Menlo Security

Menlo Security Isolation Platform

Efficacy Report

July 2016

Threat Prevention Overview

Conventional threat prevention products attempt to distinguish between 'good' and 'bad' content, and then implement policies intended to allow the good content and block the bad

Malware plays a significant role in many of today's high-profile cyber attacks and data breaches including Forbes, Target and Home Depot. More than 400 million variants of malware exist, a number that has grown consistently year-over-year. Web and email remain the two leading vectors for malware, and the threat is now so prevalent that IT organizations and individuals fear clicking on any web links.

Conventional threat prevention products (IPS, SWG, sandboxing, firewalls, etc.) attempt to distinguish between "good" and "bad" content, and then implement policies intended to allow the good content and block the bad. This approach is ineffective because it is difficult to keep up with the millions of strains of evolving malware. And even trusted websites and documents can inadvertently deliver malware.

The Menlo Security Isolation Platform (MSIP) eliminates the possibility of malware reaching user devices via compromised or malicious web sites, email or documents. This is not detection or classification, rather the user's Web session and all active content (e.g. Java, Flash, etc.), whether good or bad, is fully executed and contained in the MSIP. Only safe, malware-free rendering information is delivered to the user's endpoint. No active content—including any potential malware—leaves the platform.

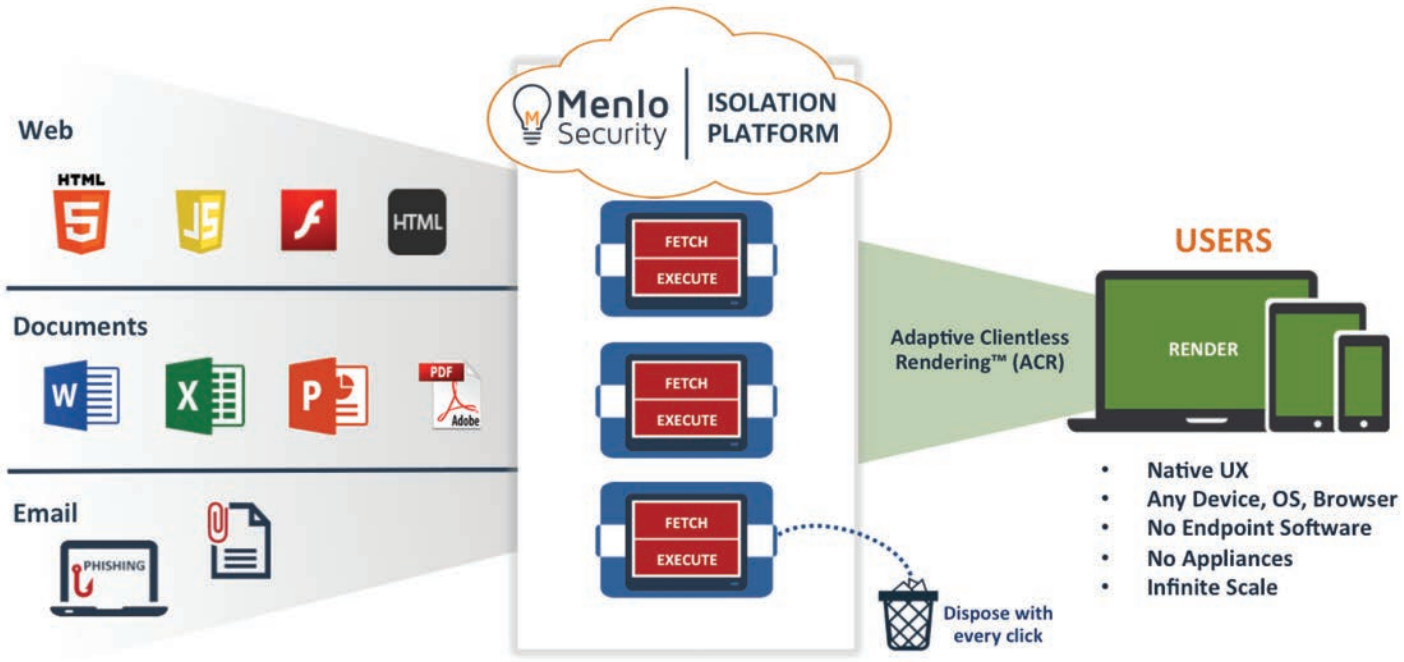


Figure 1. The MSIP eliminates malware and its effects across the most critical threat vectors

MSIP ensures that only safe rendering data reaches the client browser, all risky active content is executed in isolation

Adaptive Clientless Rendering (ACR)

The MSIP utilizes Menlo Security's patented Adaptive Clientless Rendering™ (ACR) technology which provides the secure connection from the user's session running in the MSIP to the user's native browser. ACR technology requires no endpoint software or plug-ins and delivers a completely native user experience essentially indistinguishable from direct interaction with a web site.

