大型分布式入侵检测系统

杨瑞增[1],陈天鹰[2],李玉盼[3]
(1.华北计算机系统工程研究所,北京 100083;
2.中国铁道科学研究院 研究生院,北京 100081;3.北京交通大学,北京 100044)

摘 要: 提出一种大型分布式入侵检测系统(Broad-scale Distributed Intrusion Detection System,BDIDS)的体系结构,以发现多手段多层次的攻击。这些攻击是分布式网络中多个子网之间存在的异常现象。BDIDS 由两个关键组件组成:大数据处理引擎和分析引擎。大数据处理是通过 HAMR 完成的,HAMR 是下一代内存 MapReduce 引擎。据报告,HAMR 通过多种分析算法,使得现有大数据解决方案的速度大大提高。分析引擎包括一种新颖的集成算法,该算法从多个 IDS 警报的集群中提取训练数据。基于聚类与已知潜在攻击的高度相似性,将聚类用作预处理步骤以重新标记数据集。总体目标是预测分布在多个子网中的多手段多层次的攻击,这些攻击手段如果不以综合方式进行评估,极有可能会被忽略。

关键词: 大数据;分布式入侵检测系统;集成学习
中图分类号: TP393 **文献标识码:** A **DOI:** 10.19358/j.issn.2096-5133.2020.07.005
引用格式: 杨瑞增,陈天鹰,李玉盼. 大型分布式入侵检测系统[J].信息技术与网络安全,2020,39(7):31-35.

Broad-scale distributed intrusion detection system

Yang Ruizeng[1], Chen Tianying[2], Li Yupan[3]
(1.National Computer System Engineering Research Institute of China, Beijing 100083, China;
2.Graduate School, China Academy of Railway Sciences, Beijing 100081, China;
3.Beijing Jiaotong University, Beijing 100044, China)

Abstract: In this paper, a large-scale distributed intrusion detection system (broad-scale distributed intrusion detection system, BDIDS) architecture is proposed to discover multi-level and multi-means attacks. These attacks are anomalies that exist between multiple subnets in a distributed network. BDIDS consists of two key components: big data processing engine and analysis engine. Big data processing is done through HAMR, which is the next-generation in-memory MapReduce engine. According to reports, HAMR has greatly improved the speed of existing big data solutions through various analysis algorithms. The analysis engine includes a novel integrated algorithm that extracts training data from a cluster of multiple IDS alerts. Based on the high similarity between clustering and known potential attacks, clustering is used as a preprocessing step to relabel the data set. The overall goal is to predict multi-method, multi-level attacks distributed in multiple subnets. If these attacks are not evaluated in a comprehensive manner, they will most likely be ignored.
Key words: big data; distributed intrusion detection system; integrated learning

0 引言

入侵检测旨在使用已知的攻击特征来识别未经授权的访问。入侵检测的重点是发现多手段多层次的攻击,这些攻击可能会随着时间的流逝借助复杂网络中各个点而传播[1]。特别是随着数据集变得庞大,多手段多层次的攻击检测是一项具有挑战性的任务。

2011 年 7 月在太平洋西北国家实验室曾经发生过一次复杂的多手段网络攻击事件。尽管实验室的 IT 安全边界得到了很好的保护,但这些攻击却是在非常协调和长期的过程中完成的。首先是对组织的攻击,其次是对共享关键资源的合作伙伴的攻击。在攻击的第一部分中,入侵者利用了面向公众的 Web 服务器中的漏洞[2]。此外,黑客还秘密地从受攻击

的工作站中搜索了网络，这些工作站已作为长期协调攻击的一部分而被预先锁定。攻击的第二部分始于鱼叉式网络钓鱼，第二组黑客对组织的主要业务合作伙伴发起了网络钓鱼攻击，并与之共享网络资源。黑客能够获得特权账户并破坏由组织及其合作伙伴共享的根域控制器。当入侵者试图重新创建和分配特权时，警报最终被触发，以警告组织的网络安全团队[3]。

如上述示例所示，在这样长时间的多源攻击情形下，仅查看数据的一个维度是不够的。其缺点在深度威胁检测中暴露也很明显。于是分布式入侵检测系统的概念被引入，该系统提供了用于检测针对组织及其合作伙伴的分布式网络资源的协同攻击的基础结构。鉴于多种攻击源的复杂性以及针对这种多手段多层次的攻击生成的大量数据，本文提出了一种大型分布式IDS(BDIDS)形式的多级挖掘框架。在大多数组织中收集的重要数据之一是IDS日志数据，例如Snort日志。本文使用IDS日志来筛选可能看上去良性的警报，但良性警报也可能会与其他警报一起指示严重警报[4]。

