基于信息回馈检测技术的Snort优化研究

李丽[1,2] 钟求喜[1] 杨智丹[2,3]
1 国防科学技术大学计算机学院 湖南 410073
2 66019部队 北京 100041
3 解放军电子工程学院网络工程系 安徽 230037

摘要：Snort产生大量告警信息使得用户难以提取到真正入侵事件。本文借鉴被动攻击验证技术，针对Snort提出一个基于信息回馈的检测手段，在规则匹配成功后，通过被动监听攻击会话和反馈信息，以检测出成功的入侵事件。

关键词：入侵检测；Snort；信息回馈；告警

0 引言

入侵检测系统IDS(Intrusion Detection System)是网络安全深层次防御体系结构中的重要环节，用来探测攻击或对系统、网络和相关资源未经许可的使用，发现后对相关行为发出告警或予以制止。从IDS对事件的不同分析方法来看，商用IDS中使用最广泛的技术是特征检测，即攻击库中包含已知攻击的行为特征，利用特征匹配来识别攻击。但是，基于特征的IDS存在高虚警率。其中，攻击已实施但系统无漏洞的失败攻击被报出和正常数据包恰恰与某攻击特征相吻合是产生误报的两个主要原因。

Snort是一个开放源码的跨平台的网络入侵检测系统，是基于特征检测的IDS。在很多情况下，不论攻击成功与否，一发现匹配就形成结论，而网络环境越开放，得到的报警信息就越多。对于有些网络维护，只须关注与本网络相关的真正的攻击行为，及时向网络管理员报告，采取补救措施，如断开连接、防止错误数据蔓延、向入侵者发出警告等。因此，系统需要降低虚警率，使真正的攻击行为显露出来。

文献[1]指出，大多数网络应用协议，都有反馈机制，以告知数据接受和处理的结果。而对于很多攻击，其攻击成功与攻击失败两种情况下反馈信息也大不相同，可以利用反馈信息改进误报问题。

1 Snort内部机理

Snort的结构由几大软件模块组成，数据包嗅探器、预处理器、检测引擎和报警输出模块。这些模块都是插件结构，插件程序按照Snort提供的插件函数接口完成，使用时动态加载，在不用修改核心代码的前提下让Snort的功能和复杂性扩展更容易。Snort体系结构如图1所示 。

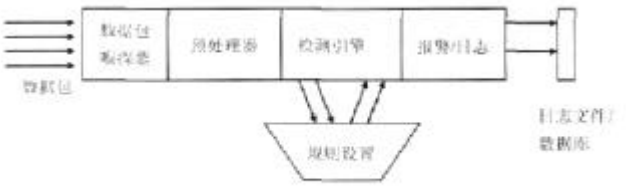

图1 Snort体系结构图

启动Snort后，先初始化设置一些缺省值，解析命令行参数，进行一系列的设置和配置。打开libpcap包捕获接口，若是捕获网络上的数据包，则将函数参数设为网卡接口。接下来根据数据链路网络的不同，确定要使用的数据包解析函数，并将其保存在函数指针中。初始化各种预处理器插件，处理插件，输出插件，读取规则文件，解析规则库，生成三维规则链表。设定报警函数和日志函数。最后进入循环抓包进行检测，即数据包处理阶段，这是一个循环调用过程。

当pcap从网卡驱动接收数据的时候，对数据链路层的原始数据包进行解码。在解码过程中，依次调用不同的函数对不同协议进行解码。并将分析结果存入到相应的数据结构中，为后续的预处理和检测引擎的分析做准备。接下来，系统调用预处理程序，写预处理器是为了解决那些直接的规则匹配不能完成的问题。包重组预处理器帮助Snort检测匹配数据分布在多个包中的攻击。协议解码预处理器对协议数据进行处理，使串匹配功能在更明确的数据上工作。异常检测预处理器使Snort不用进行彻底的重新设计就能扩展检测方法，所以Snort可以灵活地使用各种检测模型。

