White Paper

Less Than Zero:

A Survey of Zero-day Attacks in 2013 and What They Say About the Traditional Security Model

Contents

Introduction

Of all the hazards confronting enterprise IT systems, zero-day vulnerabilities are among the most pernicious and dangerous. By definition, they are unknown and unpredictable, exposing systems of even the most diligent users and administrators.

Zero-day vulnerabilities are software flaws that leave users exposed to cyber attacks before a patch or workaround is available. Sometimes, a zero-day vulnerability is unknown to anyone but a cyber attacker (or a supplier who sells zero-day discoveries on the black market). In other cases, the software vendor knows about the vulnerability but has not yet issued a fix.

In either case, the result is the same: users, even those with big-ticket defenses in place, are wide-open to attack. While all software probably has unknown vulnerabilities, these flaws become particularly threatening "in the wild" when discovered by attackers and used to launch cyber assaults.

Patching your systems will not stop them. Updating malware definitions on anti-virus (AV) software will not stop them. In many cases, not even multi-layered "defense-in-depth" security schemes are enough to prevent a zero-day attack from hitting your IT assets.

"There is almost no defense against a zero-day attack," as one 2012 security report puts it. "While the vulnerability remains unknown, the software affected cannot be patched, and anti-virus products cannot detect the attack through signature-based scanning."[1]

This paper explains the dangers of zero-day attacks and why traditional defenses are powerless against them. It also outlines 11 zero-day attacks discovered by FireEye in 2013 and how they were used in real-world attacks. Finally, the paper recommends nine practical steps for mitigating the risks of zero-day attacks.

Zero-days alarmingly common

Zero-day threats are everywhere. On any given day over the last three years, cybercriminals had access to at least 85 vulnerabilities targeting widely used software from Microsoft, Apple, Oracle, and Adobe.[2] That estimate includes only vulnerabilities that were eventually reported. The true number of zero-day vulnerabilities available to cybercriminals could be much higher.

Vulnerabilities discovered by cybercriminals remain unknown to the public—including vendors of the vulnerable software—for an average of 310 days.[3]

Not surprisingly, zero-day exploits are heavily used in targeted attacks. These secret weapons give attackers a crucial advantage over their targets, even those that have invested hundreds of thousands of dollars into traditional security products.

And thanks to an abundance of zero-day vulnerabilities and increasingly mature global black market for exploits, these weapons are proliferating. Governments remain the top buyers of zero-day exploits, according to a recent Reuters article.[4] But anyone with enough money—as little as $5,000 in some cases[5]—can purchase one.

[1] Leyla Bilge and Tudor Dumitras (ACM Conference on Computer and Communications Security). "Before We Knew It: An Empirical Study of Zero-Day Attacks In The Real World." October 2012.

[2] Kelly Jackson Higgins (Security Dark Reading). "Hacking The Zero-Day Vulnerability Market." December 2013.

[3] Andy Greenberg (*Forbes*). "Hackers Exploit 'Zero-Day' Bugs For 10 Months On Average Before They're Exposed." October 2012.

[4] Joseph Menn (Reuters). "Special Report: U.S. cyberwar strategy stokes fear of blowback." May 2013.

[5] Andy Greenberg (*Forbes*). "Shopping For Zero-Days: A Price List For Hackers' Secret Software Exploits." March 2012.

Zero-day used in highly destructive campaigns

Zero-day exploits have enabled some of the most destructive and high-profile attacks in recent years.

Operation Aurora

One of the best known was Operation Aurora, a 2009 campaign whose more than 20 targets included Google, Adobe Systems, Juniper Networks, Rackspace, Yahoo, Symantec, Northrop Grumman, Morgan Stanley, and Dow Chemical.[6]

An Internet Explorer vulnerability, CVE-2010-0249, served as the entry point. In this watering-hole attack, the threat actors compromised a legitimate website with malicious JavaScript that exploited the vulnerability. Visitors to the site unknowingly received malware that stole intellectual property, compromised user accounts, and spied on targets.

