Programmable WIDS API

Tech Con 2025 Abstract 226

HPE Edge;

Programmable WIDS API

Abstract

Wireless Intrusion Detection System (WIDS) on network device consists of 3 parts: 1. monitor incoming wireless packets; 2. analyze the packet content and behavior; 3. alert in logs and events. The most of known intrusions can now be detected by HPE Aruba AP, but new kinds of intrusions become more multifarious with the increasing network devices and proxy service. It requires long time to support new intrusion detections in official build according to current release progress. New IDS features cannot be applied to prior release because new detections have new config knobs. We planned to provide a programmable API to allow users to add new detections flexibly, without expending excessive time. Engineers can freely design their IDS solution and apply it directly.

Problem statement

Various intrusions bring up lots of problems to wireless devices. They can block wireless connections, steal user information or get illegal access to a network. Not only attack the personal device but also endanger an organization's information security. Quick development for a new IDS solution is important.

Once there is a new intrusion published, our engineer needs to develop the monitoring solution, design the alarm mechanism depending on the practical requirement, and develop how to escape and even counterattack also. Normally our product can only support new detection in future major release as it introduces new configuration knobs. Customers have to wait for the solution in a new release, it takes 6 to 12 months.

Our solution

HPE Aruba can now detect the most kinds of attacks, like spoofing, ad-hoc, and so on. Air Monitor (AM) module works in application layer, it has the different monitor and alert policies for each intrusion. There are two kinds of alarms, the first kind is triggered timely by an abnormal packet and the second kind is triggered by periodic check with abnormal counter in stats. For the real-time scenario, AM checks the illegal field of the current packet; And, for periodic detection, AM has the detection period, the threshold value of each stat, and the number of packets.

In the above two scenarios, AP can only detect the predefined intrusions following static rules. AM module works by controlling some configurable parameters and summarizing the stats of traffic. A programmable API will be very helpful to customers to detect new attacks in time.

Since Lua has been basically used on AP to monitor logs and any events (including IDS), we will support analyzing wireless packets in Lua process. So customized IDS check can be implemented. The approach is to embed Lua in AM module to securely collect the data we want, like wireless packets and other running parameters, then Lua process generates a new IDS event or log to inform customers. We will reuse the current protobuf message or update it if needed.

Our programmable IDS API allows users to customize monitor and alert actions in code level, it supports engineer to detect the wireless environment as soon as possible and has a quick development. Here are two parts:

IDS event

It happens when a predefined event is triggered. Engineers can define their new event at any step, to observe how

the detection works, how is the environment going, is the data stream idealized? It works like a log but will be more controllable and more timely. It allows the engineer to observe the abnormal state or stats immediately, scrum the development.

IDS protection

It is to modify the flags in monitor table or trigger any protection. It is to modify the flags in monitor table or trigger any protection. Engineers will define their new rule of new detection, can quickly test it to make sure the solution is workable and stable.

Evidence the solution works

We realized the function by Lua language, a programming language, which supports injecting the script into AM module. Firstly, we place a script to monitor each IDS event from AM module. We save each packet as ‘ampp’ structure, which was defined by HPE Aruba, include the packet information and the stats information passed by driver, into Redis. Then, when the intrusion happened, if it’s a known event but didn’t alert in time, we can use script to require AM trigger the event; if it’s an unknown event, we will use the script to read the ‘ampp’ of each packet, trace the features of an attacker. Finally, we will quickly modify the Lua script and apply it to support detecting this kind of attacks on AP.

Competitive approaches

On AP, it has predefined detections with specific offset in wireless packets, not flexible to support new cases. However, our proposal is a programmable API, which is very helpful to customers to detect new attacks flexibly.

Current status

Currently we can read and catch the event sent to the cloud by script. We are planning to spend 3 months to finish the API with the Lua language and embed interfaces in current C code. It will be ready for a demo when Tech Con starts.

Next steps

It is not easy for users to collect wireless packets when a random attack happens, if there is a programmable tool for monitoring, debugging becomes easy and fast. We will define a standard to make it common in future.