利用潜在语义分析检测恶意命令行为

乔纳森伍德布里奇 (/Our-Experts/Jonathan-Woodbridge) 2016年2月17日

检测异常行为仍然是安全性最具影响力的数据科学挑战之一。大多数方法都依赖基于签名的技术，这种技术本质上是反动的，无法预测恶意行为的新模式和现代对抗技术。相反，作为入侵检测研究的关键组成部分，我将关注使用基于机器学习的方法进行命令行异常检测。基于命令行历史记录的模型可能会检测到一系列异常行为，包括使用被盗凭证和内部威胁的入侵者。命令行包含丰富的信息，并作为用户意图的有效代理。用户对命令有自己的离散偏好，可以使用无监督机器学习和自然语言处理的组合进行建模。我演示了模拟离散命令的能力，突出了正常行为，同时还检测可能指示入侵的异常值。这种方法可以在不需要大量资源或领域专业知识的情况下帮助进行规模异常检测。

一点介绍材料

在深入研究模型之前，快速解决以前的研究，模型的假设以及其关键部分很有帮助。以前的一些工作只关注命令，而另一些则使用命令的参数来创建更丰富的数据集。我只关注命令并留下未来工作的论据。另外，这项工作专注于服务器资源，而不是个人计算机，其中命令行通常不是与机器交互的主要手段。由于我们专注于企业级安全性，因此我将这种个人计算机模型的应用程序留给未来的工作。由于当前的数据可用性，我也专注于UNIX / Linux / BSD机器。

以前工作中的作者常常依赖于他们的命令集的唯一性。对于（过于简单）的示例，开发人员A使用emacs，而开发人员B使用vi，因此如果用户A使用vi，则它是异常的。这些作品以多种形式出现，包括序列比对（类似于生物信息学），命令频率比较以及转换模型（如隐马尔可夫模型）。这些作品中的一个共同问题是维数的爆炸式增长。为了说明这一点，您可以从命令行输入多少个命令？我的OS X机器有大约2000个命令。现在添加Linux，Windows和所有不常见或自定义命令。这可以轻松增长到数以万计的命令（或维度）！

除了维度挑战之外，数据表示还会进一步影响数据环境的复杂性。有很多方法来表示一堆命令序列。最简单的就是把它们保持为字符串。字符串可以用于某些算法，但可能缺乏效率和泛化。例如，高斯分布假设对于字符串并不真正起作用。另外，将字符串插入需要数学运算符（如矩阵乘法）（即神经网络）的复杂模型中是行不通的。通常，人们使用单热编码(https://en.wikipedia.org/wiki/One-hot)为了使用具有标称数据的更复杂的模型，但是随着唯一名称数目的增加，这仍然受到维度的诅咒。另外，单热编码将每个独特的分类值视为完全独立于其他值。当然，这在分类命令行时并不是一个准确的假设。

幸运的是，降维算法可以抵消由单热编码引起的尺寸不断增加的问题。 主成分分析（PCA）(https://en.wikipedia.org/wiki/Principal_component_analysis)是最常用的数据缩减技术之一，但单热编码并不遵循高斯分布（PCA将为其优化地减少数据）。另一种技术是二进制编码(https://en.wikipedia.org/wiki/Truncated_binary_encoding)。这种技术是通用的，使其易于使用，但由于不考虑特定领域的知识，因此可能会在性能上受到影响。当然，二进制编码通常用于压缩，但实际上，在将每个比特视为特征时，对分类变量进行编码时效果相当好。

那么我们如何利用领域知识来压缩维度的数量，以便从分类器中挖掘出最佳性能？我在这里介绍的一个答案是潜在语义分析 (https://en.wikipedia.org/wiki/Latent_semantic_analysis)或LSA（也称为潜在语义索引或LSI）。LSA是一种接受大量文档（数千甚至更多）并通过奇异值分解（SVD）

为每个文档分配“主题”的技术。LSA是一个成熟的技术，在许多其他领域中被大量使用（意味着许多开源！）。为了生成主题，我为每个命令使用手册页和其他文档。

假设（或假设）是我们可以将命令表示为代表用户意图的一组有限且重叠的主题的分布，并且可以用于检测异常行为。对于一个重叠的例子，可以使用cp（copy）和rm（delete）来模拟mv（或move）。或者，从我们之前的例子来看，emacs和vi的功能基本相同，可能重叠很多。

