The 4-Minute Guide to Enterprise Security Threats

E-Guide

SentinelOne

Introduction

When traditional signature-based antivirus is pitted against malware that's ordinarily found in the wild, the effect could be compared to a horse-and-buggy trying to compete against a Formula One racer. Unless the faster vehicle crashes, there's no way for the buggy to catch up. It's sad to say, but the ordinary run of antivirus countermeasures has been completely out-innovated by bad actors.

There are a few reasons why the hackers have blown past security administrators in this manner. First of all, there's volume. There are many more hackers than there are security professions, and they aren't constrained by NDAs or trade secrets. Hackers share with each other. Their secret tricks disseminate into the community at light speed, and the community refines and iterates these techniques in turn. One might almost think of their combined efforts as a frighteningly efficient evolutionary algorithm, furiously straining to create an ever-more-perfect lock pick for an extremely imperfect set of locks.

The other reason why hackers are able to breach security so easily these days is because the technology itself is fundamentally slow to react. Signature-based antivirus needs a human somewhere in the loop in order to function. The technology requires someone to write or validate a rule that says, in effect, "if a file contains a certain hash, or a particular filename, or executes in this manner, then it is malware." As shown in this document, however, bad actors can modify their malware to evade signature-based detection, and they can do it much faster than security companies can write new signatures.

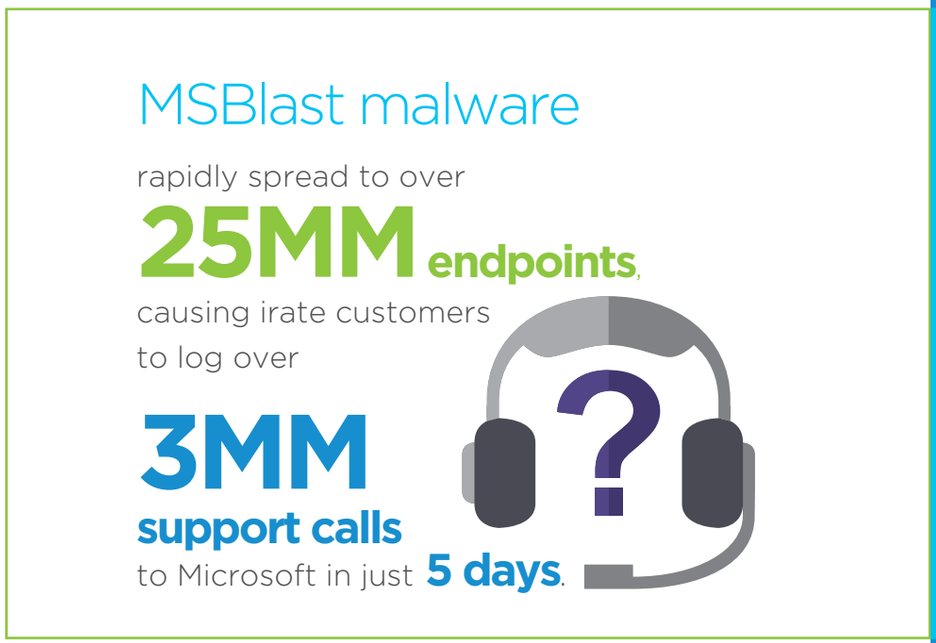
Over the next few pages, SentinelOne has compiled a list of the most damaging tools and techniques that are commonly seen in the wild. Learn how they are used to attack the enterprise and steal sensitive data—and how traditional security tools can do little to stop them.

Sophisticated Malware

DISK-BASED EXECUTABLES

Disk-based executables are the most common type of malware in which the malicious payload is written to disk. These types of attacks are typically easier to detect, as they leave remnants, or files behind, that can often be detected with endpoint security software. However, malware has evolved beyond early viruses that could be easily detected with signatures. Sophisticated examples include Trojans and worms that have caused large scale damage to organizations and their customers.

One of the most damaging worms was the MSBlast malware that rapidly spread to over 25 million endpoints, causing irate customers to log over 3 million support calls to Microsoft in just 5 days. A more recent example is BlackEnergy, a Trojan that crippled energy SCADA systems across Europe in late 2015.



