QUTTERA

Quttera investigation engine

Added value and user benefits

Contents

This document contains insight into technology utilized by Quttera investigation engine, types of threats that could be detected by this engine and user benefits from this technology.

1. The problem

Hackers install malware on popular web sites by exploiting security weaknesses on their servers and thus gaining full access to the compromised web site. In most cases the malicious code is not visible or easily detected, and it infects computers of web site visitors when they simply browse this web site.

This is one of the main approaches used by hackers to spread viruses, hijack Internet devices or steal sensitive data such as credit card numbers or other personal information. As such, hackers are planting a malicious code on legitimate websites in order to distribute malware among the web site visitors and infect as much victims as possible. These attacks can take several forms, including “drive-by-downloads” and “dangerous downloads”.

In a “drive-by-download” attack, a malware is downloaded to user’s computer, simply by loading an infected web page in a browser; no interaction on the user side other than loading the web page is required to accomplish the attack.

In a “dangerous download” attack, hackers plant malicious files such as executable, documents, images, that contain malicious code on a legitimate, victim web site, and users get infected when they click on links to the malicious files.

Once a malware infects certain computer, hackers then can take advantage of those compromised devices in a various ways, including: logging users’ keystrokes, using the compromised computer to send spam, converting it to become a part of a bot, distribute more malware or simply modify search results provided by search engines like Google, Bing and Yahoo.

One of major roles in such kind of attacks is the JavaScript language which is an integral part of modern web and PDF documents. JavaScript is a high level language which in addition to its direct functionality is also used to obfuscate malicious code used to generate malicious input and exploit 0-day security vulnerabilities found in Internet client applications like web-browsers and PDF readers.

In general, modern malicious content can be divided into two groups. The first one is JavaScript code that is used to generate malicious inputs like binary exploits or shell-codes; and the second group is these binary exploits which are finally being injected into attacked process and provide full remote control over the attacked device. Due to simplicity of JavaScript language and in order to overcome signature and pattern-based detection mechanisms, malware writers encode both kinds of content using widely used generators and thus making injected malicious code undetectable by signature-based and pattern-based detection engines.

2. Quttera investigation technology

Quttera investigation technology utilizes non-signature investigation approaches which are based on content emulation and penetration testing. This technology is capable to recognize encoded JavaScript code and binary shell-code inside legitimate media files and digital documents.

3. Quttera infrastructure technology

In order to improve existing identification capabilities we have developed a heuristic non-signature based detection infrastructure which is capable to detect and protect from various kinds of web-threats. Quttera malicious content detection engine comprises of multiple non-signatures based investigation and analysis methods. Quttera engine identifies JavaScript based attacks and security vulnerability exploits. On top of that, Quttera engine detects encoded shell-codes, JavaScript obfuscation techniques and JavaScript packers which are used to hide malicious content and dangerous code from signature and pattern based identification mechanisms.

Quttera investigation infrastructure embeds several execution emulators which are not only emulating execution of the targeted device but also penetrate the investigated content and detect web-treats regardless of the kind of the targeted web browser or operating system or Internet device.

Quttera investigation engine includes three main modules:

- X86 emulator – emulation and detection of shell-codes and sensible malicious sequences of executable instructions
- JavaScript emulator – emulation and detection of malicious JavaScript scripts and HTML pages and
- PDF reader emulator – detection of malicious PDF files.

Based on this architecture, Quttera investigation engine is capable to recognize and detect:

- Security vulnerability exploits referencing system internals (x86 architecture)
- Security vulnerability exploits referencing process internals(x86 architecture)
- Sensible sequences of CPU instructions inside text and binary files(x86 architecture)
- Hidden Java-script code which is being generated during emulation of the original script or web page
- Suspicious Java-script containing code obfuscation or injection of hidden Java-script
- Hidden HTML elements generated during emulation of the original script or web page
- PDF files containing embedded malicious PE files, hidden suspicious actions, hidden suspicious elements and Java-script code obfuscation
- Malformed PDF files
- Encrypted PDF files

Quttera infrastructure is designed and implemented as a generic and modular investigation engine and can be adopted and integrated into various information security software like:

- Intrusion detection/prevention systems (IDS/IPS)
- Antiviruses and malware detection tools
- Malicious and suspicious web sites detection systems
- Web sites investigation systems
- Security Internet suits
- Application gateways
- Mail servers

4. User benefits

Based on heuristic static and dynamic investigation analysis Quttera engine capable to detect and recognize malicious files containing suspicious JavaScript code and completely new binary shell-codes regardless the attacked operating system, attacked device and attacked Internet client application.

Quttera detects the following types of threats:

- Security vulnerability exploits referencing system internals(x86 architecture)
- Security vulnerability exploits referencing process internals(x86 architecture)
- Sensible sequences of CPU instructions inside text and binary files(x86 architecture)
- Hidden Java-script code generated during emulation of the original script or web page

- Suspicious Java-script containing code obfuscation or injection of hidden Java-script
- Hidden HTML elements generated during emulation of the original script or web page
- PDF files containing embedded malicious PE files
- PDF files containing hidden suspicious actions
- PDF files containing hidden suspicious elements
- PDF files containing Java-script code obfuscation
- Malformed PDF files
- Encrypted PDF files
- Unconditional re-directions (new feature)

5. Implementation and industrial features

Main features

1. A core code which is a basis of the technology.
2. The core has a form of a generic and independent engine.
3. A self-learning mechanism that improves the detection ratio.
4. Engine has a modular structure. Each module is an independent unit.
5. A built-in feasibility to be adopted in almost any other solution/ system.
6. A unique approach to the dynamic investigation of the data.

Problems that exists in the computer security and can be solved with Quttera

1. It solves the problem of the need of the additional data (signature, attacked process info, attacked OS info and etc...). Quttera technology doesn't need it.
2. Investigation is automatic and can significantly reduce the load on the threats investigation team.
3. Detects encoded JS/HTML/PDF threats.
4. No need in constant updates of the signature database.
5. Detects JS obfuscation techniques
6. Detects encrypted binary shell-codes

Recent use of the technology

1. It is currently used in cloud-based online url scanning system. 'WIS'.(http://www.quttera.com/)
2. It is currently used in the PC based version of url scanning. 'CLI URL scanner'. (http://www.quttera.com/qurlscanner)

Quttera technology can be used in/as/with

- As an integrated module in any other security suite.
- As a separate tool to investigate the data.
- Intrusion detection/prevention systems (IDS/IPS)
- Antiviruses and malware detection tools
- Malicious and suspicious web sites detection systems

- Web sites investigation systems
- Security Internet suits
- Application gateways
- Mail servers

Quttera technology can improve/ add value

- It can accelerate the process of the data investigation.
- It can improve the false-positive ratio.
- It can address the zero-day exploits problem.
- It can recognize suspicious/malicious URLs