩阵 λ 的沿对角线排列的元素是分类的特征值，它与矩阵 W 的相应特征向量有关。

根据数据的表示，PCA 只说明一个特征空间中不同成分的贡献，并不能确定应保留的主成分个数。因此，第 2 级需要另外的技术和 PCA 分析一起来决定应保留的主成分的最佳个数。

第 2 级，使用累积能量[31]、碎石检验[32]和并行分析方法实现良好的数据预处理，尽可能多地保持相关信息。累积能量、碎石检验和并行分析方法各自用来选择对应的 k_1，k_2 和 k_3 主成分。k_1，k_2 和 k_3 小于或等于 k，其中 k 等于 256。这些数学和非数学方法用来验证相互的结果。主成分子集对应于选定的 k_1，k_2 和 k_3，代表已降维的特征空间，它提供了最好的数据包有效载荷展示。通过把特征向量 $q_i=[f_1 f_2 \cdots f_{256}]^{T}$ 投影到这些所选定的降维特征空间上，特征向量的维数可以降低到较小值，分别为 k_1，k_2 和 k_3。同时，这些方法能够保证降维的特征向量可以正确表示数据包有效载荷。下面给出每个方法的简短解释。

累积能量。与成分相关联的能量由相应的特征值表示。特征值越大，相应成分的能量越大。假设 (λ_1,u_1)，(λ_2,u_2)，$\cdots$，(λ_k,u_k) 是 k 个特征值，是从协方差矩阵 C_Q 分离的特征向量对。k_1 成分的累积能量由能量的总和定义，它根据式(5)计算得到。

$$CE=\sum_{j=1}^{k_1}\lambda_j \tag{5}$$

$k_1\in\{1,\cdots,k\}$，由目标函数决定，如式(6)所示。

$$\frac{CE}{\sum_{j=1}^{k}\lambda_j}\geqslant\alpha \tag{6}$$

α 是对应于原始空间中总变化的子空间变化率。该目标函数拟取得尽可能小的 k_1 值，而得到一个相当高的 CE 值。

碎石检验。它是一个图形化的方法，在 1966 年由 Cattell[32]首次提出。在碎石图中，所有的特征值依据主成分以降序描绘。在碎石图中，我们寻找第 k_2 个点，这里特征值急剧下降并趋于水平。此点被确定为“肘”。第 k_2 个点之后，其余的 $(k-k_2)$ 个主要成分被忽略，并不在模型中使用。这是基于最重要的成分从协方差矩阵中抽取大比例的方差，而其余的不重要的 $(k-k_2)$ 个成分与低值方差有关。碎石检验方法中，碎石开始时没有急剧的转变，不具鲁棒性和可重复性。

并行分析。它是 Cattell 的碎石检验的改进。它减缓了成分的不确定性问题，并确定哪些可变载荷对于每个成分是重要的。此操作重复两次，在两次迭代中，每个成分所得到的特征值被用于计算均值和标准差(SD)。从均值和标准差中，获得第 95 个百分位数值(第 95 个百分位数值 = 均值 + 1.65SD)。如果一个成分的特征值超过第 95 个百分位的模拟值，该成分将被保留。

第 3 级，特征细化和评估模块。在细化阶段，我们扩展了来自第 2 级选定的主成分的范围。然后，我们观察到主成分子集的不同能量表示数据包的有效载荷。最后通过正常训练模型的迭代评估，选择了最终主成分 $k_{final}\in\{k_1,k_2,k_3\}$。模型训练使用了 F 值，其定义如式(7)所示。

$$F=(1+\beta^2)*Recall*\frac{Precision}{\beta^2(Recall+Precision)} \tag{7}$$

式(8)中定义了精度，它表示多少事件是实际入侵。低精度值意味着更高程度的误报，反之亦然。式(9)定义了召回率，它表示分类器所涵盖的真实入侵的百分比。低召回率值代表较高程度的漏报，反之亦然。

$$Precison=\frac{TP}{TP+FP} \tag{8}$$

$$Recall=\frac{TP}{TP+FN} \tag{9}$$

式(8)和式(9)中，入侵检测(TP)表示入侵检测系统正确检测到的攻击数量；正常检测(TN)表示正常数据包被入侵检测系统正确分类为正常数据包的数量，没有犯任何错误。误报(FP)表示正常的数据包被入侵检测系统错误分类为攻击数据包的数量。漏报(FN)表示攻击数据包被入侵检测系统错误分类为正常包的数量。