ACR leverages the fact that all modern browsers use a common framework for describing the elements on a web page, including text, graphics, video, etc. When web content executes normally in a user's browser it generates document object model (DOM) elements and an associated rendering tree that tells the browser how to create the user's display. When a web session is executed in the MSIP it also generates DOM and rendering tree information which is then optimized and delivered via the ACR to the user's browser. The user's browser takes the information delivered via the ACR and generates the user's view as if the content were executing in the local browser.

A trusted JavaScript delivered to the user's browser from the MSIP at the beginning of each session establishes and maintains the ACR channel using SSL. For each type of web content the ACR engine selects the optimal encoding and transport mechanism for delivery to the user's browser. For example, dangerous content such as Flash is executed in the MSIP and then delivered as a hi-fidelity, interactive experience in the user's browser. In all cases, the user's browser receives non-executable, malware-free content that renders naturally and preserves the user's native experience. The ACR protocol carries user activity (keystrokes and mouse clicks) to the MSIP and prevents malicious activity from flowing in the upstream direction.

Study Purpose & Methodology

The purpose of the study is to showcase how MSIP can eliminate 100% of web-borne malware.

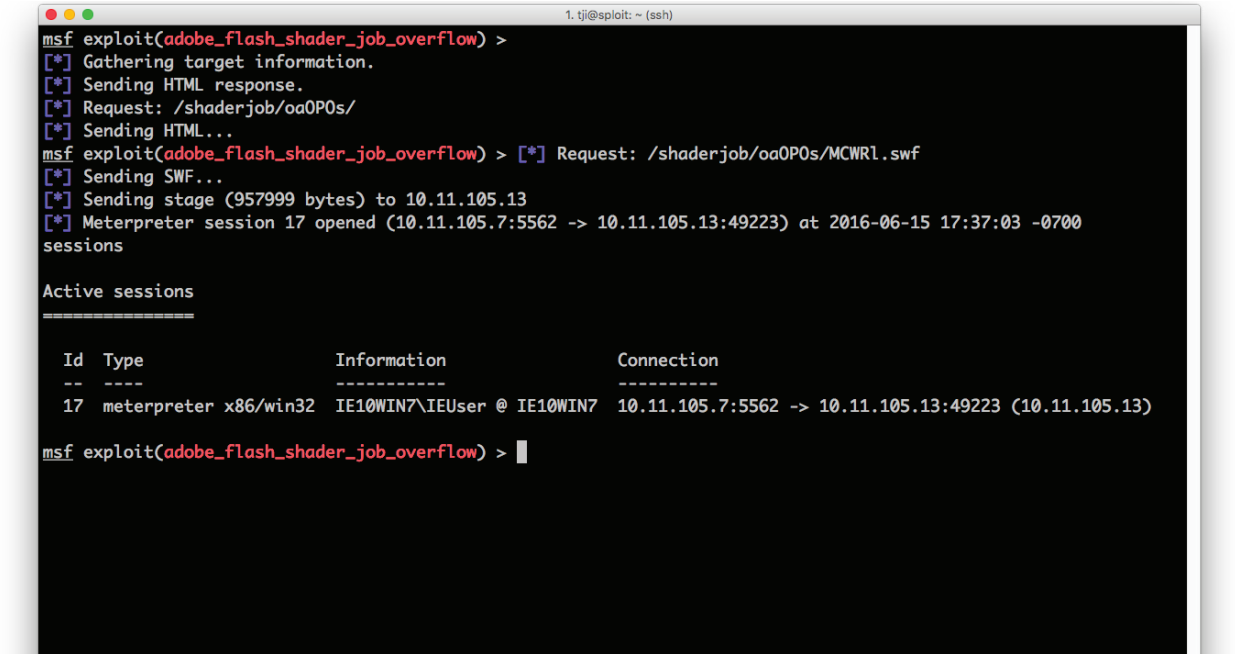
The study was conducted using Metasploit, a leading penetration testing platform, to serve a variety of malicious websites and documents to a device under direct and isolated conditions. It was then determined whether or not the device was exploited by each attack.

