从漏洞及攻击分析到NIDS规则设计 PDF转换可能丢失图片或格式，建议阅读原文
https://www.100test.com/kao_ti2020/181/2021_2022__E4_BB_8E_E6_BC_8F_E6_B4_9E_E5_c101_181838.htm 一谈到NIDS，这个产品最为人所诟病的往往就是大量的误报和漏报，满屏乱滚的误报使管理员麻木和厌烦，失去使用的兴趣，漏报则会使管理员怀疑NIDS的检测能力，明明主机已经被入侵了在NIDS的日志中却找不到有用的线索。对于NIDS产品，漏报和误报产生的原因是多方面，但其中最大的来源在于检测规则定义的不严谨。 针对已知的网络攻击，当前主流网络入侵检测检测及防护系统主要还是基于规则的，这是因为特定的攻击，特别是基于特定攻击代码的报文比较容易抽取其报文特征进行匹配，使用规则可以快速方便地实现检测能力的扩展。 网络攻击的本质就是利用目标系统设计或实现上的问题，从被攻击进程的角度看，攻击报文或在内容结构上，或在出现的时序上，或在流量的大小上，攻击报文一定是处于服务程序无法正确处理的畸形状态。限于篇幅，本文暂不讨论比较复杂的与时序及统计相关的攻击，只考虑相对比较简单的单请求攻击，要使攻击得以成功完成，网络攻击的报文的内容和结构必须满足某些必要的条件，这些必要的报文特征用于驱使受攻击进程的处理流程走到触发漏洞的操作，我们用特征集A表示这些必要的报文特征的集合。 攻击者通过编写攻击代码来实现对安全漏洞的利用，特定的攻击代码发出的报文除了必定包含的特征集A内的特征外一般还会包含一些本攻击代码特有的特征，比如特定的shellcode、特定的填充数据等，我们用特征集B表示特定攻击代码的报文特征。 基于规则

的NIDS引擎实现一个基本检测框架，它提供给用户各种匹配选项和操作符作为应用接口，用户可以通过组合选项和操作符来描述关心的网络报文特征，对满足匹配条件的报文进行告警。NIDS厂商的规则支持部门通常会跟踪分析新出现的安全漏洞及攻击代码，提取特征编写出用于检测此攻击的NIDS规则，我们用特征集C表示用规则描述的攻击报文特征。很明显特征集A和B的关系如下图： ----------|||-------||||||==========> 匹配特征增加，满足条件的报文减少 |||||-------|||---------- 理想的攻击检测规则描述的特征集C应该与A完全重合，由于特征集A内的特征描述了所有攻击代码攻击报文所共有的特征，因而与特定的攻击代码无关，这样就避免了漏报。如果对于检测对象的协议类型所做假设正确，由于特征集A内的每个特征是造成攻击所必要的，缺一不可，这些特征中的只要有一个不满足都无法触发漏洞而完成攻击，所以也不存在误报的可能。图示如下： ----------|||=======||""|""|或结构方面的特征|""|=======|||---------- 事实上由于规则描述能力的限制，大多数情况下规则无法完全描述出攻击报文内容和结构上的完成攻击所必需的特征，也就是说，特征集A与特征集C并不是完全重叠的，而是存在各种可能的关系：如果特征集C是如下图中这样作为A的子集，那么规则将可能导致误报，但不会产生漏报： ----------|||-------|||---|||||||---|||-------|||---------- 如果特征集C是如下图中这样包含了特征集A，那么规则将可能导致漏报，但不会产生误报： ----------|||-------|||---|||||||---|||-------|||---------- 如果特征集C与A互不包含，只是如下图存在一

个交集，那么规则将既可能产生误报也有可能导致漏报：
---------- ||| ------- |||||| --- -- ||||||| --- --- ||||||| ------ | ---------- 如果特征集C与A互不包含且没有交集，那么规则本身只是在检测特定的攻击代码发出的报文，而与攻击本身无关，通过简单修改攻击代码可以非常轻易地绕过这类规则的检测，要NIDS发生漏报还是误报完成掌握在攻击者手中： ---------- | ------- |||||||| ------- || ------ ||||| |||| ------ | ---------- 那么如何才能使特征集C与A尽可能地重合？这主要取决于两方面的努力：一、透彻地分析漏洞的成因和利用条件，目的是归纳出准确的特征集A；二、提高NIDS规则的描述能力，使之描述出的特征集C可以尽可能地接近特征集A。NIDS厂商的实力往往体现在这两方面的水准上，如果厂商没有持续而专业的漏洞及攻击的分析能力，在没有分析清楚漏洞细节的情况下胡乱设计NIDS规则必然会带来大量的漏报和误报，NIDS产品使用效果可想而知。下面通过分析一个实际漏洞攻击的NIDS规则设计来使上述看起来有些抽象的描述具体化，突显漏洞分析在设计NIDS规则过程中的极端重要性。