式(7)中的 β 对应精确与召回率的相对重要性，通常设置为 1。一方面，当精度和召回率有同等的权重，并接近为 1 时，该系统可以实现 F 值接近 1，这表明良好的性能，意味着分类有 0%误报和 100%入侵检测率。在另一方面，F 值接近 0 表示差的性能。因此，期望一个分类器的 F 值尽可能的高。

所选择的主成分 k_{final} 有利于分类器实现最大的 F 值。然后，选择的主成分 k_{final} 用于轮廓生成。

3.2.4 使用 MDM 生成轮廓

网络流量轮廓使用 MDM 产生。MDM 能够捕捉复杂的非线性的数据相关性。通过使用 MDM，可以获得从原始特征向量 $q=[f_1 f_2 \cdots f_{256}]^{T}$ 到 k_{final} 维特征子空间 $[u_1 u_2 \cdots u_{k_{final}}]$ 的投影所产生的特征向量 $[x_1 x_2 \cdots x_{k_{final}}]$ 的特征间隐藏的相关性和数据包间的相关性，其表示如下。

$$\sum_a=(x_a-\mu)(x_a-\mu)^{T}(1\leqslant a\leqslant k_{final}) \tag{10}$$

$$d_{(a,b)}=\frac{(x_a-x_b)(x_a-x_b)^{T}}{\sum_a+\sum_b}(1\leqslant a,b\leqslant k_{final}) \tag{11}$$

$$D=\begin{bmatrix} d_{(1,1)} & d_{(1,2)} & \cdots & d_{(1,k_{final})} \\ d_{(2,1)} & d_{(2,2)} & \cdots & d_{(2,k_{final})} \\ \vdots & \vdots & \ddots & \vdots \\ d_{(k_{final},1)} & d_{(k_{final},2)} & \cdots & d_{(k_{final},k_{final})} \end{bmatrix} \tag{12}$$

其中，x_a 代表投影特征向量中第 a 个投影特征；μ 代表每个投影特征的平均值；$d_{(a,b)}$ 定义了第 a 个投影特征到第 b 个投影特征的马氏距离；$\sum_a$ 是每个投影特征的协方差值；D 是 MDM，用于生成训练和测试数据的网络数据流轮廓，这些轮廓用于进入网络数据流的分类。

3.2.5 流量分类

马氏距离负责测量已形成的网络数据流轮廓和新进入的网络数据流量轮廓间的差异性。加权分 W 使用式(13)计算，用来检测入侵活动。

$$W=\sum_{a,b=1}^{k_{final}}\frac{(d_{obj(a,b)}-\overline{d}_{nor(a,b)})^2}{\sigma^2_{nor(a,b)}} \tag{13}$$

$\overline{d}_{nor(a,b)}$ 和 $\sigma^2_{nor(a,b)}$ 是正常轮廓的 MDM 中第 (a,b) 个元素的平均值和方差，正常轮廓的 MDM 如式(14)所示。$d_{obj(a,b)}$ 是新进入数据包的 MDM 中第 (a,b) 个元素，如式(15)所示。

$$D_{nor}=[d_{nor(a,b)}]_{k_{final}\times k_{final}} \tag{14}$$

$$D_{obj}=[d_{obj(a,b)}]_{k_{final}\times k_{final}} \tag{15}$$

如果加权分 W 超过阈值，进入数据包被认为是一种入侵。

4 实验结果及分析

本节首先简要介绍了数据集和攻击类型，然后讨论了模型的训练和测试，最后介绍了实验结果和分析。

通过在 DARPA 99 和 GATECH 数据集上的一系列的实验评估了模型的性能。这两个数据集也应用到同类主流的 IDS 上，第 5 节对结果进行了比较。

4.1 数据集

4.1.1 训练数据集

为了训练模型，我们从 DARPA 99 数据集中提取第 1 周和第 3 周入站的“HTTP 请求”流量。提取的正常流量对应两个不同的 HTTP 服务器。主机 A 和主机 B 的训练使用的总的数据包个数分别是 13933 和 10464。

