A Brief Technical Overview of the Cyberlytic Platform

Edward Amoroso, *CEO, TAG Cyber*
St. John Harold, *CTO Cyberlytic*

Version 3.0
February 2018

Abstract
This technical note describes the underlying design and data analytics of the Cyberlytic platform, including how web traffic is analyzed for evidence of cyber security threats, processed using artificial intelligence-based methods, and then assigned a risk score based on key characteristics of the observed attack. The primary focus here is on the Cyberlytic Profiler, a commercially available cyber security solution.

Preface

This note is based on continuing technical discussions and interactions between the authors, and several managing principals from Cyberlytic during the 4Q17 to 1Q18 timeframe. The note incorporates new learnings and insights from the authors based on discussions, review of technical materials, and several briefings from the Cyberlytic technical team.[1] It is written to help anyone interested in the Cyberlytic platform to better understand its foundational design and operation in monitoring real-time cyber risk within web application traffic. It is also written to provide general introduction to the use of artificial intelligence and machine learning methods for web application security.

Introduction

The commercially-available Cyberlytic platform is designed to address web application security in a way that improves on the weaknesses of traditional web application firewalls (WAFs). As most security experts would agree, WAFs rely on static signature and regular expression (regex) matching, which can be easily bypassed [1]. Such approach requires intensive and constant manual intervention, which can result in limited or non-protection if rules are not managed properly. This WAF approach is also generally considered to be less effective against zero-day attacks, and prone to generation of false positive and false negative alerts [2, 3].

The primary component of the Cyberlytic platform is called the *Profiler*, which is a commercially-available, cloud hosted platform that operates passively to detect and prioritize web injection attacks [4]. The tool supports detection of web injection attacks, and prioritization of these attacks based on the associated cyber risk[2]. It improves incident response processes through the generation of risk alerts and management reports. It also uses artificial intelligence (AI), including machine-learning (ML) algorithms, to dynamically learn and identify threats such as cross-site scripting (XSS) and Standard Query Language injection (SQLi) attacks, as well as zero-day web injection exploits.

A new component of the Cyberlytic platform is called the *Defender*. Currently in beta testing by the Cyberlytic team, the Defender is being designed as a cloud-based security component that will operate in-line to actively prevent malicious content from reaching the web application. It will do so using the detection models learned by the Profiler. The Defender capability is described only at a high-level in this paper; as it becomes generally available for use, interested readers can find more information at the Cyberlytic website.

[1] Edward Amoroso is new to the Cyberlytic platform with this initiative, whereas St. John Harold has been deeply involved in the design from its inception.
[2] Cyberlytic's definition of cyber risk is a function of threat, vulnerability, and business impact.

This paper is written specifically to provide a technical and analytical overview of how Cyberlytic's unique Profiler approach is designed, and how it utilizes a threat classification approach for web security based on AI techniques [5]. The threat detection and classification approach combines deep packet inspection with AI algorithms that apply continuous learning to measure malicious features of a web application attack. In addition to providing good information about the commercial tool, this paper also hopefully serves as a useful case study into how AI and ML methods can be used to reduce the risk of web application attacks.

Profiler Overview

The primary high-level goal of the Profiler is to identify malicious web traffic, and to then determine a risk score, based on relevant characteristics. The risk score is intended to offer a means for subsequent security mitigation either by complementary security tools, or eventually in conjunction with Cyberlytic's emerging Defender component (described briefly later). The output from the analytics engine is displayed via the Profiler portal, which offers security analysts a near-real-time view of risk exposure, including threat analysis, targeted hosts, attack methods observed, and risk measured across a timeline.

The solution is deployed as a centrally-hosted cloud infrastructure platform, with collector agents installed within the organization's web application or network. It collects all web application traffic, analyzing both the browser request and server response to identify malicious activity, and then determines the associated cyber risk score. The Profiler requires access to unencrypted HTTP traffic for optimal analytical performance. Therefore, the HTTPS traffic must be offloaded for the Cyberlytic solution to analyze all the potential injection attack paths within a web application session. A transport layer security (TLS) offload is achieved using the client web server's TLS decryption or via a TLS offload device.