然后进入检测引擎，规则中的每个关键字选项对应于检测引擎插件，能够提供不同的检测功能。通过各种规则文件中的不同选项来对每个包的特征和报信息进行检测，做出是

作者简介：李丽(1980-)，女，国防科学技术大学计算机学院工程硕士，研究方向：网络安全、入侵检测。钟求喜(1969-)，男，副研究员，博士，硕士生导师，研究方向：网络与信息安全。杨智丹(1980-)，男，工程师，硕士研究生，研究方向：信息隐藏、协议隐写。

否发生入侵行为的判断。如果有符合某条规则的数据包，就会被检测出来，通知报警模块，根据这条规则所定义的响应方式以及输出模块的初始化定义情况，选择进行各种方式的日志记录和报警操作。

2 基于信息回馈的入侵检测技术

在文献[1]中提出的被动攻击验证技术(Passive Attack Verification,PAV)是“基于被动监听,结果判定的攻击验证”技术的简称。PAV技术不主动探测受保护网络的拓扑以及主机的操作系统或服务，而只关心捕获的数据流，检测入侵信息，识别攻击行为。发现攻击后，转入反馈监听模式监听反向数据流，利用许多攻击的攻击成功与攻击失败反馈信息的不同，判断攻击结果，再采取后续措施。PAV技术应用于PAV系统（PAVS），是该系统的核心技术。在PAVS中，要建立攻击特征库以检测是否有攻击或类似攻击的数据包出现；要建立攻击成功反馈库和攻击失败反馈库,以判断攻击是否成功，从而决定是否要发出告警。由协议本身决定的,很多攻击的攻击失败反馈是相同的,所以攻击失败反馈库相比攻击特征库和攻击成功反馈库要简单得多。而要及时而完整地扩充攻击特征库和攻击成功反馈库是一件很繁重的工作。另外，目前PAVS在异常检测技术和DoS攻击验证功能方面还存在弱点。

借鉴PAV思想,可在Snort中采用基于信息回馈的检测方法和手段。基于信息回馈的入侵检测技术即指，根据规则匹配发现包含有攻击特征的数据包后,加入信息回馈检测模块。捕获攻击数据包后，不直接告警，而是监听该连接，检查回馈信息，根据回馈信息判断入侵是否成功，若捕捉到与攻击失败反馈库匹配的回馈信息,则认为攻击失败放弃对该连接的监听，若在预设的时间内没有捕捉到失败反馈信息，则发出告警，以采取后续措施。信息回馈检测模块工作流程如图2所示。

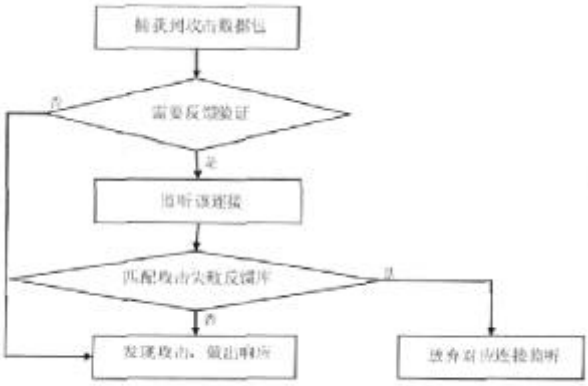

图2 信息回馈检测模块工作流程

3 基于信息回馈检测技术的Snort改进

Snort是一个开放源码的系统,在所有开发者和志愿者的支持下，每个版本都在不停改进，基本每个新版都增强一定的引擎的检测能力，同时在规则集中加入新的选项，利用预处理,Snort已经能进行非规则和协议异常检测,如端口扫描插件、Back Orifice检测插件等。

但是,在原始Snort中的检测引擎后,发现与规则匹配的数据包，直接送到报警模块。使得告警信息中，成功攻击行为与失败攻击，非攻击可疑数据混杂在一起，由此造成的高虚警率，无法迅速定位攻击，不利于网络安全管理。所以提出利用Snort的可扩展性,将信息回馈检测技术引入Snort中，用以降低虚警率。