Stuxnet

The 2010 Stuxnet worm used at least three separate zero-day exploits—an unprecedented feat—to damage industrial controllers and disrupt Iran's Natanz uranium enrichment facility.[7]

The zero-day vulnerabilities included the following:

- **CVE-2010-2568**—Executes arbitrary code when user opens a folder with maliciously crafted .LNK or .PIF file
- **CVE-2010-2729**—Executes arbitrary code when attacker sends a specially crafted remote procedure call (RPC) message
- **CVE-2010-2772**—Allows local users to access a back-end database and gain privileges in Siemens Simatic WinCC and PCS 7 SCADA system

The attack also exploited already patched vulnerabilities, suggesting that the attackers knew the systems likely would not have been updated.

RSA Attack

In 2011 attackers breached EMC's RSA Security division and stole secrets related to its widely used authentication system. The data theft had the potential of rendering the SecurID token authentication systems less effective—and leaving countless RSA customers less secure.[8]

Attackers sent phishing emails with a weaponized Microsoft Excel file labeled "2011 Recruitment Plan" to RSA employees. The file had a malicious Adobe Flash object that exploited the CVE-2011-0609 vulnerability, allowing attackers to install the Poison Ivy remote-access tool (RAT). With Poison Ivy, the attackers gained login credentials of high-profile targets with access to RSA's authentication information.

"[C]ompanies deploy any imaginable combination of state-of-the-art perimeter and end-point security controls, and use all imaginable combinations of security operations and security controls," wrote Uri Rivner, who was an RSA executive at the time, in a blog post. "Yet still the determined attackers find their way in. What does that tell you?"[9]

[6] Wikipedia. "Operation Aurora." October 2013.

[7] Gregg Keizer (*InfoWorld*). "Is Stuxnet the 'best' malware ever?" September 2010.

[8] John Markoff (*The New York Times*). "SecurID Company Suffers a Breach of Data Security." March 2011.

[9] Uri Rivner (RSA). "Anatomy of an Attack." April 2011.

Standard defenses are powerless against zero-day threats

Rivner does not answer that question directly in the blog post, but one implication is clear: traditional security tools are no match for zero-day exploits.

Traditional security tools rely on malware binary signatures or the reputation of outside URLs and servers. By definition, these defenses identify only known, confirmed threats. An attacker can easily hijack a legitimate website to bypass a blacklist. Code morphing and obfuscation techniques generate new malware variants faster than traditional security firms can generate new signatures. And spam filters will not stop low-volume, targeted spear-phishing attacks.

At the same time, operating system-level protections such as Address Space Layout Randomization (ASLR) and Data Execution Prevention (DEP) are becoming less effective. Several zero-day exploits discovered by FireEye in recent months used ASLR-bypassing methods that have all but neutered this once-effective protection.[10]

It's no wonder that the typical zero-day attack lasts an average of eight months—and can last close to three years in some cases.[11] That gives attacks ample time to steal organizations' most valuable assets—and leave before anyone knows what happened.

FireEye Zero-day Discoveries In 2013

FireEye discovered and reported 11 zero-day vulnerabilities in 2013—by far the most of any security company. Among the top 10 cyber security companies (ranked by security-related revenue), only two other zero-day vulnerability were reported that year. This gap underscores the difficulty in detecting zero-day attacks, especially by traditional cyber defenses.

Here are the FireEye-reported zero-day vulnerabilities, which are explained in the corresponding sections:

- CVE-2012-4792
- CVE-2013-0422
- CVE-2013-0634
- CVE-2013-0640 / CVE-2013-0641
- CVE-2013-1493
- CVE-2013-1347
- CVE-2013-3893
- CVE-2013-5065
- CVE-2013-3918 / CVE-2014-0266

[10] Xiaobo Chen (FireEye). "ASLR Bypass Apocalypse in Recent Zero-Day Exploits." October 2013.

[11] Leyla Bilge and Tudor Dumitras (ACM Conference on Computer and Communications Security). "Before We Knew It: An Empirical Study of Zero-Day Attacks In The Real World." October 2012.

CVE-2012-4792

Discovered close to New Year's Day,[12] this vulnerability allows attackers to execute code on machines of Internet Explorer users who visit maliciously crafted websites. The exploit takes advantage of a "use-after-free" flaw, in which faulty code tries to improperly access memory that has been "freed up" for another purpose. The problem can allow attackers to remotely execute malicious code on a targeted system.