在本文建议的分布式环境中，每个子网都包含一个DIDS代理，该代理执行本地入侵检测并生成IDS日志数据。来自每个DIDS代理的日志数据被发送到控制中心，并在其中进行汇总分析。当检测到针对已知威胁的攻击时，每个基于签名的代理都会生成与警报关联的优先级，并针对其他“异常”行为生成高、中和低优先级警报。对于高优先级警报，可以清楚地被标记，但是中低优先级警报数据非常大，使得管理员难以执行手动分析[5]。在具有高流量的大型网络中，此数据可能更大。系统管理员通过查询查看警报数据，以检测网络中的可疑行为。但是，在这样的审查中，作为协同攻击的一部分的多个警报将被遗漏。

本文认为这是一类不平衡的学习问题。本文使用集成分类技术自动对大量汇总的警报数据进行分类，并向系统管理员警告潜在的协同攻击。本文认为每个代理都提供一个训练集，该训练集是在通过聚类算法对数据进行预处理之后生成的。使用拆分比率(Split Ratio，SR)从每个聚类中选择训练元组，以使训练集由相对于聚类质心的近、远或离群的元组组成。这种有针对性的选择在样本多样性较高的情况下(如入侵检测为少数)产生了高度准确的结果。在这项研究中，本文使用先进的大数据处理工具探索机器学习技术的优势。

本文使用HAMR处理分布式IDS传感器内部和传感器之间的海量数据集，这是由HAMRTech(HAMR Analytic Technologies)开发的下一代内存MapReduce引擎，无缝支持批处理和流分析。在执行机器学习算法时，HAMR支持MapReduce编程模型，并通过Hadoop产生高速度数据处理[6]。

1 本文方法简介

本节简要概述了本文提出的用于挖掘多个IDS警报的方法，这些警报是从多个子网提交的，以预测多手段多层次的攻击，如图1所示。本架构包括通过HAMR进行的大数据分析处理以及将驻留在HAMR上的分析引擎。

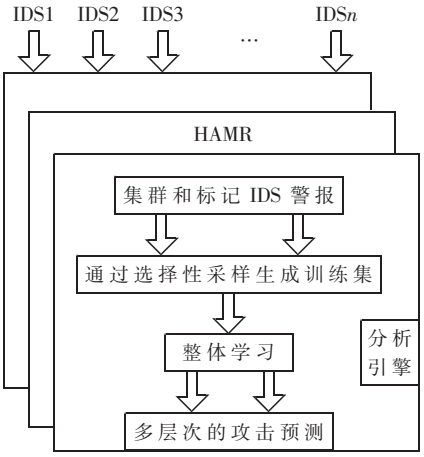

图1 BDIDS控制中心分析

2 分析引擎

2.1 集群和标记IDS警报

预处理后，对IDS日志数据进行聚类。数据采用已解析IDS警报的形式，其中每个警报都是优先级为高、中或低的数据点。如图1所示，该数据是从多个IDS来源收集的。本文架构的前提是，与来自不同IDS的其他一些警报一起查看时，低优先级警报可能表示协同攻击。由于本文方法可以查看来自多个IDS的所有警报，因此有机会研究警报之间的相似性，然后判断警报是真正的低优先级还是潜在的高优先级。最终目标是在多个IDS之间折叠并重新分类相关警报，这些警报可能表明存在网络攻击。

所有警报都收集在控制子网中，并对其进行聚类。应当注意，传统的聚类可以对数字属性和分类属性分别执行[7]。根据该思想，如果低优先级警报

“l”与多个高优先级警报“h”属于同一群集 C^h，则IDS很有可能将警报“l”标记为错误。这可以通过以下方式进一步验证：(1)“l”与高优先级群集 C^h 的群集质心的距离很小，这表明“l”确实靠近质心，因此与 C^h 中的其他“h”警报高度相似。(2)群集 C^h 的总体平方误差(Sum of Squared Error，SSE)将指示群集的质量，高SSE表示数据点在质心周围广泛散布，而低SSE表示数据点在质心周围紧密编织。较低的SSE将更有效地证明本文声称“l”确实贴错了标签。另外，可以执行关联以识别具有高关联度的数据点，以增加检测错误分类的警报的可能性，尤其是在低优先级警报与高优先级警报一致地关联的情况下。

如果这些多次验证或多数验证都肯定错误地标记了“l”，那么本文将为该警报提供一个新标签“p”，表示该警报肯定表示攻击。其他未通过此测试的警报被标记为“n”，表示它们是负数据点。本文将这些新标签用作分类集合，以便根据本文从多个IDS衍生的元知识，将任何新传入的警报预测为“p”或“n”。