4.1.2 测试数据集

为了测试模型的检测性能，我们使用两种数据集的攻击包。标记测试数据被进一步预处理形成两个测试集，它们的实例不在我们的训练集中。实验中，我们仅专注于来自 HTTP 服务的攻击。

基于 HTTP 的攻击主要来自于对 Web 服务器的 HTTP GET/POST 请求。DARPA 99 数据集中有些基于 HTTP 的攻击，如 Apache2 攻击、CrashIIS 攻击和 Phf 攻击。GATECH 攻击数据集中有些非多态的 HTTP 攻击（由 Ingham 和 Inoue 提供[15]）和一些多态 HTTP 攻击（由 Perdisci 等人使用 CLET 引擎生成[16]）。Generic、Shell-code 和 CLET 攻击被放在不同的组中，每个组都有同一类别的攻击。所有的 HTTP 请求攻击数据包用在我们的实验中。

4.2 模型训练和测试过程

实验包括模型的训练和测试：

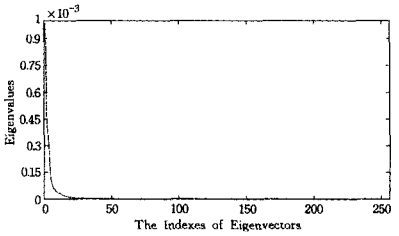
1. 正如 3.2.2 节中讨论，我们使用长度为 1-byte 的滑动窗口，分析了 HTTP GET 请求数据包有效负载的 185 个字节，然后使用 256 维特征空间的一个特征向量 q 对它进行表示。

2. 正如 3.2.3 节中讨论，第 1 级利用 PCA 技术分析原始数据，即 ASCII 字符发生频率；在训练数据集中，投影原始数据到一个降维的特征空间上。第 2 级，重要的主成分选择在 PCA 的基础之上进一步由累积能量、碎石检验和并行分析方法完成。

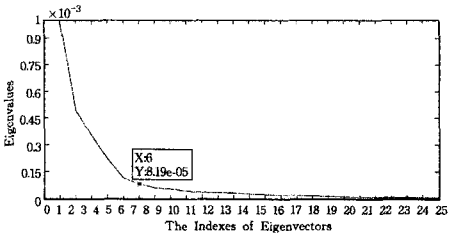
首先，CE 用于主成分选择。在式(6)中，设 CE 级别为 93%。$k_1=7$，表示最先的 7 个主成分被选择为最佳的子空间来表示数据。

然后，我们使用碎石检验方法绘制碎石图和选择另一组主成分。碎石图可以看作是由给定的主成分捕获的方差。图 2(a)显示了完整碎石图，它是使用 $k(k=256)$ 个主成分(X 轴)和相应的方差即特征值(Y 轴)绘制而成的。主成分依据相应的方差按降序进行排序。在图 2(a)中我们发现了一个“肘”。为了提供更好的视觉，我们放大碎石图，在图 2(b)中显示了前 25 个主成分。从图 2(b)中可以发现在碎石图的前部分方差急剧递减，然后在第 6 个主成分后开始变平。在图 2(b)中我们可以观察到“肘”在第 6 至第 9 个主成分范围处。$k_2=6$ 个主成分能够捕捉到约 92%的方差。在第 k_2 个点之后，剩下的$(k-k_2)$个主成分捕捉到只有约 8%的总方差，而被忽略。

在实验室中，我们使用 $k_2=6$ 作为重要的主成分。然而，从图 2(b)中已经观察到主成分的范围在 6 至 9 之间，k_2 为何值才是最合适的还不是很清楚。为了克服这个歧义性，我们使用并行分析方法来验证 k_2 的选择。



(a) Full scree plot



(b) Enlarged scree plot with 25 eigenvetors

图 2 碎石检验图

我们通过使用并行分析方法验证了碎石图的结果，第 3.2.3节中对同一数据集进行了讨论。并行分析的结果建议采用前 7 个主成分的选择，它们与通过使用累积能量方法获得的相同。表 1 列出了 3 种方法的结果。