In this case, the malicious code was JavaScript hidden in the website of Council on Foreign Relations (CFR), a popular foreign-policy website. The intended targets are unknown, but CFR members include Secretary of State Sen. John Kerry, Senate Intelligence Committee Chairman Sen. Dianne Feinstein (D., Calif.), and former Secretary of State Hillary Clinton.[13]

CVE-2013-0422

CVE-2013-0422, which was exploited as early as Jan. 2, 2013,[14] is a Java 7 vulnerability that allows attacks to bypass Java security checks and executes code on a target machine. Though designed for Windows, the vulnerability probably left other operating systems that run Java open to attack.

Attacks exploiting this flaw download ransomware known as Tobfy. This ransomware locks users out of their computer, displaying a full-screen message, purportedly from the FBI, accusing the user of a crime and demanding a payment to unlock the machine. Tobfy also disables Windows Safe Mode and terminates processes such as taskmgr.exe, msconfig.exe, regedit.exe, and cmd.exe to deter users from trying to find or disable the malware.

Because of a coding mistake, the malware cannot communicate with the attacker—including communication that indicates whether the user has paid the ransom and should remove itself from the system. So even if targets paid up, they still could not access their PC.

CVE-2013-0634

Identified on Feb. 7, 2013,[15] this Adobe Flash vulnerability allows attackers to run malicious ActionScript code on Windows, Macs, Linux, and even Android mobile devices.[16]

The exploit was used in a cyber espionage campaign dubbed "LadyBoyle." Attacks exploiting this vulnerability sent Microsoft Word documents to target Windows users. Though the contents of Word files are in English, the codepage of Word files are Windows Simplified Chinese (PRC, Singapore). The Word files contain a macro to load an embedded SWF Flash object.

The SWF file, in turn, contains an action script with the name "LadyBoyle" that contains the exploit code. The exploit supports only a limited version of Flash, and it checks for the presence of ActiveX component, a Windows-only feature.

Once it has reached an appropriate target, the SWF file exploits the Flash Player vulnerability to run the payload DLL and embedded executables. The payload itself was from a known malware family that had been used in previous campaigns. One of the dropped executable files is digitally signed with an invalid certificate from MGAME Corporation, a Korean gaming company. The executable renames itself to try to pass itself off as the Google update process.

[12] Darien Kindlund (FireEye). "CFR Watering Hole Attack Details." December 2012.

[13] Bill Gertz (*The Washington Free Beacon*). "Chinese Hackers Suspected in Cyber Attack on Council on Foreign Relations." December 2012.

[14] Yichong Lin (FireEye). "Happy New Year from New Java Zero-Day." January 2013.

[15] Thoufique Haq and J. Gomez (FireEye). "LadyBoyle Comes to Town with a New Exploit." February 2013.

[16] National Vulnerability Database. "Vulnerability Summary for CVE-2013-0422." February 2013.

From there, the malware creates a startup entry (so it can restart after a reboot) and checks for anti-virus (AV) software. Oddly—and perhaps sloppily—the malware payload was not encrypted or obfuscated.

CVE-2013-0640 / CVE-2013-0641

Attackers exploited this pair of PDF vulnerabilities to install a remote administration tool (RAT) with a flexible, extensible architecture. Attackers can add new features easily with plug-in dynamic-link libraries (DLLs). The malware shellcode bypassed ASLR and DEP security features, upping the ante in the security arms race.

The JavaScript embedded in the crafted PDF is highly obfuscated using string-manipulation techniques. Most of the variables in the JavaScript are in Italian. The JavaScript has version checks for various versions of Adobe Reader as shown below and it creates the appropriate shellcode based on the version found.

Attackers dropped three DLLs onto the target system that work in harmony to steal information. The main component (LangBar) inserts itself into Windows processes and helps coordinate other DLLs used in the attack. The second DLL (lbarhlp) performs most of the data-stealing functions. And the third DLL (lbarext) packages and encrypts the stolen data.

The malware, dubbed "666," was used in a spear-phishing email campaign against Japanese targets. The emails contained a weaponized PDF attachment purporting to be a security report.

CVE-2013-1493

This Java Runtime Environment vulnerability allows attackers to compromise the integrity of the HotSpot virtual machine and bypass Java's SecurityManager to manipulate heap memory and execute malicious code.[17]

Attacks exploiting the vulnerability downloaded the McRAT Trojan to give attackers control over the targeted systems. The exploit was unusual because it allowed attackers to read and write memory directly within the Java Virtual Machine process. It also reflected the trend of targeted attacks bypassing ASLR, which until then had been one of the most effective safeguards for operating systems.