2.2 训练集生成

为了创建训练集，本文将使用数据集中点分布的知识。为此，本文使用聚类并提取接近质心的点、远离质心的点和似乎离群的点[8]。这将使本文能够很好地表示与聚类质心和异常值非常相似的数据点(警报)。与其他数据点相比，高度相似的数据点模拟群集的平均行为，而离群点则具有极端行为。

2.3 整体学习

本文提出的集合分类器通过创建实例的选择性子集来最大化多样性，这些实例子集彼此相似但与其他子集中的实例不同。每个子集都是从实例创建的，这些实例已经足够相似，可以通过k均值聚类放入相同的集群中。研究表明，总体的多样性对类不平衡表现有积极影响。因此，本文旨在通过集合中的多个不重叠的训练集来实现高度多样性。

本文创建两个名为Near和Far的训练集。这些是根据拆分率定义的，拆分率显示了应进入“近”或“远”训练集的每个群集中示例的百分比。例如，拆分比率(SR)值为40%，表明最接近群集中心的示例中有40%进入了近距离训练集中，其余60%进入了远距离训练集中。接下来，使用每个训练集训练一个弱分类器。然后，每个受过训练的分类器用于对相同的不重叠测试集进行分类。分别针对每个分类器计算总体绩效指标。最后，使用投票系统将分类器的预测结合起来。根据每个分类器的整体表现对它们进行加权，然后输出预测的分类器标签。

3 大数据处理

模型开发和分类器都需要大数据解决方案，但是出于不同的原因。在模型开发阶段，静态数据集将包含在几年内从许多IDS代理收集的IDS警报数据中，并且可能在10 TB范围内。数据量与执行模型训练所需的复杂算法相结合，就需要分布式的内存解决方案，以便模型开发在合理的时间内完成。但是在实时系统中，分布式IDS代理将向中央分析服务器发送警报。实时系统中收集处理警报数据的速度还需要分布式内存解决方案，以使系统跟上峰值负载。

本文使用HAMR来处理分布式IDS传感器之内和之间的海量数据集。HAMR是由HAMR Analytic Technologies(HAMRTech)开发的下一代内存MapReduce引擎，无缝支持批处理和流分析[9]。HAMR在执行机器学习算法时支持熟悉的MapReduce编程模型。机器学习的两个阶段(模型训练和实时分类)与HAMR的组件库无缝集成。在HAMR基准测试报告的初步分析中，HAMR在各种分析算法中产生了多个加速顺序。表1显示了在各种分析中对类似大数据产品的加速。HAMR展示了使用朴素的贝叶斯训练算法可以高倍提高Hadoop和Mahout的速度，这表明它是当前工作的有效可扩展解决方案。此外，HAMR证明其延迟是流行的实时流引擎Apache Storm的2.6倍。表1进一步显示，HAMR的性能比Spark提高了7倍，并且能够处理内存中10倍以上的数据。

本文建议在批处理模式下使用HAMR来生成具有多个训练数据集的模型集合。该集合与HAMR实时引擎一起将被集成到BDIDS中，以对汇总的警报数据进行分类，从而向系统管理员提供汇总警报，以高精度地警告潜在的针对网络资源的协同攻击。

4 Hadoop与HAMR性能比较试验

本次实验采用4台计算机搭建的集群，其中1个主节点3个从节点。试验数据由HiBench Benchmark Suite 4.06版本生成。运行在HAMR上的PageRank算法代码包括在HAMR0.4.1版本中。

Hadoop上运行PageRank基本思想是使用一个

表 1 HAMR 对各类算法加速概述

基准	最小加速倍数	数据集大小	中等加速倍数	最大加速倍数	数据集大小	最大数据集加速倍数	最大数据集大小
WordCount1	1.7	8 GB	2.6	3.82	256 GB	3.8	256 GB
PageRank(Hadoop)2	12.4	4M Pages	13.3	15.7	2M Pages	12.4	32M Pages
PageRank(Spark)3	3.6	2M Pages	4.9	7	8M Pages	Spark OOM	32M Pages
Classification1	1	4 GB	1.3	1.8	256 GB	1.8	256 GB
K-Cliques(K=3)1	6.2	211 Vertices	10.4	14.3	216 Vertices	11.4	219 Vertices
NaiveBays Trainer5	12.2	100K Pages	39.2	87.1	3.2M Pages	87.1	3.2M pages
Grep1	1.4	256 GB	2.1	3.6	64 GB	1.4	256 GB
Histogram1	1.1	4 GB	1.5	2	128 GB	1.8	256 GB
Latency(Storm)4	1.2	64K Message	1.7	2.6	4K Message	1.2	64K Message

MapReduce 过程作为 PageRank 的一个迭代。每次迭代中,Map 输入值为单位网页,输出值为当前 PageRank 值。每次迭代过程分为两个阶段。阶段一,每个网页将当前 PR 值与连接数的比值分配给每个指向其他网页的链接,该过程由映射函数实现;阶段二,每个网页统计指向自己链接携带的 PR 值,该聚合过程由 Reduce 函数实现。