命令行上的LSA

为了检验假设，我需要评估LSA如何使用手册页中的文本将命令组织成主题。我使用大约3100个命令（及其各自的手册页）来训练我的LSA模型。接下来，我将使用前50个最常用的命令，并显示它们如何与使用余弦相似性的 (https://en.wikipedia.org/wiki/Cosine_similarity)其他命令进行集群。我可以想象更加的命令，但目的是显示命令的连贯和理解集群（这样你就不必运行 *男人* 百倍了解图形）。类似地，只有权重大于.8的边才被保留用于可视化目的（余弦相似性以[0,1]为边界，而1为最相似的边界）。

如果仔细观察，可以看到类似命令的集群。这完全是无人监督的。没有领域的专家。这很酷！

这是一个很好的第一步，但我们如何使用它来对命令行进行分类？这个想法是平均意图在小窗口的命令（如三，十或五十命令），并将其用作特征向量。例如，如果用户输入 *cd*，*ls*，*cat*，我们从它们相应的手册页中找到每个命令的LSA表示。假设我们使用200个主题对命令进行建模，我们取三个200点特征向量中的每一个，并做一个简单的均值，以获得这三个命令的一个200点特征向量。我尝试了其他一些简单的方法来组合特征向量，例如连接，但发现平均作品是最好的。当然，也许有更好的更先进的技术，但这是留给未来的工作。我们可以通过在用户的命令序列上应用滑动窗口来生成大型训练和测试集。为了好玩，我使用sklearn (http://scikit-learn.org/stable/modules/svm.html#svm-outlier-detection)的单类SVM (http://scikit-learn.org/stable/modules/svm.html#svm-outlier-detection)并从11位同事的命令行历史中采用数据。我为每个用户创建了11个模型。这些都是一类模型，因此在任何培训中都没有出现正面（即异常）的例子。我使用此设置运行十次并平均结果。对于每一次折叠，我都会训练50％的数据，并保留每位用户所有命令的50％进行测试。我承认这种设置并不完全代表真实世界的部署，因为

异常命令序列的数量远远超过正常数量。我还做了最基本的预处理，例如在运行LSA创建主题之前，在手册页上使用NLTK (http://www.nltk.org/)和stop_words (http://pypi.python.org/pypi/stop-words)（可以通过pip安装）停用词的删除和删除。

对于基线，我针对每个命令使用单热，二进制和PCA编码的特征向量运行相同的实验。我将这些特征向量的均值作为我之前做的。

我在三个，十个和五十个窗口上运行实验，并显示相应的接收器操作特性（ROC）。ROC曲线描述了十一个用户模型如何识别保留的命令。一个警告是，并非所有的命令都在手册页中表示。为了简单和可重复性，我目前忽略这些命令并将其留待将来工作。

第一个图像不是很好。在这里，我们显示的窗口大小为3的ROC。除了PCA，一切都差不多。LSA稍好于单热和二进制编码。但是，如果窗口尺寸很小，则最好使用PCA。

随着窗口大小的增加，结果会变得更有趣。单热和LSA编码在性能上获得最大提升，而PCA降低。正如我前面所说，PCA是减少分类变量的不错选择，所以这种下降并不令人吃惊。另一个有趣的地方是，较大的窗户可以制作出更好的分类器。这也不是很令人惊讶，因为本研究中的用户在使用模式上非常相似。较大的窗口包含更多的上下文，从而提供更多的信息特征向量。

对于窗口大小为50的LSA，结果会更好。当然，我们可以通过命令行参数丰富我们的功能，并且可能会获得更好的结果，但我们已经在使用这些命令时表现得非常好。

最后的想法

LSA在聚集命令行参数方面工作得很好，作为用户意图的有用代理，更重要的是检测异常行为。这是完全无人监督的，使模型容易适用于标签通常不存在的真实世界部署。这篇文章的一个假设是训练数据没有被污染（即，不包含来自其他用户的命令行序列）。此外，这些数据来自软件开发人员或研究人员的使用模式非常相似的命令行。这意味着一个命令行模式可能在几个用户中很常见，导致在这个实验设置中出现假阴性。因此，我们可能会看得***更好***当我们将恶意用户的命令行提供给普通用户的模型时会出现这种结果。另外，我们可以通过使用多个普通用户的命令历史记录（而不是从单个用户构建模型）来创建更健壮的模型。我将把这些问题的答案留给另一篇文章！