Results

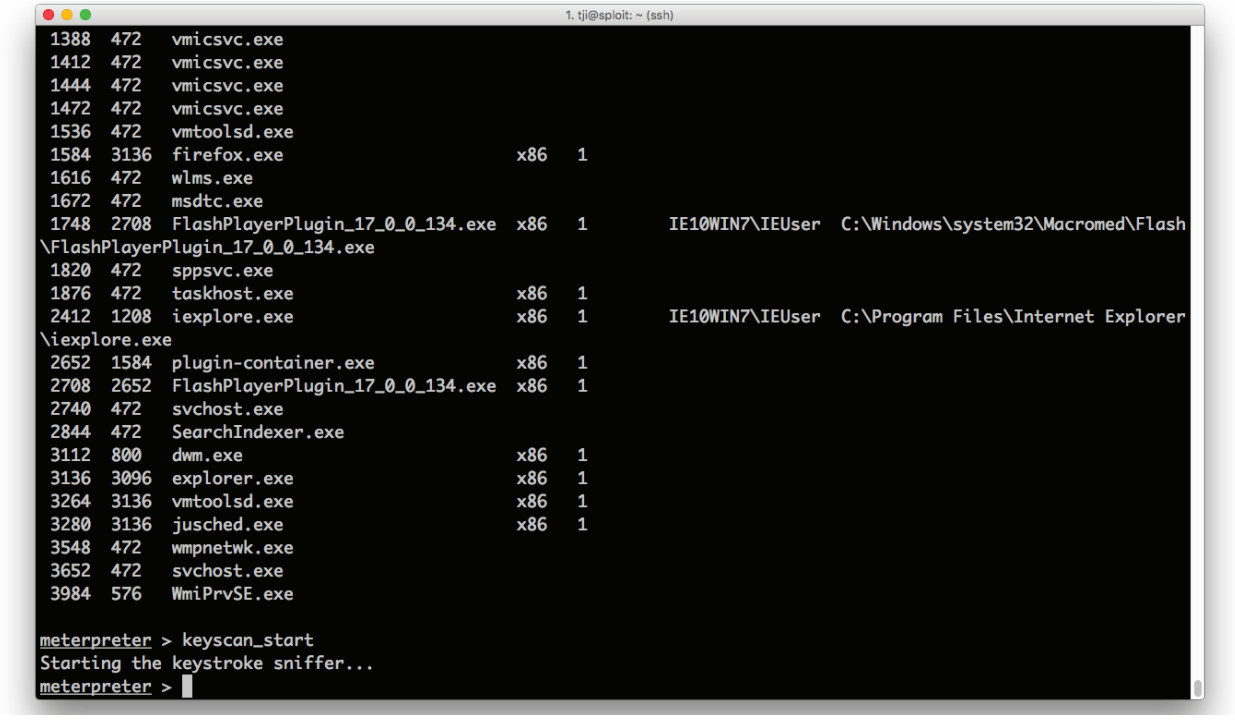
Metasploit Exploit Name	Description / CVE	Direct Web Connection	Web Connection via MSIP
adobe_flash_opaque_background_uaf	CVE-2015-5122 Flash DisplayObject use after free vulnerability on opaqueBackground property Flash version: 18.0.0.203 or earlier	Exploited	Protected
adobe_flash_hacking_team_uaf	CVE-2015-5119 ByteArray class user-after-free (UAF) vulnerability Flash version: 18.0.0.194 or earlier	Exploited	Protected
adobe_flash_shader_drawing_fill	CVE-2015-3105 Shader fill memory corruption Flash version: 17.0.0.188 or earlier	Exploited	Protected
adobe_flash_nellymoser_bof	CVE-2015-3043 Buffer overflow on 'nellymoser' audio in an FLV video objects Flash versions: 18.0.0.161 OR 17.0.0.169 or earlier	Exploited	Protected
adobe_flash_shader_job_overflow	CVE-2015-3090 Shaderjob buffer overflow Flash version: 17.0.0.169 or earlier	Exploited	Protected
adobe_flash_domain_memory_uaf	CVE-2015-0359 ByteArray use after free vulnerability Flash version: 17.0.0.134 or earlier	Exploited	Protected
adobe_flash_net_connection_confusion	CVE-2015-0336 Type confusion vulnerability in NetConnection class Flash version: 16.0.0.305 or earlier	Exploited	Protected
adobe_flash_uncompress_zlib_uaf	CVE-2015-0311 Use after free vulnerability in ByteArray zlib uncompress Flash version: 16.0.0.287 or earlier	Exploited	Protected
adobe_flash_worker_byte_array_uaf	CVE-2015-0313 ByteArray use after free Flash version: 16.0.0.296 or earlier	Exploited	Protected
java_rhino	CVE-2011-3544 Java Rhino script engine vulnerability Java version: 1.6u27, 1.7 or earlier	Exploited	Protected
java_jre17_reflection_types	CVE-2013-2423 Java reflection type confusion vulnerability Java version: Java 1.7u17 or earlier	Exploited	Protected
java_jre17_jmxbean	CVE-2013-0422 JMX Bean Server class abuse vulnerability Java version: Java 1.7u10 or earlier	Exploited	Protected
java_jre17_provider_skeleton	CVE-2013-2460 Provider skeleton insecure invoke method Java version: 1.7u21 or earlier	Exploited	Protected
firefox_proto_crmfrequest	CVE-2012-3993 ExposedProps property code execution vulnerability Firefox version 5-15	Exploited	Protected
firefox_tostring_console_injection	CVE-2013-1710 CMRF request format vulnerability Firefox version 15-22	Exploited	Protected

Sample of Exploit in Progress

Flash exploit succeeded, "Active sessions" shows a background connection from the attack host to the exploited desktop which allows full system control.



