Combating Malvertising and Drive-By Downloads

How to Ensure Better Protection Against Growing Threats with the Juniper ATP Appliance

Table of Contents

©2018, Juniper Networks, Inc.

Executive Summary

New strains of malware are constantly threatening businesses, creating angst for executives and security operation teams (SOCs). As cyber risks grow in volume and sophistication, the tools used to find and eradicate them have to get smarter and more scalable. This paper discusses the dynamic nature of advanced threats and the need for an advanced threat defense solution that offers a modern approach to cybersecurity, helping enterprises protect their intellectual property and the privacy of their employees.

Introduction

Thanks to its convenience, online shopping has become increasingly popular over the last few years. Capitalizing on this trend, marketing experts are exploiting this new avenue to advertise their products; it's become virtually impossible to browse even a few pages without encountering an advertisement. Advertisers, however, aren't the only ones to recognize the potential of this new medium. Malware experts are leveraging it to launch a new type of attack: malvertising.

Malvertising involves injecting malicious code into legitimate advertisements. These attacks started as amateur "Click on Me!" buttons that fooled a lot of people into committing dangerous or foolish acts—for instance, installing a fake antivirus program that is actually a malware in disguise.

However, over time and after a lot of education, people have become more mindful about what they click. As a result, attackers have had to become more sophisticated, paving the way for a new type of threat called "drive-by downloads" whereby attackers automatically install malware on endpoints without users taking any action at all.

Anatomy of a Drive-By Download

A drive-by download is a multistage attack:

1. The attacker embeds malicious code into an online advertisement displayed on a trusted website.
2. A user visiting the website gets redirected to the attacker's site without the user clicking on the advertisement.
3. An exploit kit from the attacker's site looks for possible vulnerabilities on the user's endpoint.
4. Based on the exploit discovered, a desired malware is downloaded to the endpoint without the user's knowledge, and without requiring any action on the user's part.

Normally benign websites are compromised any number of ways—for instance, by embedding malicious code in a comment field or a poorly secured Web form. The easiest way to infect a site, however, is by taking advantage of a flaw in an online advertisement and injecting malicious code into it. This way, trusted websites visited by thousands every day unknowingly host advertisements running malicious code.

The malicious code redirects users to the attacker's website by loading the offending URL in a new window. This new window goes undetected because attackers use a common HTML feature called Inline Frame, or iFrame for short—an HTML document embedded within another HTML document. For example, a YouTube video can be seamlessly embedded into a webpage. In actuality, it's just a regular webpage playing a YouTube video inserted into the main page by adjusting the size and removing the borders, creating the illusion that the video is part of the overall page. When the malicious code redirects the user to a different website, it opens up in a tiny window that is not easily spotted.

Once the user is redirected to the attacker's webpage, an exploit kit examines the endpoint for possible vulnerabilities or openings. This is the beginning of the attack; the exploit kit gathers information about the operating system, browser type, browser version, and browser plugins, looking for security gaps. Plugins such as Java Runtime Environment, Adobe Flash Player, and Adobe Reader are popular targets. The exploit itself doesn't cause any actual damage—it has merely cracked the security code of the targeted site. Nothing has been stolen yet.

Armed with knowledge about how to attack the victim, the exploit kit proceeds to download appropriate malware to the victim's endpoint. The malware, also known as "payload," is installed on the endpoint without the user's knowledge. The download goes unnoticed because it is usually obfuscated—a common technique used by attackers to evade traditional signature-based detection engines and mask the real purpose of the malicious code. Once the malware has been downloaded and executed, it does what it's designed to do: make money for the attacker by extracting crucial banking information, for example, or locking folders until money is paid to release them (commonly known as ransomware).

More insidious attacks may start with reconnaissance tools that stay "low and slow," taking stock of critical assets on the network and searching for access credentials.

©2018, Juniper Networks, Inc.

Drive-By Downloads on the Rise

Drive-by downloads have become a serious threat. There are several reasons why.

One of the most compelling reasons is the fact that any person with malicious intent, even if they have zero malware writing skills, can stage a drive-by download attack on several endpoints across the globe. Exploit kits and payloads are sold on the darknet or in underground markets, making them easy to get. Since the darknet is anonymous, it is hard to trace these purchases.

Sophisticated hackers have also developed exploit kits that are easy to use. These kits provide a graphical user interface that helps the attacker decide who the next victims will be and show the progression of infections on the targeted machines. Some kits even offer a fancy dashboard that shows statistics on the number of machines infected that day. The attacker can sort this data any number of ways: by OS, browser, or country. They can even generate pie charts and graphs to organize attack details.

Some exploit kits have taken it to the next level by including multiple-user support and an authorization system that allows groups of users to manage their data.