表 1 主成分选择

主成分选择方法	累积能量(0.93)	碎石检验	并行分析
主成分个数	7	6	7

3. 为了测试，我们把进入数据包有效载荷的特征向量投影到低维的特征向量空间(最终选择的主成分)上，并使用马氏距离差异方法检测侵入行为。在检测攻击上，使用 F 值评估 MlRePIDS 的性能。

在实验中，使用了来自于 DARPA 99 数据集中 10 天正常的“HTTP GET request”流量。正常流量随机分为 3 个子集。随机选择一个子集来训练模型。剩下的两个子集用来测试该模型。

为了检测 DARPA 99 数据集中各种各样的攻击，要把原始特征向量投影到最优主成分上得到实验需要的特征，最优主成分是由 3LIFSEng 确定的。在 GATECH 攻击数据集上我们进一步评估模型，这些攻击由 generic，polymorphic (CLET)和 shell-code 攻击组成。实验在一台计算机上进行，它的配置为 2 个 3.33 GHz/8M cache 4 核 Xeon CPU、24GB DDR3-1333EC 内存。这是一个共享的计算环境，主要用于重量级的数学计算和模拟实验。性能很大程度上会受到同时运行进程个数的影响。MATLAB 用于模拟。

4.3 结果与分析

实验结果通过两步进行分析。第一步的实验中，我们得到了主成分的最优子集。然后，我们设计了一些基于图 1 的实验。我们使用各种各样的主成分子集，即从 5 个到 9 个主成分。实验也使用了不同的阈值，从 2σ 到 3.5σ。结果如表 2 所列。

表 2　不同主成分个数(PCs)的相应性能情况

性能指标	5PCs(%)	6PCs(%)	7PCs(%)	8PCs(%)	9PCs(%)
TNR	98.63	99.33	99.15	98.69	98.01
TPR	98.70	99.50	100	100	99.97
FNR	1.30	0.50	0	0	0.03
FPR	1.37	0.67	0.85	1.31	1.99

表 2 表明了在主成分个数改变时，TNR，TPR，FNR 和 FPR 的变化。为了获得最优的主成分个数，每个特征子空间的 F 值使用式(7)计算。图 3 表明了随着主成分个数的变化，F 值的变化。结果显示 7 个主成分可以达到最佳的 F 值，也即 7 个主成分的特征子空间有最好的表现和识别能力及高精度。特征向量的增加和减少都会降低 MlRePIDS 的性能。

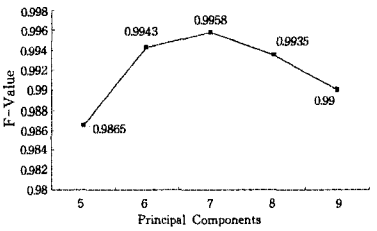


图 3　F 值的趋势

由此得出结论，即 PCA 和其它 3 个选择方法有助于数据集降维，从 256 降到 7 维度。在选定的 7 维特征空间中，使用 3LIFSEng 提取信息量，这有助于创建更准确的正常流量轮廓，其中 MDM 负责对流量进行分类。

0	0.001406625	0.001449804	0.001332988	0.001463112	0.001270879	0.001241186
0.001406625	0	0.000289528	0.000305565	0.000268982	0.000231624	0.000208517
0.001449804	0.000289528	0	0.000239652	0.000214287	0.000194018	0.000163789
0.001332988	0.000305565	0.000239652	0	0.000287999	0.000198282	0.000159613
0.001463112	0.000268982	0.000214287	0.000287999	0	0.00016282	0.000170964
0.001270879	0.000231624	0.000194018	0.000198282	0.00016282	0	9.17989exp−05
0.001241186	0.000208517	0.000163789	0.000159613	0.000170964	9.17989exp−05	0