Metasploit "attacker" has complete control of exploited machine. Attacker can perform many functions, including remote code execution, keystroke logging, screenshots, data exfiltration and jumping to other machines.



Conclusions

The results of this test demonstrate that the Menlo Security Isolation Platform is 100% effective in preventing a wide variety of web-based exploits from infecting the test endpoint.

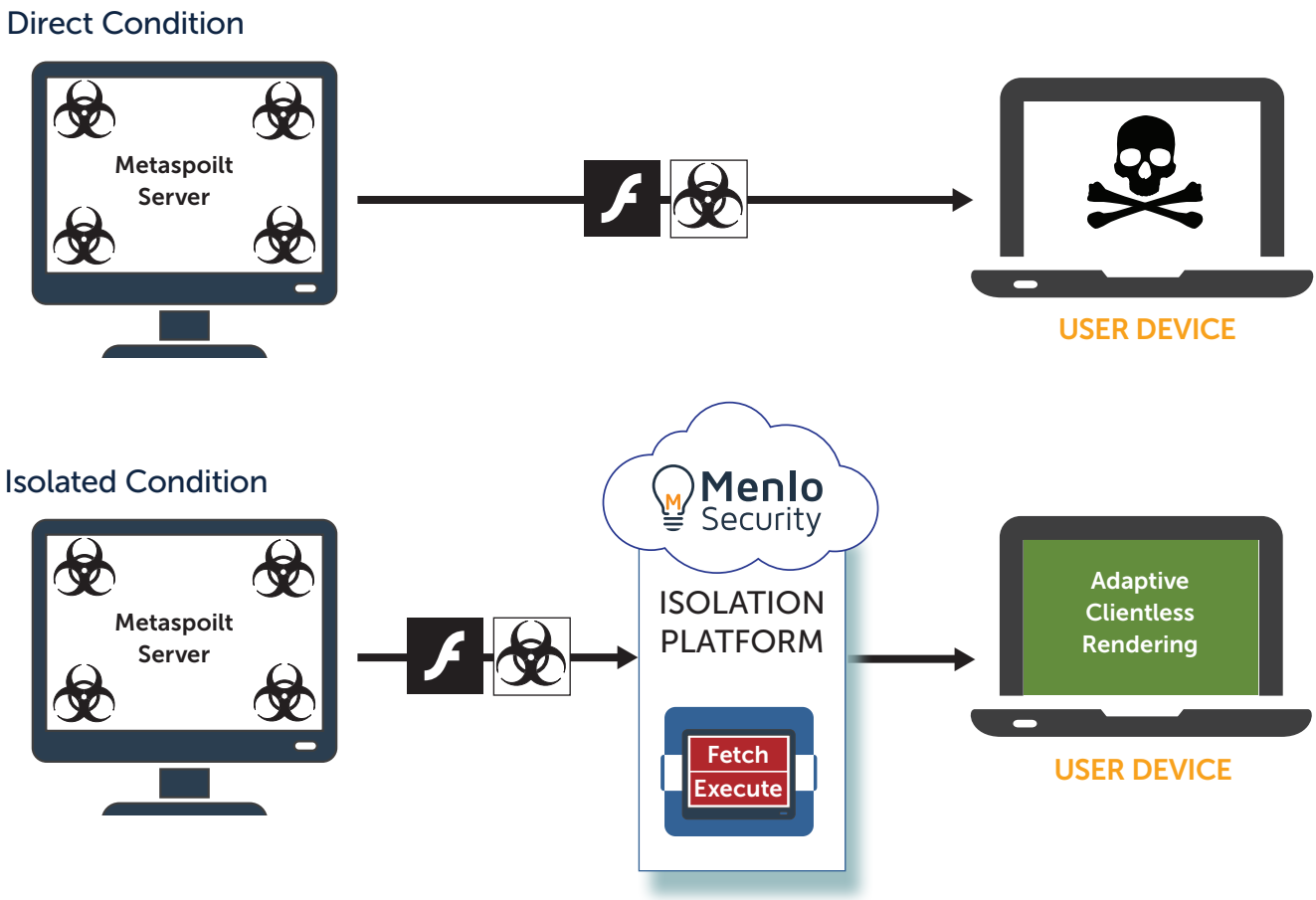


Figure 2. Test architecture and results for direct and isolated conditions

For more information, contact us at info@menlosecurity.com

Menlo Security

934 Santa Cruz Avenue
Menlo Park, CA 94025
Tel: 650 614 1795

menlosecurity.com