基于信息回馈检测技术的Snort工作流程如图3所示。刚起动Snort时，反馈验证监听连接开关处于关状态，此时，与原Snort的工作过程一样。当数据包经过预处理器,进入检测引擎,出现与Snort规则匹配的情况,则触发反馈验证监听连接开关，监听该连接状态。数据包经过解析后进入检测方式判断模块，判断是否与触发规则的数据包属于同一连接。如果是与攻击包来自同一连接，进入攻击失败反馈匹配模块，采用特征匹配算法与攻击失败反馈库进行匹配，若有数据包与反馈库中的特征相匹配，则放弃对该连接的监听。使用超时轮询技术，若到超时也没有匹配，则发出告警。在这里，超时时间的设置问题需要考虑，时间过短，有可能还没收到反馈，时间过长的话，如果是成功的攻击则不利于及时采取后续处理措施。如果捕获到的数据包与攻击包不在同一连接上，则进入与Snort规则链表匹配阶段,没有发现匹配就继续进入循环抓取数据包阶段,发现匹配,则触发对应的监听开关,进入新的连接监听。

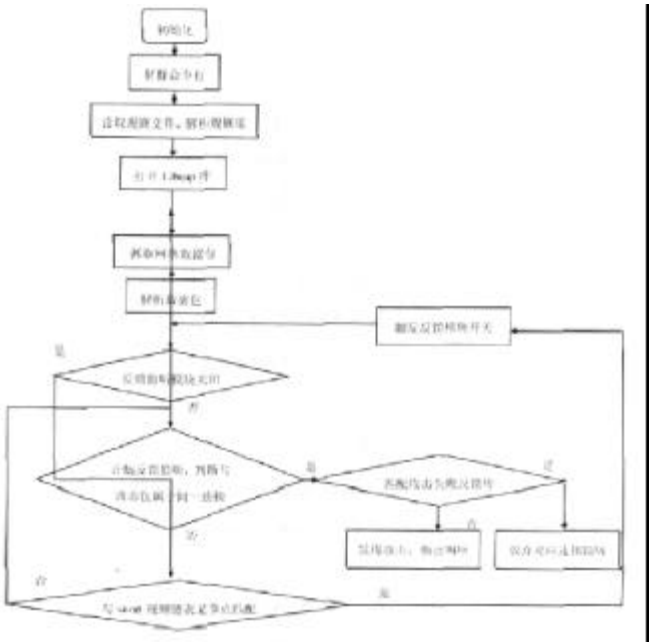

图3 基于信息回馈检测技术的Snort工作流程图

以isapi.ida access攻击为例,说明改进后的Snort工作流程。Snort原本的规则文件web-iis.rules中包含了关于此攻击的检测规则：

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS
$HTTP_PORTS (msg:"WEB-IIS ISAPI .ida access"; flow:to_server,
established;uricontent:".ida";nocase;reference:arachnids,552;
```

reference:bugtraq,1065;reference:cve,2000-0071;classtype:web-application-activity;sid:1242;rev:12;)

在原来的Snort中,发现与上述规则匹配的数据包即发出告警。引入信息回馈检测技术后，发现有数据包与该规则匹配，先触发反馈监听开关，监听该连接，若匹配到攻击失败反馈404 NOT FOUND，则放弃监听不告警，若没有匹配超时，则发出告警。

但是，在同攻击特征不同后果判定方面，只能依靠于Snort原有规则的告警信息,如果也采用PAVS提出的方法,则需要大量的攻击样本试验,而且像PAVS一样,需要其他技术的进一步支持。而对于有些DoS攻击，正是利用连接失败来消耗服务器大量资源，不适合使用信息回馈检测技术，这就要求在与Snort规则匹配后的反馈监听触发方面进一步细化。另外,加入信息回馈检测技术的Snort会在一定程度上降低告警的实时性，需要在准确性和实时性之间进行衡量和抉择，可以考虑的解决方案是加入神经网络等学习算法使得系统始终处在两者中同时令人满意的状态。

4 结束语

基于信息回馈检测技术的Snort,可以由用户决定是否需要进行反馈监听,因为用于某些网络的Snort,被要求记录或者及时通知网络管理员处理任意可能的攻击。此项功能可以由编写新的预处理插件完成，在Snort.conf文件中加入preprocessor<name><options>就可以方便地启动所需的预处理器。利用信息回馈检测技术改进后的Snort可以在一定程度上有效降低虚警率，提高攻击定位速度，为网络维护与网络安全管理提供方便，相比文献[1]中提出的PAVS，只用重新建立攻击失败反馈库，而且可以检测更多类型的攻击。