图 4　正常 HTTP 有效载荷的 MDM

0	7.211042exp−05	3.686978exp−05	0.00237153	0.00010235	0.00078711	0.00072289
7.211042exp−05	0	5.214637exp−05	0.00163562	0.00033709	0.00132127	0.00034557
3.686978exp−05	5.214637exp−05	0	0.00225582	0.00012659	0.00085651	0.00065864
0.00237153	0.00163562	0.00225582	0	0.00344129	0.00586325	0.00048362
0.00010235	0.00033709	0.00012659	0.00344129	0	0.00032706	0.00135888
0.00078711	0.00132127	0.00085651	0.00586325	0.00032706	0	0.00300482
0.00072289	0.00034557	0.00065864	0.00048362	0.00135888	0.00300482	0

(a) Apache2 攻击有效载荷

0	0.000677245	0.00081015	0.00032632	0.00019996	0.00095956	0.000148432
0.000677245	0	0.00022798	0.00036073	0.00038006	0.000451205	0.00033855
0.00081015	0.00022798	0	0.00029764	0.0003492	0.00030296	0.00032801
0.00032632	0.00036073	0.00029764	0	8.31436139	0.00032254	0.00011205
0.00019996	0.00038006	0.0003492	8.31436139	0	0.00033113	8.634902exp−05
0.00095956	0.000451205	0.00030296	0.00032254	0.00033113	0	0.00055453
0.000148432	0.00033855	0.00032801	0.00011205	8.634902exp−05	0.00055453	0

(b) CrashIIS 攻击有效载荷

0	0.051788147	0.047358766	0.045255171	0.037653843	0.039655825	0.051041546
0.051788147	0	0.03508168	0.05975747	0.05529712	0.05478485	0.03144298
0.047358766	0.03508168	0	0.03686035	0.0250256	0.0571498	0.0332321
0.045255171	0.05975747	0.03686035	0	0.05269052	0.05324839	0.05400761
0.037653843	0.05529712	0.0250256	0.05269052	0	0.03450803	0.04522816
0.039655825	0.05478485	0.0571498	0.05324839	0.03450803	0	0.04336399
0.051041546	0.03144298	0.0332321	0.05400761	0.04522816	0.04336399	0

(c) Phf 攻击有效载荷

图 5　攻击 HTTP 有效载荷的 MDM

为了表明 MDM 是如何呈现特征间的相关性，图 4 和图 5 中分别给出了使用投影特征和最优 7 维空间生成的正常和攻击 HTTP 有效载荷的 MDM。从图 4 和图 5 中可以看出，每个 MDM 都是一个对称矩阵，其对角线的元素值等于零。这是因为对于本身，特征的相关性值始终为零。MDM 也证明了正常投影特征间的相关性不同于攻击投影特征的相关性。此外，7 维空间也有助于高效和准确地区分正常和各种攻击有效载荷。图 4 表明了正常 HTTP 有效载荷的 MDM，图 5(a)－(c)表明了 Apache2，Phf，CrashIIS 攻击轮廓的 MDM。

虽然我们可以直接比较正常和攻击轮廓，以确认正常和各种攻击有效载荷间的差异，但是它比较耗时。有了训练数据集的 MDM 轮廓和新进入数据，可以计算出加权分 W。如果加权分 W 的偏移量大于预先选定的阈值，则新进入的数据包被认为是攻击包。

此外，为了评估 MlRePIDS 的鲁棒性，我们在 GATECH 攻击数据集上选定了一些未知攻击来进行同样的实验。表 3 报告了在最优 7 维空间上的 TNR，TPR，FNR，FPR 和 F 值。从表 3 可以得出，MlRePIDS 具有较高的检测率、低的误报率和低的漏报率。F 值为 0.976，这证实了该模型检测精度高，并证明了其良好的性能。

表 3　性能情况

性能指标	7 个特征向量(%)
正常检测率	99.15
攻击检测率	96.29
漏报率	3.71
误报率	0.85
F 值	0.976

总之，MlRePIDS 能够很好地检测到新型的攻击，具有非常高的 F 值(0.976)和低误报率。

5　性能比较

本节把 MlRePIDS 和目前同类主流的两款入侵检测系统——PAYL 和 McPAD 进行比较；然后，进一步将 MlRePIDS 与中型企业网真实的吞吐量进行比较。