The zero-day exploit was one of three (along with Internet Explorer zero-day vulnerability CVE-2013-1347 and Java exploit CVE-2013-2423) used in the Sunshop campaign over the summer.[18]

Sunshop compromised several strategic websites including:

- Multiple Korean military and strategy think tanks
- A Uyghur news and discussion forum
- A science and technology policy journal
- A website for evangelical students

[17] National Vulnerability Database. "Vulnerability Summary for CVE-2013-1493." March 2013.

[18] Ned Moran (FireEye). "Ready for Summer: The Sunshop Campaign." May 2013.

CVE-2013-1347

Like CVE-2013-1493, this IE vulnerability was used in the Sunshop campaign.[19] It uses an ASLR-bypass technique to exploit a use-after-free vulnerability in IE versions 6 through 8 running on Windows XP.

In addition to Sunshop-related attacks, the exploit was used in a watering hole attack against visitors to the U.S. Department of Labor website, typically federal employees.[20] JavaScript embedded into the site redirected visitors to a site hosting the Poison Ivy RAT, which gives attackers control of target systems.

CVE-2013-3893

This IE vulnerability allows attackers to execute code on machines of users who visit maliciously crafted websites. It anchored a malware campaign dubbed "Deputy Dog."

The campaign began as early as August and targeted organizations in Japan. At least three other APT campaigns—dubbed Web2Crew, Taidoor, and th3bug—used the same exploit.

CVE-2013-3918 / CVE-2014-0266

These ActiveX vulnerabilities, which affected nearly every version of Windows since XP Service Pack 2, allows attackers to execute malicious code on machines of IE users who visited maliciously crafted websites.

Attackers used this vulnerability in an "exceptionally accomplished and elusive" watering hole attack dubbed "Operation Ephemeral Hydra," which targeted a strategically important website, known to draw visitors that are likely interested in national and international security policy.[21] The attack used the same infrastructure as the DeputyDog campaign (see CVE-2013-3893); the Trojan used in both attacks included a text string that also appeared in the infamous Operation Aurora attacks.

In a twist likely to make detection, forensics, and remediation tougher, the attackers loaded the payload in this attack directly into computer memory without writing anything to disk. So when an infected computer reboots, nearly all traces of the attack disappear.[22]

CVE-2013-5065

Identified on November 27, this Windows XP and Windows Server 2003 vulnerability escalates local-user privileges to allow a standard user account to execute code in the kernel.[23] Although this vulnerability does not allow attackers to execute code remotely, remote attackers can use it in conjunction with other vulnerabilities to that end.

Attackers exploited CVE-2013-5065 along with CVE-2013-3346, an Adobe Reader vulnerability patched in May. The targeted attacks used a weaponized PDF to drop the malware payload into a temporary directory in Windows and execute it.

[19] Yichong Lin (FireEye). "IE Zero-Day is Used in DoL Watering Hole Attack." May 2013.

[20] Michael Mimoso (*Threat Post*). "Watering Hole Attack Claims US Department of Labor Website." May 2013.

[21] Ned Moran, et al (FireEye). "Operation Ephemeral Hydra: IE Zero-Day Linked to DeputyDog Uses Diskless Method." November 2013.

[22] Ms. Smith (*NetworkWorld*). "IE zero-day attack delivers malware into memory then poofs on reboot." November 2013.

[23] Xiaobo Chen and Dan Caselden (FireEye). "MS Windows Local Privilege Escalation Zero-Day in The Wild." November 2013.

Conclusion

The zero-day vulnerabilities discovered in 2013 reflect several trends that should prompt organizations to reassess their security posture:

- Operating system-level protections are becoming less effective against zero-day attacks. ASLR and DEP were big steps forward, but attackers are finding ways around them.
- Watering hole attacks are growing more common. By compromising trusted websites that cater to well-defined audiences, attackers can target precise industry or government segments. And rather than having to find ways into targeted systems, attackers can wait for the targets to come to them.
- Attacks are growing more sophisticated. Random crimeware and clumsy, high-volume attacks still occur. But laser-focused attacks against high-value targets are mushrooming. And these attacks are becoming much more adept at bypassing organizations' defenses.