The recent Angler exploit provides some insight into how mature these kits have become. The developers of the Angler exploit kit were always one step ahead of the game; updates designed to exploit new vulnerabilities were produced faster than the security updates created to patch the targeted software. The Angler exploit kit could even detect whether antivirus software was installed on the endpoint itself or was being run in a sandbox.

Another reason for the increase in drive-by download attacks is the means by which hackers spread exploits. Planting drive-by downloads on trusted websites using vulnerabilities in online advertisements have allowed exploits to proliferate exponentially. Juniper Threat Labs investigated the Angler exploit kit and discovered several infected domains spread across the United States, Italy, Germany, Japan, India, and other countries. At least 10 million people visited those compromised websites within 10 days. One of the most popular of these infected domains was The Huffington Post.

One reason drive-by downloads are so successful is a general lack of awareness about the need to keep software applications up to date. People use Netflix, Hulu, and YouTube every day, but only a handful know that these sites internally use other applications such as Flash Player, Microsoft Silverlight, and Java, all of which need frequent security patches. Even for the well aware, with new vulnerabilities being discovered every other day, it becomes tedious for users to regularly update the software by closing all applications that use the software, waiting for the update to complete, and then restarting all the applications.

Thanks to the availability of sophisticated yet easy-to-use tools, a convenient medium for spreading attacks, and a healthy supply of unsuspecting victims, it's clear why drive-by downloads stand out among other types of attacks.

Shortcomings of Existing Security Solutions

The earliest known occurrence of a drive-by download was in 2006, but it has only started getting attention in recent years. As a result, existing security solutions are simply not equipped to cope with these attacks. Here's why:

- **Antivirus Products:** Antivirus products rely on detecting known attack signatures. When it comes to zero-day exploits, antivirus engines are always a few days behind, giving the exploit time to spread. Even when signatures have been updated, they are not always effective, since some attacks—such as the Angler and Nuclear exploits—can detect the presence of an antivirus product on an endpoint and avoid that target. Some forms of malware also use various obfuscation techniques to hide from antivirus engines.
- **Standalone Sandbox Solutions:** Sandbox solutions have received a lot of attention in the advanced persistent threat (APT) market for identifying zero-day malware that antivirus products cannot. Unfortunately, they are not perfect. The Angler exploit, for instance, was able to evade a standalone sandbox solution. If the exploit kit learned it was being run in a virtual box, or in a VMware or Parallels Desktop environment, it could evade detection.
- **Web-Filtering Software:** Web filtering classifies websites into categories based on their content and reputation for malicious behavior. This approach requires constant updates, making it difficult to keep pace with drive-by downloads.
- **Software Updates:** Operating systems and software vendors recommend frequent updates to defend against potential attacks. Although this is essential for keeping malware at bay, it is not sufficient in and of itself. With attackers eager to update their exploit kits based on new found vulnerabilities, software patches cannot be installed quickly enough to defend against exploits, leaving periods of vulnerability. Some updates involve human intervention and interruption of current tasks, which increases the time required to install the patches.

Each of these solutions tries to convince users that it will protect them from drive-by downloads. The bottom line, however, is that adding more functionality or signatures to products that were originally designed to detect viruses or malicious websites is no match for the sophisticated attacks we have seen in the past few years.

©2018, Juniper Networks, Inc.

The Juniper Advanced Threat Prevention Appliance Solution

The Adaptive Detection Fabric (ADF) on Juniper Networks® Advanced Threat Prevention Appliance is designed to address the dynamic nature of advanced threats by continuously collecting Web traffic and performing multistage analysis within its SmartCore engine. For a complicated problem such as drive-by downloads, a single, traditional approach will not do the trick.

The Juniper ATP Appliance attacks the problem from multiple angles:

- **Chain Heuristics:** The Juniper ATP Appliance uses a heuristics model to identify potentially malicious traffic. As employees of any given company visit literally thousands of webpages each day, this is a crucial step for focusing on suspicious traffic and providing quick results. The Juniper ATP Appliance analyzes all traffic and looks for indicators such as whether a browser is running a vulnerable version of a plugin, a webpage was referred from a valid resource link, a field is missing from a header, a webpage is part of a trusted domain, and many similar questions.
- **Browser Behavior Analysis Engine:** If a particular HTTP session is considered potentially malicious by the heuristics model, more analysis is required. The entire session is simulated using a browser running in the Juniper ATP Appliance's sandbox, where browser logs and downloaded artifacts are examined to confirm any suspicious activity.
- **Dropper Analysis:** The Juniper ATP Appliance looks for any executable artifacts (droppers) that were downloaded as part of the chain, subjecting the dropper to static, behavioral, and reputation analysis to determine whether it is malware.