Cyberlytic has developed several connectors that can be installed within different parts of the stack, dependent on where the TLS offload occurs. The connector captures both the browser request and server response, which are collectively defined as an event. It then sends a copy of the full HTTP session to the Profiler for analysis. Each connector maintains its own secured TLS connection to the centralized Profiler. The results of the analytics are then processed and stored within the Profiler portal, and if an attack exceeds a risk threshold, then an email alert is generated. The following are the three core connector methods that are available to collect the HTTP traffic for analysis:

- *Web Agent Connector* – This agent is installed into the web server framework, and acts as middleware to capture the HTTP sessions, and then forward a copy of this traffic via TLS secured path to the Profiler (see Figure 1).
- *Network Connector* – This component monitors a mirrored port of a switch. It captures all the network traffic, extracting the HTTP session stream from it, and then forwarding a copy via TLS secured path to the Profiler (see Figure 2). As stated above, for the Profiler to work correctly, all HTTPS traffic must be offloaded. Therefore, the traffic mirroring should occur on a switch that is either connected to, or after, the outbound traffic of the TLS offload device.[3]
- *Proxy Connector* – This device is a lightweight NGINX component that acts as a man-in-the-middle HTTP proxy intercepting web traffic. The proxy can provide TLS offload capability. A copy of the intercepted HTTP traffic is forwarded via a TLS secured path to the Profiler (see Figure 3).

[3] The network connector is deployed as a virtual machine image and is capable of handling IETF standards SPAN/RSPAN/ERSPAN.

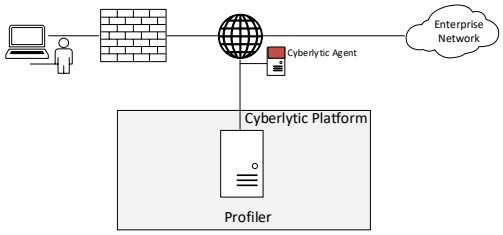

Figure 1. Web Agent Connector Method

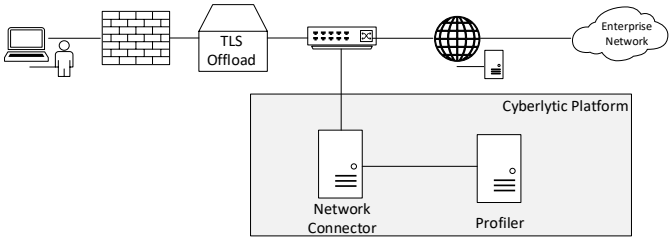

Figure 2. Network Connector Method

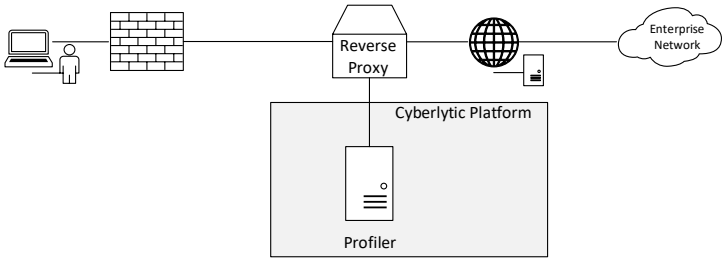

Figure 3. Proxy Connector Method

As described above, the role of the connector is to capture in near real-time a copy of the full HTTP traffic stream for the Profiler to analysis. There are three key decision points in this analysis process: *Anomaly check*, *malicious check*, and *risk analysis*. These decision points and associated data flow are represented in Figure 4.