HAMR 上运行 PageRank 基本思想:在初始化阶段,从 HDFS 上读取输入文件;然后创建图表 Key-ValueStore,再初始化 RanksKeyValueStore;接下来执行迭代算法。迭代中,每个页面的 PR 值为所有指向其链接的 PR 值之和。一旦所有页面被遍历,迭代更新保存 PR 值得 KeyValueStore。为了保持 HAMR 的稳定,固定迭代次数。

实验输入数据集为 200 万~300 万网页,输入数据从 1 GB~20 GB 大小不等。每个数据集执行5 次迭代。

运行时间比较见图 2。随着输入数据增大,HAMR 优势更加明显。当输入数据集较小时,HAMR 的内存使用率保持稳定;随着数据集增大,HAMR 内存使用率明显增高,总体而言内存使用率高于 Hadoop。HAMR 在每个节点中具有比 Hadoop 高的吞吐量。当输入数据集变大时,HAMR 展示出比 Hadoop 更好的自适应特性,见图 3。

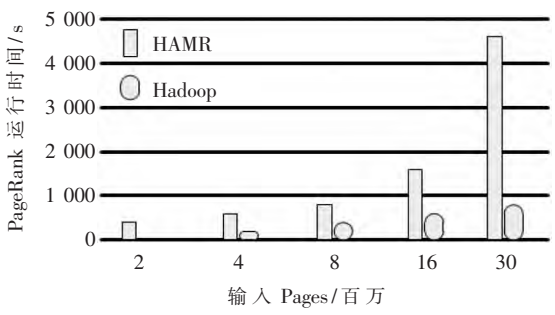

图 2 运行时间比较

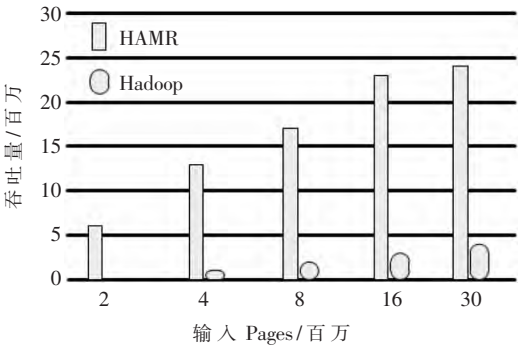

图 3 包通过量

5 结论

本文提出了一种大型分布式入侵检测系统,在此架构中,利用了大数据处理引擎 HAMR,并提出了一种新颖的集成方法来识别多手段多层次的攻击。本文计划在公共数据集上进行广泛的实验,并就提速和结果质量,针对现有的大数据解决方案提供基准。

参考文献

[1] NAMAYANJA J M, JANEJA V P. Discovery of persistent threat structures through temporal and Geo-Spatial characterization in evolving networks[C]. IEEE International Conference on Intelligence and Security Informatics, 2013.

[2] CHEN S, JANEJA V P. Human perspective to anomaly detection for cybersecurity[J]. Journal of Intelligent Information Systems, 2014, 42(1): 133-153.

[3] AZARI A, JANEJA V P, MOHSENI A. Healthcare data mining: predicting hospital length of stay(PHLOS)[J]. International Journal of Knowledge Discovery in Bioin-

formatics, 2012, 25(4): 50-53.

[4] HEILIG B, TURNER S, COLLIER R, et al. Beyond MapReduce: the next generation of big data analytics[C]. 2014 Asebigdata/Socialcom/Cybersecurity Conference, Stanford University, 2014.

[5] WANG S, YAO X. Relationships between diversity of classification ensembles and single-class performance measures[C]. IEEE Transactions on Knowledge and Data Engineering, 2013, 25(1): 206-219.

[6] 吴晓平, 周舟, 李洪成. Spark 框架下基于无指导学习环境的网络流量异常检测研究与实现[J]. 信息网络安全, 2016(6): 22-23.

[7] 陈虹君. 基于 Hadoop 平台的 Spark 框架研究[J]. 电脑知识与技术, 2014(35): 21-23.

[8] 黎文阳. 大数据处理模型 Apache Spark 研究[J]. 现代计算机(专业版), 2015(8): 3-5.

[9] BLEI D M. Probabilistic topic models[J]. Communications of the ACM, 2012, 55(4): 77-84.

(收稿日期: 2020-04-06)

作者简介:

杨瑞增(1994-), 男, 硕士研究生, 主要研究方向: 计算机科学与技术。

陈天鹰(1963-), 男, 硕士, 高级工程师, 主要研究方向: 网络安全、轨道交通自动化与控制。