5.1　检测性能

我们首先比较了 3 款 IDS 的检测性能。为了展示一个合理的比较，使用了文献[16]的误报率和检测率。在文献[16]的图 6 和图 7 中，我们估算了 generic，shell-code 和 polymorphic 的平均检测率。在 GATECH 数据集上，使用 1%的误报率分别计算了 PAYL 和 McPAD 的 F 值。正如文献[16]提到的，它们与 DARPA 99 和 GATECH 数据集上的结果相似。表 4 表明了 DARPA 99 和 GATECH 数据集上 3 款 IDS 的 F 值比较。从表 4 中得出结论：MlRePIDS 具有更好的 F 值。

表 4　性能对比

	MlRePIDS	PAYL	MCPAD
DARPA 99	0.9958	0.969	0.953
GATECH	0.976	0.969	0.953

注：后两列 F 值来自于文献[16]。

5.2　复杂度分析

本小节分析了 3 款 IDS 所使用的算法的计算复杂度。因为算法的训练可以进行脱机执行，它并不影响算法在检测时的效率，所以分析中我们只考虑计算复杂度在测试中所涉及的计算。

给定一个长度为 n 的有效载荷 P 和一个固定值 v 时，v-gram 的发生频率的计算复杂度为 $O(n)$。这些算法提取特征的数量是恒定的，而不管实际的 n 和 v 值(MlRePIDS 和 PAYL 提取 2^8 特征和 McPAD 提取 2^{16}特征)。

MlRePIDS 的特征降维过程可以通过简单的运算即 $2^8 * 2 * 7 = 3584$ 来完成。相比之下，McPAD 算法通过使用简单的查找表和若干和运算将 v-gram 发生频率分布映射到 k 个特征集，从而达到特征减少。和运算总是小于 2^{16}(与 k 值无关)。因此，MlRePIDS 和 McPAD 的特征降维过程的计算复杂度为 O(1)。然而，在 PAYL 中没有进行特征减少。

因此，MlRePIDS、PAYL 和 McPAD 数据预处理算法的计算复杂度等于特征提取和降维过程的计算复杂度。由于 MlRePIDS 使用一个固定的有效载荷长度(185 字节)提取发生频率，因此数据预处理的计算复杂度为 O(1)。因为没有执行特征降维，PAYL 数据预处理的计算复杂度为 $O(n)$。对于 McPAD，它必须被重复 m 次(m 代表不同分类器的个数)，且每次选择不同的 v 值。因此，McPAD 的数据预处理的计算复杂度为 $O(nm)$。

一旦特征被提取并维数降低到 k，每个有效载荷根据 m 个分类器进行分类。为了分类一个有效载荷 P，MlRePIDS 在有效负载 P 和预先确定的正常轮廓之间计算马氏距离。鉴于预先所确定的特征个数为 7 和一个单一的分类器用于分类，MlRePIDS 分类过程的计算复杂度为 O(1)。同样，PAYL 使用一个单一的分类器来分类有效载荷 P。因此，PAYL 分类过程也可以在 O(1)内完成。相比 MlRePIDS 和 PAYL，McPAD 有 m 个分类器。每个分类器计算有效载荷 P 与每一个支持向量 s 之间的距离，此处 P 有 k 个特征集表示，s 在训练中获得。因此，使用 McPAD 完成有效载荷分类的计算复杂度为$O(ks)$。McPAD 必须重复分类处理 m 次，然后将它们的结果合并。因此，McPAD 整个分类过程的计算复杂度为 $O(mks)$。表 5 给出了 3 款 IDS 算法复杂度的详细分解。

表 5　MlRePIDS、PAYL 和 McPAD 的计算复杂度

	MlRePIDS	PAYL	McPAD
数据预处理	O(1)	O(n)	O(nm)
分类	O(1)	O(1)	O(mks)
整体	O(1)	O(n)	O(nm+mks)

如表 5 所列，MlRePIDS、PAYL 和 McPAD 的整体计算复杂度分别为 $O(1)$，$O(n)$ 和 $O(nm+mks)$。这证明：与 PAYL 和 McPAD 相比，MlRePIDS 具有最低的计算复杂度。