Recommendations

Defending your IT assets against zero-day threats requires a fundamentally new approach to cyber security. Yesterday's signature-based defenses are not equipped for today's tidal wave of exploits. Reputation-based defenses are not designed to detect brand-new attacks or those that commandeer trusted websites and servers to do their dirty work. And file-based sandboxes, which are easily fooled by the newest generation of malware, often miss zero-day attacks.

Today's security professionals must equip themselves for not only known threats, but the new reality of unknown threats. To paraphrase the ancient Greek philosopher Heraclitus: expect the unexpected.[24]

To that end, FireEye recommends the following:

- **Segment your networks.** Limit access between network segments with different risk profiles. This step includes limiting access from the Internet to the DMZ, the DMZ to the internal network, and so on. It also includes preventing systems in one functional unit from accessing systems in another when that access is not required. An example: preventing systems in the finance department from accessing systems in the engineering department. This move can block an attacker's access to an unpatched vulnerability.
- **Limit network privileges.** Users and applications should access only the information and resources that are required to function properly. This step can shrink the attack surface, because some attacks require elevated privileges to work. It also reduces the risk posed to the environment by a successful attack by reducing the attacker's ability to access systems or information.
- **Use application whitelisting.** By allowing user to install only preapproved applications, you prevent unauthorized files from executing, including some executable exploits and malware payloads.
- **Have an incident response (IR) plan in place.** By definition, you cannot predict a zero-day attack. This uncertainty makes a robust, resilient IR plan even more crucial. By quickly detecting an attack and having a defined, tested IR response at the ready, security professionals can mitigate any damage.

[24] Heraclitus (Edited by Charles H. Kahn). "The Art and Thought of Heraclitus : An Edition of the Fragments." 1979.

- **Know your environment.** Security teams cannot hope to mitigate the risk of an application with an unpatched vulnerability unless they know the application is present and understand the network well enough to put an effective mitigation plan in place.
- **Deploy a security platform that identifies both known and unknown threats.** Security experts widely agree that signature-based defenses are toothless against today's fast-moving, ever-evolving threats.[25] Signature-based defenses work only for threats that have been discovered and documented.

  Likewise, reputation-based defenses, by design, stop only known threats. Even file-based sandbox technology, touted as a fresh approach to security, cannot provide the deep insight required to block zero-day attacks. Zero-day attacks call for new technologies built from the ground up for today's advanced threat landscape.
- **Keep your systems patched.** The security team should apply the latest patches and audit the environment for missing patches. No, this step will not in itself protect your systems from zero-day attacks. But many organizations remain vulnerable to already fixed zero-day vulnerabilities simply because they have failed to fully patch their systems.
- **Use operating systems and applications that support DEP and ASLR.** As explained earlier, more zero-day attacks are bypassing DEP and ASLR protections. So this step is not a cure-all. But when the operating system and applications support DEP and ASLR, exploiting vulnerabilities becomes significantly tougher. When possible, organizations should use the newest operating system releases, which usually incorporate new techniques to mitigate threats.
- **Foster more collaboration in the security industry.** Zero-day attacks move fast. The good guys need to move faster. To identity and counter zero-day exploits more quickly, the security industry must collaborate more often and more seamlessly. By sharing intelligence and quickly sounding the alarm, the community can contain the damage—and make everyone collectively safer.

[25] Gartner. "Best Practices for Mitigating Advanced Persistent Threats." January 2012.

About FireEye, Inc.

FireEye has invented a purpose-built, virtual machine-based security platform that provides real-time threat protection to enterprises and governments worldwide against the next generation of cyber attacks. These highly sophisticated cyber attacks easily circumvent traditional signature-based defenses, such as next-generation firewalls, IPS, anti-virus, and gateways. The FireEye Threat Prevention Platform provides real-time, dynamic threat protection without the use of signatures to protect an organization across the primary threat vectors and across the different stages of an attack life cycle. The core of the FireEye platform is a virtual execution engine, complemented by dynamic threat intelligence, to identify and block cyber attacks in real time. FireEye has over 1,900 customers across more than 60 countries, including over 130 of the Fortune 500.