FILE-LESS ATTACKS

A common mechanism to evade malware detection mechanisms is to not leave any tracks on the hard disk. Hackers accomplish this by loading malware directly into the endpoint's memory, or hiding it pointers and files in the system registry. Poweliks, which is a registry resident malware that tries to steal credentials and other critical data, is an example of a file-less threat vector that has bypassed malware detection products.

Vulnerabilities and Exploits

DOCUMENT/APPLICATION-BASED ATTACKS

Hackers have been using the familiarity of well known document formats to distribute malicious payloads for decades. Microsoft Office (MS Word, Excel) and Adobe PDF documents are a common vector in which malware can be embedded, distributed and then covertly activated when these documents are opened.

Some of the most damaging examples of malware have originated as infected documents targeting specific organizations with spear-phishing and eventually costing billions of dollars. Detecting such attacks is extremely difficult since a minor change to the carrying file – a name change, manipulating the contents etc. makes the attack vector invisible to traditional detection mechanisms.

a minor change to the carrying file – a name change, manipulating the contents etc. makes the attack vector **invisible** to traditional detection mechanisms.

BROWSER-BASED ATTACKS

Internet browsers have become an integral part of computing today, acting as the primary user interface for virtually any kind of application. Internet users rely on browsers to create documents, interact with their financial institutions, search for information, buy and sell goods and services, watch “TV”, play online games and much more.

Hackers therefore have also taken note of this trend and created sophisticated attacks targeting web browsers. Browser malware can use the programming power of JavaScript/VBScript, browser extensibility through plug-ins. An extreme case is with the eFast browser that completely replaces and mimics Google Chrome while opening backdoors and stealing credentials and other data from the victim’s endpoint device.

eFast browser completely **replaces and mimics** Google Chrome while **opening backdoors and stealing credentials** and other data from the victim’s endpoint device.

Live User/Insider Attacks

SCRIPT-ORIENTED ATTACKS

Scripting and command-line automation and system orchestration tools pack a lot of programming power. However, they also open the door for hackers to programmatically access deep system capabilities. PowerShell and WMI based attacks leverage this power and access, couple it with system credentials to manipulate system files, logs, user access, registry settings and much more.

Poweliks, mentioned earlier, uses this technique to execute. In fact it will download and install PowerShell if it does not detect the framework on its target endpoint. Once installed, Poweliks will actively evade detection by disguising network traffic to command and control servers, prevent detection tools from executing and hence remain hidden from signature-based detection tools.

LIVE/INSIDER ATTACKS

Subverting an insider, or an insider's credentials remains one of the easiest and stealthiest – and hence one of the most effective methods to conduct a cybersecurity attack on an organization.

The 2015 Verizon Data Breach Investigations Report (DBIR) found that insider privilege abuse was **the cause for 55% of all incidents**. Hackers go about compromising insider accounts in a number of ways. In addition to memory scraping and network sniffing hacks designed to pick up usernames and passwords, tools like Windows' "runas" and Unix su allow privilege escalation. Once any inside access has been established by malicious software, hackers start to covertly escalate account privileges, access additional systems and open back door access mechanisms that don't trigger any flags on legacy detection tools.

Conclusion

As shown, the number of tools, techniques, and exploits that bad actors use to breach the enterprise has proliferated beyond all reason. When hackers are commonly able to spin up an entire malicious browser that invisibly steals user data, how are traditional security tools supposed to cope?

They aren't. Current-gen malware can easily be altered to evade signature-based detection, and has now evolved to detect sandboxing.
Even if traditional endpoint protection is able to block malicious software based on its signature, it can do nothing against even simpler attacks. Attackers can do a complete end-run around signature-based detection by stealing a user's credentials, an act that no amount of security-awareness training has ever been able to prevent.

A new kind of security must recognize a potential threat not because of the way it looks, but because of the way it behaves. This kind of security must natively incorporate the capability to recognize any threat, from malware to compromised credentials, while incorporating digital forensics. This capability requires a new baseline technology: machine learning and behavioral analysis. The industry must rapidly switch over to this new reality, or else bad actors will continue to breeze by layers of defense-in-depth as though they weren't even there.