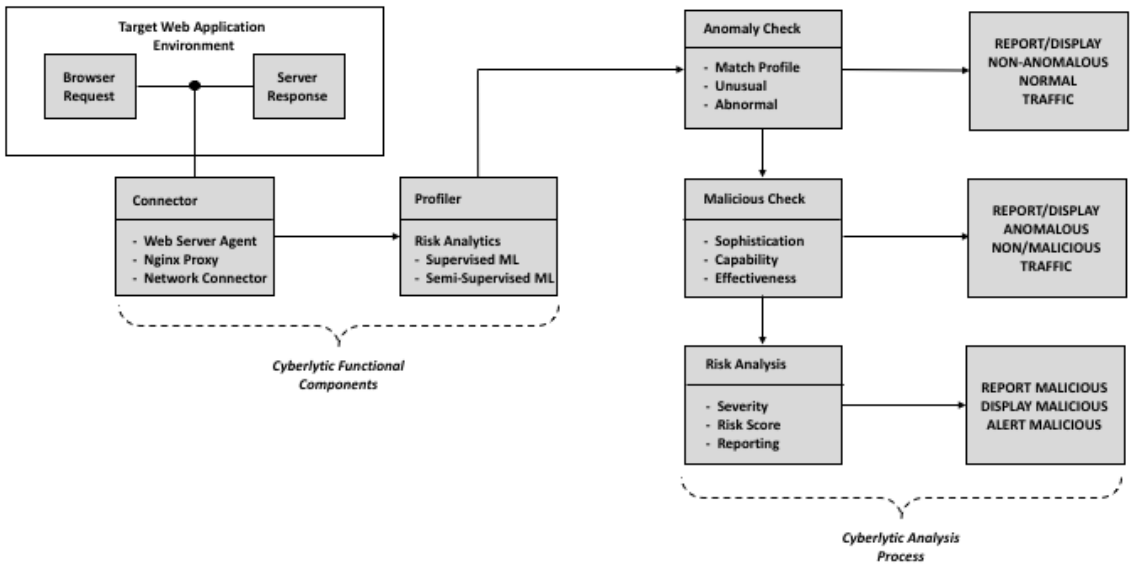

Figure 4. Profiler Decision Flow

Profiler Analysis Points

The first analysis point determines if the HTTP activity is normal or anomalous. Each HTTP stream is parsed into a structured event consisting of an individual HTTP browser request and server response. The analysis is carried out across the full HTTP stack including HTTP methods, headers, URL parameters, and request body, if present. This analysis point uses unsupervised machine learning model to detect anomalous web activity. This model characterizes the data flows to the web application based on profiles of normal behavior. Web applications are generally unique, and therefore the analysis is tailored to each website's user interactions and traffic patterns. The anomalous component requires initial training to understand specifics of the web application, and this period is subject to factors such as diversity and volume of data.

For unsupervised analysis, the Profiler extracts feature sets from the session data, such as content length of the request, character distribution of the query arguments, fingerprinting techniques to abstract the session, and string distance analysis. These features are captured and used to build the training data for the unsupervised model. After the training period has collected enough data to profile characteristics of the web application, any incoming data is compared against known normal traffic. Due to the nature of this data collection, the models are dynamically updating and learning as new events are processed. Ultimately, the request is classified as normal or sent for further analysis which is outlined next.

The second analysis point determines if the anomalous event shows malicious traits. If the event is determined to be malicious, then a set of labels is generated covering the following attack characteristics.

- *Attack Sophistication* – This characteristic identifies the relative complexity of the web injection construct.
- *Attack Capability* – This characteristic identifies the attacker threat profile based on techniques used.
- *Attack Effectiveness* – This characteristic identifies the server response dependent on the success of an attack.

This analysis point uses supervised machine learning model to categorize the most common web attack types [6], such as SQLi, XSS and Shell-injection attacks, generating a label for each of the key characteristics analyzed.

The process methodically checks for the presence of attack strings based on a series of models for SQLi, XSS, and Shell-injections. A so-called *support vector machine* approach determines if there are characteristics that are like the predefined training models and then categorizes the event with sophistication, capability and effectiveness labels. The sophistication and capability labels range from 'Very Low' to 'Very High' and the effectiveness label is graded as either 'Effective' or 'Not Effective'. Should any outlier events be detected, they are then tagged for manual review to update and improve the classification model and training data. Cyberlytic is continuously developing models for new attack vectors. The labels are used for the generation of a risk score in the next process.

