

A Tale of Three Bogeys: Zero-day Vulnerabilities, Unpatched Client Applications, and Browser Exploits

Defending your systems by understanding the good, the bad, and the unknown

Table of Contents

| | |
|---|---|
| More Malware X More Vulnerabilities = More Work (or More Risk) | 3 |
| Bogey Number One: Unknown Zero-day Vulnerabilities | 3 |
| Situation: Zero-day vulnerabilities in the Adobe Acrobat Reader | 3 |
| Remedy | 4 |
| Bogey Number Two: Unpatched, Known Vulnerabilities in the Client | 4 |
| Situation: Unpatched PowerPoint | 5 |
| Remedy | 5 |
| Bogey Number Three: Browser-based Exploits | 6 |
| Situation: DirectShow exploit in Internet Explorer | 6 |
| Remedy | 6 |
| Conclusion | 7 |

The year 2009 demonstrated outrageous malware volumes made more potent through clever exploit of vulnerabilities, both dreaded zero-day vulnerabilities and known, patchable client-side vulnerabilities in standard desktop applications and browsers. Through appropriate deployment of protections, IT teams can build up an integrated base of countermeasures to eliminate fear of the unknown, while protecting against the bad and enabling the good, both good code and the successful operation of your business.

More Malware X More Vulnerabilities = More Work (or More Risk)

Do you want more work for IT or do you prefer more risk? Which option can your business better afford? As of October 1, 2009, McAfee Labs was projecting 2009 would deliver three million new pieces of malware, doubling the record set in 2008. By September 24, 2009, the National Institute of Standards and Technology (NIST) National Vulnerability Database (NVD) had archived more than 39,000 CVE vulnerabilities, with nineteen new vulnerabilities being published every day.¹

The math is simple: more malware/exploits + more vulnerabilities = more work for IT (to patch holes, repair systems, or install new protections) or more risk for business (of lost data, customers, and downtime).

This equation is not new, but the urgency has climbed. The high volume of malware and vulnerabilities increases the likelihood something will affect your users and systems. With more complex web applications and more users browsing the web on business systems, the likelihood of a breach or major infection increases dramatically.

In this paper, we describe three common bogeys, events IT teams must target to protect themselves and prevent costly cleanup. Our goals are to:

- Show you what the issues are
- Build your understanding of how malware and vulnerabilities are used together for a bigger payoff
- Explain how to use a mix of new and traditional tools to mitigate these risks

Bogey Number One: Unknown Zero-day Vulnerabilities

Our first challenge covers vulnerabilities that are unknown by the good guys: product vendors, security researchers, or IT teams. As operating systems, devices, and applications proliferate, this problem merits and receives tremendous attention from researchers and the press.

Situation: Zero-day vulnerabilities in the Adobe Acrobat Reader

A cybercrime network discovered—or, more common these days, probably purchased—a way to exploit an unpublished bug in the Adobe Acrobat Reader. Our example, a real instance in January 2009, saw the attack take advantage of a bug in the Adobe Reader to overwrite the memory, and then use JavaScript to take control and execute code that installed a backdoor. Through this backdoor, the attacker enforced remote control and monitoring on the infected system. They might have had an eye to data theft or disruption, or hoped to use the system to penetrate farther into the compromised network in search of bigger rewards, a likely goal of cyberespionage.²

Malware plays a key part in the success of this type of zero-day exploit. If the criminal were attempting a targeted attack, the malware might be delivered through a phishing email made to look like it came from a co-worker or friend. If the criminal were interested in capturing user data, the attack might be planted within a PDF file downloadable on a website devoted to financial news (presumably being read by people with money and bank accounts).

1. <http://nvd.nist.gov>.

2. Vulnerabilities like this are documented in the McAfee Labs blogs and throughout the web:
<http://www.avertlabs.com/research/blog/index.php/2009/02/19/new-backdoor-attacks-using-pdf-documents>
<http://www.avertlabs.com/research/blog/index.php/2009/07/22/new-0-day-attacks-using-pdf-documents/>
<http://isc.sans.org/diary.html?storyid=6286>
<http://www.h-online.com/security/Zero-day-vulnerability-in-Adobe-Flash-Player-Reader-and-Acrobat--/news/113828>