通过比较 MlRePIDS 与拥有 1GB 速率网关的中型企业网的吞吐量，我们评估了 MlRePIDS 的效率。吞吐量的比较是基于处理数据包的数量，而且考虑了网络对我们数据包处理速度最理想的参数。一方面，对于一个中型企业网络吞吐量，所考虑的理想参数为每秒内 25600 报文。然而，我们期望是实时处理，所以吞吐量会比用于比较的少得多。另一方面，本方案可以处理每秒 33146 个数据包，这是企业网络上的数据包处理速度的 1.3 倍以上，这表明本方案有能力实现实时。

依据检测精度和算法计算复杂度总结了 MlRePIDS 的整体性能，结果表明它的性能优于目前同类主流的 IDS——PAYL 和 McPAD。此外，在吞吐量方面，与一个拥有 1 GB 速率网关的中型企业网的吞吐量相比，MlRePIDS 每秒处理更多的数据包。因此，本文所提出的模型 MlRePIDS 有能力实时处理数据包。

结束语　本文提出一个高效的基于有效载荷的多级实时入侵检测系统，它通过使用 3 级迭代特征选择引擎和马氏距离图分析 HTTP 有效载荷来检测 Web 应用程序的攻击。马氏距离用于分类网络数据。该模型选择低维特征空间来检测攻击。此外，MlRePIDS 能够实时识别正常和攻击模式。

3LIFSEng 适用于选择最优特征子空间和降维。这是因为在 MlRePIDS 中所使用的数据存在高维问题而影响检测效率。

此外，MDM 有利于探寻特征间及数据包有效载荷间隐藏的相关性。而且，MDM 能够捕捉到有效载荷的部分结构信息，从而提高该模型的性能。

MlRePIDS 通过了 DARPA 99 和 GATECH 数据集的测试。实验结果表明，该方法能够有效地检测攻击，具有高检测

率和低误报率。利用该方法，生成网络轮廓所需的特征个数非常少。这表明该模型具有较低的计算复杂度，所需的训练和测试时间较短。该模型对于实时检测具有较好的潜能。

参 考 文 献

[1] 黄金钟，朱淼良. 基于程序的异常检测研究综述[J]. 计算机科学，2011，38(6)：6-13

[2] Patcha A, Park J M. An overview of anomaly detection techniques: existing solutions and latest technological trends[J]. Computer Networks, 2007, 51(12): 3448-3470

[3] 边婧，彭新光，闫建红. 入侵检测大数据集代价敏感重平衡分类策略[J]. 小型微型计算机系统，2012，33(11)：2526-2530

[4] Lazarevic A, Kumar V, Srivastava J. Intrusion detection: a survey[M]// Managing Cyber Threats. Springer, 2005: 19-78

[5] Early J, Brodley C. Behavioral features for network anomaly detection[M]// Machine Learning and Data Mining for Computer Security. Springer, 2006: 107-124

[6] Mahoney M, Chan P K. PHAD: packet header anomaly detection for identifying hostile network traffic[DB/OL]. http://cs.fit.edu/~ mmahoney/paper3.pdf, 2013-06-17

[7] 魏小涛，黄厚宽，田盛丰. 在线自适应网络异常检测系统模型与算法[J]. 计算机研究与发展，2010，47(3)：485-492

[8] Kotsiantis S, Kanellopoulos D, Pintelas P. Data preprocessing for supervised leaning[J]. International Journal of Computer Science, 2006(1): 111-117

[9] Garca-Teodoro P, Daz-Verdejo J, Macia-Fernandez G, et al. Anomaly-based network intrusion detection: techniques, systems and challenges[J]. Computers&Security, 2009, 28(1/2): 18-28

[10] 宁卓，龚俭，顾文杰. 高速网络中入侵检测的抽样方法[J]. 通信学报，2009，30(11)：27-36

[11] Damashek M. Gauging similarity with N-grams: language independent categorization of text[J]. Science, 1995, 267: 843-848

[12] Davis J J, Clark A J. Data preprocessing for anomaly based network intrusion detection: a review[J]. Computers&Security, 2011, 30(6/7): 353-375