相关文章

(/blog/technical-blog/endpoint-malware-detection-hunt-real-world-considerations)

(/blog/technical-blog/endpoint-malware-detection-hunt-real-world-considerations)

2016年8月14日搜索的

端点恶意软件检测：真实世界的注意事项 (/blog/technical-blog/endpoint-malware-detection-hunt-real-world-considerations)

查看详情 (/BLOG/TECHNICAL-BLOG/ENDPOINT-MALWARE-DETECTION-HUNT-I

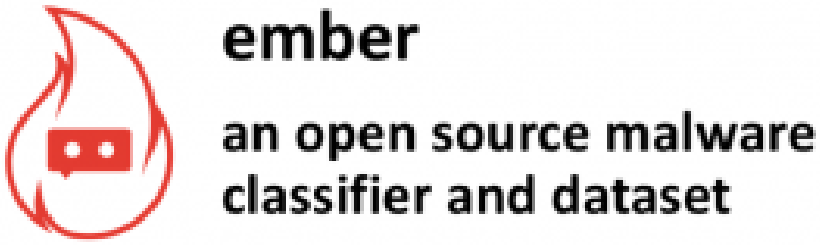
(/blog/technical-blog/stopping-certified-malware)

2017年11月9日

停止认证的恶意软件 (/blog/technical-blog/stopping-certified-malware)

查看详情 (/BLOG/TECHNICAL-BLOG/STOPPING-CERTIFIED-MALWARE)

(/blog/technical-blog/introducing-ember-open-source-classifier-and-dataset)



(/blog/technical-blog/introducing-ember-open-source-classifier-and-dataset)

2018年4月16日

引入Ember：开源分类器和数据集 (/blog/technical-blog/introducing-ember-open-source-classifier-and-dataset)

查看详情 (/BLOG/TECHNICAL-BLOG/INTRODUCING-EMBER-OPEN-SOURCE-CL

703.650.1250 (tel:703-650-1250)

(/blog/technical-blog/using-deep-learning-detect-dgas)

2016年11月18日

使用深度学习检测DGA (/blog/technical-blog/using-deep-learning-detect-dgas)

查看详情 (/BLOG/TECHNICAL-BLOG/USING-DEEP-LEARNING-DETECT-DGAS)

ALL (/BLOG) TECHNICAL BLOG (/BLOG/TECHNICAL-BLOG)

EXECUTIVE BLOG (/BLOG/EXECUTIVE-BLOG)

请求演示 (HTTP://PAGES.ENDGAME.COM/REQUEST-DEMO-WEBSIT

请求演示 (http://pages.endgame.com/request-demo-website.html) 博客 (/blog)

保持联系。

703.650.1250 (tel:703-650-1250)

请求演示 (HTTP://PAGES.ENDGAME.COM/REQ

联系我们 (/COMPANY/CONTACT)

招贤纳士 (/CAREERS)

联系我们

703-650-1250

注册我们的通讯

电子邮件地址

提交

探索

E.HTML)

为什么是残局 (/why-endgame)

平台 (/platform)

公司 (/company)

资源 (/resource)

新闻 (/news)

请求演示 (http://pages.endgame.com/request-demo-website.html) 博客 (/blog)

703.650.1250 (tel:703-650-1250)

网络安全字典

威胁狩猎 (/resource/solution-brief/pdf/automated-threat-hunting)

安保服务 (/endgame-services)

勒索 (/blog/technical-blog/wcrywanacry-ransomware-technical-analysis)

机器学习 (/blog/machine-learning/you-gotta-tame-beast-you-let-it-out-its-cage)

安全新闻 (/news)

端点保护 (/why-endgame)

网络狩猎 (/blog/technical-blog/hovering-hunt-file-path-less-traveled)

网络安全金融服务 (/resource/solution-brief/endgame-financial-services-booklet)

自动寻线 (/resource/white-paper/sans-white-paper-automating-hunt-hidden-threats)

网络攻击 (/blog/hackers-guide-not-having-your-passwords-stolen)

连

3101威尔逊大道

阿灵顿，VA 22201

703-650-1250 (tel:703-650-1250)

联系我们 (https://www.endgame.com/contact)

请求演示 (http://pages.endgame.com/request-demo-website.html)

(https://www.linkedin.com/company/endgame-inc) (https://twitter.com/endgame) (https://www.facebook.com/endgame)

(/)