本文提出了将信息回馈检测技术应用到Snort中的思路，对具体技术的实现进行了讨论。下一步的工作，针对不适合使用信息回馈检测技术的攻击类型,进行更深入的研究探讨，完善实现上的技术细节,使改进后的Snort能够应用于入侵检测实践。

参考文献

[1]庄天舒,田志鸿,张宏莉.基于被动监听的攻击验证技术研究[C].全国网络与信息安全技术研讨会.2007.

[2]蒋建春,马恒太,任党恩.网络安全入侵检测:研究综述[J].软件学报.2000.

[3]BrainCaswell,JayBeale,JamesC.Foster,JeffreyPosluns著,宋劲松等译.Snort 2.0 入侵检测[M].北京:国防工业出版社.2004.

[4]李晓芳,姚远.入侵检测工具Snort的研究与应用[J].计算机应用与软件.2006.

[上接66页]

于其他系统的调度是独立的。应用层调度器就是一个很好的例子。在合作式的情况中，每个网格调度器都有责任执行它自己调度任务,但是所有的调度器是为了一个公共的系统宽度目标。每个调度器的本地策略是与其他的网格调度器合作并作出决策来实现全局目标,而不仅仅是作出策略来使本地服务更有效。在合作式的网格调度算法中，发送者初始化和接收者初始化功能是通过分布式网格调度器来完成的,而不是通过集中式或本地调度器来完成。

3 网格调度研究的新方向

网格有着动态性和异构性等特点,但是这些特点是有等级结构的。目前的资源一般都分布在集群中，在同一个集群中的资源通常有相同的组织域,大部分是同构的并且在一定时期内很少是动态的。在一个集群中，通信成本非常低，同时运行的应用程序也比较少。这些分布特性可能会带来一些新的思想法来应对网格调度的难题,例如可以采取多层次或多阶段的策略,网格调度器首先用一个粗略的调度进行全局调度，然后在一个本地集群中进行精确的调度。

由于网格计算的独特性和调度问题本身的复杂性，在网格调度中还有很多未解决的问题,包括调度体系结构、协议、模型等。仅仅考虑特定环境下的算法并不能满足现实的要求，排除任何特定的假设,可以得出一些网格调度研究的新的方向，例如异构调度算法在网格环境中的改进与应用、利用动态性能预测的算法、适用于性能变化的再调度算法、基于QoS限制的算法、同时混合计算和数据调度的算法等。

4 结论

本文总结了网格计算系统的体系结构和特征,分析了网格任务调度算法的基本原理和性能指标,并对各种调度算法进行了分类和比较，并指出一些新的研究方向。本文对网格任务调度的研究具有很好的参考价值。

参考文献

[1]I.Foster and C. Kesselman (editors),The Grid: Blueprint for a Future Computing Infrastructure.Morgan Kaufmann Publishers. USA.1999.

[2]I Foster, C. Kesselman and S. Tuecke.The Anatomy of the Grid: Enabling Scalable Virtual Organizations. in the International J. Supercomputer Applications. fall 2001.

[3]M. Baker, R. Buyya and D. Laforenza.Grids and Grid Technologies for Wide-area Distributed Computing. in J. of Software-Practice & Experience.December 2002.