The third analysis point predicts the level of cyber risk to the website using the category labels. This generates a human readable representation of cyber risk. Highly sophisticated attacks from capable humans with a successful malicious attack are therefore the highest risk. If there is no evidence of any malicious characteristics, then the event is determined to be anomalous for the security analyst to review in slower time.

For malicious events, the Profiler generates a risk score associated with the classified malicious attack. Cyberlytic defines cyber risk as a function of threat, vulnerability, and business impact. In the context of the Profiler, the risk score is calculated by the labels generated in the second decision point.

To calculate the cyber risk, the Profiler assesses the degrees of truth, where sets of labels have degrees of imprecision about the label boundaries and their relationship is defined as a matter of degree of truthfulness or falseness. The resulting risk score denotes a normalized and comparable analysis of the probability of a malicious attack that is egressing sensitive data.

Upon completion of the analysis, if a malicious event is identified, this is displayed in the Profiler's attack dashboard in near-real-time. Historical events can be viewed retrospectively within the reporting feature. Where the event has no malicious characteristics, the event is displayed within the anomaly dashboard. Other key features of the Profiler include automated email alerts, syslog integration with security information event monitoring (SIEM) tools and other third-party correlation systems.

Defender Future

Cyberlytic is currently developing a Defender component that will actively stop web application threats. The architectural objective for the Defender is to involve a hybrid operation where the component can reside in the web traffic flow, as one might find with a WAF, but also with direct API connectivity to existing mitigation tools from complementary cyber security vendors. Some practical considerations include the following:

- *Active Versus Passive Operation* – When operating offline in passive mode, the Cyberlytic platform can be easily integrated into an enterprise architecture without due consideration for service interruption, which is an advantage found in any passive cyber security tool. Once operating in active mode, however, the Defender must incorporate enterprise-grade (or even carrier-grade) functional assurance of continued fail-safe operation.
- *Integration with Third-Party Components* – Establishing connectivity with external vendor WAFs and other security tools is a natural design objective for the Cyberlytic platform. Determination of which tools require such integration (e.g., Cisco, F5, Palo Alto Networks) is an on-going task driven by market interactions with Cyberlytic customers.
- *Human Versus Machine Attack Origination Detection* – Considerable attention is required in any web application security tool to differentiate between human and machine-originated attacks. The Cyberlytic platform comfortably deals with both cases, but any design decision that involves actions such as source IP shuns would require different handling if the origin is determined to be a botnet.

As of the date of this report writing (1Q2018), the Cyberlytic team is currently in beta testing the Defender and aims to have an active mitigation platform available for its customers in 3Q2018. Readers interested in Defender design and operations, should watch for information on the Cyberlytic website.

References

[1] P Lupták, "Bypassing Web Application Firewalls," *Security and Protection of Information,* 2011 (SPI), pp 79-88.
[2] L Dali*, "A Survey of Intrusion Detection System"* IEEE.
[3] S George, "An Imperative Analysis of diverse State of Art Solutions for Internet and Web Application Security," *International conference on Computer Science and Information Systems (ICSIS'2014)* Oct 17-18, 2014 Dubai (UAE), pp 21-27.
[4] *The Profiler – AI for Web Security – Technical Data Sheet, https://www.cyberlytic.com/*

[5] S. Laidlaw, S. Harold and M. Hillick, *"Profiling cyber threats detected in a target environment and automatically generating one or more rule bases for an expert system usable to profile cyber threats detected in a target environment,"* Patents: US9503472B2 & GB201321565D0 (EU Pending)
[6] *"State of the Internet – Security,"* Volume 4, Number 3, Q3 2017, Report: https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/q3-2017-state-of-the-internet-security-report.pdf, page 15.

Additional Bibliography

Cyberlytic Intelligence Web Security – Fact Sheet, https://www.cyberlytic.com/
Cyberlytic Profiler Analytics – https://www.cyberlytic.com/
Private Notes from Cyberlytic Development Team, 4Q2017