[13] Lippmann R, Haines J W, Fried D J, et al. The 1999 DARPA off-line intrusion detection evaluation[J]. Computer Networks, 2000, 34(4): 579-595

[14] Ingham K, Inoue H. Comparing anomaly detection techniques for HTTP[M]// Recent Advances in Intrusion Detection. Springer, 2007: 42-62

[15] Wang K, Stolfo S. Anomalous payload-based network intrusion detection[M]// Recent Advances in Intrusion Detection. Springer, 2004: 203-222

[16] Perdisci R, Ariu D, Fogla P, et al. McPAD: a multiple classifier system for accurate payload-based anomaly detection[J]. Computer Networks, 2009, 53(6): 864-881

[17] 孙卫，宋连涛，庄卫华. 基于有效载荷的异常入侵检测技术研究[J]. 计算机工程与设计，2009，30(23)：5348-535

[18] Jamdagni A, Tan Z, Nanda P, et al. Intrusion detection using geometrical structure[C]// Proceedings of the Fourth International Conference on Frontier of Computer Science and Technology. 2009: 327-333

[19] Bolzoni D, Etalle S, Hartel P. POSEIDON: a 2-tier anomaly-based network intrusion detection system[C]// Proceedings of the Fourth IEEE International Workshop on Information Assurance. 2006: 156-165

[20] Wang K, Parekh J J, Stolfo S J. Anagram: a content anomaly detector resistant to mimicry attack[C]// Proceedings of the 9th International Conference on Recent Advances in Intrusion Detection. 2006: 226-248

[21] Rieck K, Laskov P. Language models for detection of unknown attacks in network traffic[J]. Journal in Computer Virology, 2007, 2(4): 243-256

[22] Chu Y M. Deep packet inspection in network intrusion detection and prevention systems[D]. Institute of Communications Engineering, National Tsing Hua University, 2010

[23] Porter T. The Perils of Deep Packet Inspection[DB/OL]. http://www.securityfocus.com, 2005-1-11

[24] Yu F. High speed deep packet inspection with hardware support [D]. Berkeley: EECS Department, University of California, 2006

[25] Jolliffe I. Principal Component Analysis [DB/OL]. http://onlinelibrary.wiley.com, 2013-6-17

[26] Bouzida Y, Cuppens F, Cuppens-Boulahia N, et al. Efficient intrusion detection using principal component analysis[C]// Proceedings of the 3me Conference sur la Scurit et Architectures Rseaux (SAR). 2004

[27] Bouzida Y, Gombault S. Eigenconnections to Intrusion Detection [M]// Security and Protection in Information Processing Systems. Springer, 2004: 241-258

[28] Wang W, Guan X, Zhang X. Processing of massive audit data streams for real-time anomaly intrusion detection[J]. Computer Communications, 2008, 31(1): 58-72

[29] Nwanze N, Sun-il K, Summerville D H. Payload modeling for network intrusion detection systems[C]// Proceedings of the Military Communications Conference. 2009: 1-7

[30] Liao Y, Vemuri V R. Using text categorization techniques for intrusion detection[C]// Proceedings of the 11th USENIX Security Symposium. 2002: 51-59

[31] Nelson L R. Some observations on the scree test, and on coefficient alpha[J]. Thai Journal of Educational Research and Measurement, 2005, 3(1): 1-17

[32] Cattell R B. The scree test for the number of factors[J]. Multivariate Behavioral Research, 1966, 1(2): 245-276

[33] Jamdagni A, Tan Z, Nanda P, et al. Intrusion detection using GSAD model for HTTP traffic on web services[C]// Proceedings of the 6th International Wireless Communications and Mobile Computing Conference. 2010: 1193-1197

[34] Tan Z, Jamdagni A, He X, et al. Network Intrusion Detection based on LDA for Payload Feature Selection[C]// Proceedings of the GLOBECOM Workshops (GC Wkshps). 2010: 1545-1549

[35] Chapell L. Wireshark Network Analysis: The Official Wireshark Certified Network Analyst Study Guide[M]. California: Laura Chappell University, 2010