The true strength of the Juniper ATP Appliance when combating drive-by downloads lies in using a combination of these techniques to counter various kinds of exploits. Each exploit has its own traits, or "tells," and it would be difficult to detect them all with a single method approach. It is important to know what to look for; otherwise, the search could easily end up being a wild goose chase.

These clues are subtle and spread across several requests and responses. Chain heuristics do not look at packets as mere 0s and 1s that it can match a signature against; it understands the context by inspecting the sequence of HTTP requests and responses between a particular source and destination. Each of these sequences is called a chain, and chain heuristics check for suspicious indicators in the headers and body of each HTTP request and response, as well as overall in each chain.

Suspicious indicators are constantly updated, depending on which exploits are out there. Juniper Threat Labs researchers study new exploits in the wild and come up with these indicators. The indicators by themselves may not draw attention, but when all the indicators are added up along with enough context, things will start to look suspicious.

For example, consider an endpoint in an enterprise that fetches a few webpages from an outside Web server hosted on port 8000. On its surface, that doesn't seem suspicious; a lot of Web servers run on nonstandard ports for enhanced security. However, if the same endpoint also downloads an encrypted executable file and its browser runs a vulnerable version of a browser plugin, then red flags begin to appear. The strength of chain heuristics lies in the context extracted from the traffic. When threat intelligence data from Juniper Threat Labs' research team is combined with heuristics, the solution offers a unique perspective on the problem.

Depending on the verdict obtained from chain heuristics, the ATP Appliance decides whether the suspicious chain needs to be looked at by the Browser Behavior Analysis Engine. It recreates the attack by executing the suspicious HTTP session in a browser environment on the ATP Appliance's sandbox. As the browser requests the session webpages, the exact responses captured as part of the session are served. Using this method, the Juniper ATP Appliance can replicate the exploit as it happened on the infected endpoint.

The Juniper ATP Appliance looks for suspicious activity by examining the properties and function calls made by scripts executed on the browser. For example, if the ATP Appliance notices a particular script making a function call to check for device drivers, and the user was simply browsing news at the time, it raises a flag. The Juniper ATP Appliance also inspects the source code of Javascripts used as part of the exploit. This is a valuable source of information as it provides clues about the malware's actual intentions. The Browser Behavior Analysis Engine offers the ability to zoom in on the attack and observe how it happened step by step. Armed with information about where the attack originated, how it took place, and what it left behind, security administrators can take informed actions.

The Juniper ATP Appliance performs detailed analysis on malware payloads dropped on the endpoint. The executable is detonated in the ATP Appliance's array of sandboxes, its behavior is observed, and a machine learning analytics engine is used to render a verdict. It also allows customers to create custom behavior analysis sandboxes that mimic their actual endpoints, helping to assess the impact of malware in their environment. In addition, the Juniper ATP Appliance can detect callbacks made to external Command and Control servers (C&Cs). By using a multidimensional approach, the ATP Appliance can provide details about the severity of an attack and how far it has progressed

©2018, Juniper Networks, Inc.

Conclusion

Hacking is no longer an art; today, it is more like a commercial venture. Sophisticated malware is available for purchase on the darknet, while spear-phishing attacks target specific individuals in an enterprise, lay low for months, and don't stop until they reach the main vault. Social security numbers, home addresses, and personal e-mails now have price tags on them.

For a long time, enterprises were content with firewalls that defended their network and antivirus solutions that protected their endpoints. However, these solutions have become archaic and fail to effectively protect companies against sophisticated malware. The need for an advanced threat defense solution like the Juniper ATP Appliance has become crucial to protect the intellectual property of enterprises and the privacy of their employees.

About Juniper Networks

Juniper Networks brings simplicity to networking with products, solutions and services that connect the world. Through engineering innovation, we remove the constraints and complexities of networking in the cloud era to solve the toughest challenges our customers and partners face daily. At Juniper Networks, we believe that the network is a resource for sharing knowledge and human advancement that changes the world. We are committed to imagining groundbreaking ways to deliver automated, scalable and secure networks to move at the speed of business.

Corporate and Sales Headquarters
Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, CA 94089 USA
Phone: 888.JUNIPER (888.586.4737)
or +1.408.745.2000
Fax: +1.408.745.2100
www.juniper.net

APAC and EMEA Headquarters
Juniper Networks International B.V.
Boeing Avenue 240
1119 PZ Schiphol-Rijk
Amsterdam, The Netherlands
Phone: +31.0.207.125.700
Fax: +31.0.207.125.701

Copyright 2018 Juniper Networks, Inc. All rights reserved. Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners. Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

2000705-001-EN Apr 2018