中文摘要

隨著網路行為日亦複雜，以特徵基礎的網路入侵偵測系統中特徵規則的設計已經無法只用單獨一個事件來判斷是否發生攻擊行為。高度辨識度的特徵規則須包含多個事件，同時滿足彼此間特定關係。

例如在TCP型態目的埠為80的封包下，內容中若是出現 content: "GET " 之後依序在特定範圍出現 URL “/../../../../../../../../../../../”，如此便可以判斷出發生攻擊『WEB-MISC iPlanet ../../ DOS attempt』；定義此類的rule稱為 multi-event rule。觀察最具公信力開放程式碼的網路入侵偵測系統-Snort，隨著版本的更新，multi-condition所占的比例也越來越多。

原本純軟體架構的網路入侵偵測系統的運作流程，先由 Pre-processor 將所有需要判斷的條件找出，Detection Engine 再以 link list 的方式循序對所有 rule 的每一個條件進行判斷，在 multi-event rule 成為特徵規則的主流的趨勢下，軟體架構的 Detection Engine 將不利於處理。為了解決這樣的問題，本篇論文嘗試由三個方向對原本的系統進行改良。

第一，因為需要判斷的條件會依據封包屬於不同的網路型態下而有所不同，調整原本的字串比對演算法，在不造成漏判下大幅度減少不必要內容比對。

第二，觀察及分析 Snort rule，對所有 rule 進行分組，獨立的子集合彼此之間可以比對各自的關聯性，以縮小需要的比對範圍。

第三，將原本純軟體架構的 Snort 中 Detection Engine 部分改以 SoC（System on Chip）實現，pre-processor 發現的 condition 以 SoC 上的軟體部份進行分類收集，再以自行設計的硬體電路執行比對的演算法，縮短等待時間。

依照上述三點我們在 FPGA 上設計一個 SoC 系統，驗證我所提出的方法。並經過測試在相同情況之下可以達到軟體執行上八到十四倍的效能。

Abstract

Given the complexity of the Internet connectivity involved, signature-based NIDS (Network Intrusion Detection System) can not rely only on the use of one payload string, regular expression and URL to detect attack intension. A highly recognizable signature must be included in the basis of this particular relationship.

For example, with HTTP flow, when the content “GET ” appears on the fourth byte of the payload and then the URL “/../../../../../../../../../../../” appears, this activates that the detect attack intension “WEB-MISC DOS attempt.” These are called “multi-event rules.” Under the public open-source of NIDS - *Snort*, the number of multi-events in the total number of rules increases depending on the version update.

Because the original Snort program is a pure software system, the “Detection Engine” must search the entire option node (conditions) under the entire sequential RTN list. Under multi-event rules, the general trend is for the ‘Detection Engine’ to become the time-critical component of the Snort system. The objective of this thesis is to attempt to improve the original NIDS using three approaches.

The first objective is to observe target patterns with a specified network application. To do this, we modify an AC multi-pattern match algorithm to filter 60~70% of unnecessary pattern information, which substantially reduces the amount of post-processor overhead.

The second objective is to divide the Snort rule set into several groups. Each event needs only to search the corresponding independent rule group. Therefore, narrowing the search range can reduce the frequency of matching numbers.

Lastly, this thesis proposes a novel architecture to re-plan components of the Snort detection engine. We see that a SoC-based solution is better because it looks

after both programmable software and the logic circuits of specific hardware.

We devise a SoC system, which incorporates the above three objectives, on an Altera FPGA development board (1C20). This SoC-based system is shown to detect all attack intentions correctly. Furthermore, according to latency observations in three networks, the speed of the SoC-based approach is 8 to 14 times faster than a pure software system.