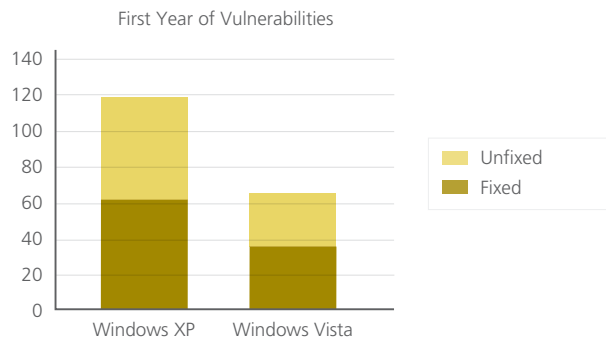
Remedy

Zero-day flaws, by definition, have no patch or fix, so IT should protect desktops and servers with host intrusion prevention, which can include several beneficial layers to increase the chances of catching and fending off the threat. We call these layers countermeasures. In addition, a web security tool might help to keep users from visiting a risky site. We'll start with the web tool as an easy first line of defense.

- **Web rating service**—A web security service can use automated crawlers, analytics, researchers, and user ratings to identify sites that have risky code on them, that link to sites with risky code, or that spam users untutored enough to provide their email addresses. Depending on the breadth and detail of this analysis, a service like this might flag a site known to host suspicious code. The service might block poorly-rated sites to reduce the most obvious risks. In this example, since the malware might not have been on a website or that site might not yet have been classified as risky, we would recommend this tactic as a first line of defense only.
- **Host intrusion prevention behavioral analysis**—Behavioral analysis detects suspicious patterns and activities, applying complex heuristics to identify potentially risky activities, such as illegal API calls, unexpected or inappropriate use of memory, or unusual communications with other systems. If it sees something it doesn't like, the intrusion prevention system keeps the attack from passing through the kernel to its destination, essentially blocking the activity. In our example, behavioral analysis would look at the attempt to overwrite the memory and squelch the action before it occurs.
- **System firewall (can be part of host intrusion prevention)**—A stateful firewall applies policies, bars unsolicited inbound traffic, and controls outbound traffic. In this case, it could prevent both the outbound and the inbound communication through the backdoor. Depending on the value of the asset and the way it is used, you can do more. For example, if you are protecting a system housing sensitive corporate data, you could set up a rule to allow only network transmissions within a trusted network. The firewall would block transmissions outside your network, again thwarting the malicious code.

Bogey Number Two: Unpatched, Known Vulnerabilities in the Client

It takes time to develop patches once a vulnerability is uncovered. Some operating system and application patches take a long time to receive a patch, usually because they perceive a given bug to be less severe or the software less prevalent.³ Some older systems may stop receiving patches. Some bugs will never be patched. Behavioral techniques in intrusion prevention will remain important for these unpatched situations.



While Microsoft may be getting better about finding bugs before they ship, their products still have vulnerabilities, and many of them go unpatched.⁴

When patches become available, Microsoft's scheduled patch release program, known as Patch Tuesday, means some companies track and rigorously (or better still, automatically) install patches for operating systems and Microsoft applications. More and more, other server- and client-side application-layer

vendors, including Oracle and Adobe, are moving to this scheduled release approach, which has the advantage of allowing companies to have formal patch testing and distribution programs.

However, the published schedule also allows attackers to plan. They can capitalize on the window between the time the vulnerability is discovered and the time the signature or patch is actually installed.

Nowadays, many attackers increase their hit rate through a focus on the client. They put their energies here because the client is now a weaker link than servers, since servers tend to be patched first and protected most aggressively because of their value as a shared resource.

Within the client, many attackers focus on applications more than operating systems. There are more applications, with more variations, and therefore more opportunities for a hole—a vulnerability, misconfiguration, out-of-date DAT, or disabled protection—that they can exploit if they can get in with malware. Let's look at how companies can efficiently protect their clients by doing a better job against known vulnerabilities and malware.

Situation: Unpatched PowerPoint

On September 25, 2009, McAfee researchers documented a scam that takes advantage of a vulnerability that has been patchable since 2006.⁵ In this case, spammers embedded an exploit within a PowerPoint file. They cloaked the content in an email that claimed the file contained details of aircraft in the Pakistan Air Force. This social engineering tactic tempted recipients to click the file. If it opened using a vulnerable version of PowerPoint, the malware decrypted itself and executed a malicious binary, installing a backdoor that allowed remote code execution, which could mean installation of a keylogger, rootkit, or other malware.⁶

While this may seem like an easy problem to address with a patch, the fact that attackers took the time to develop the spam and the exploit shows that they believe plenty of systems remain vulnerable. Spammers and attackers only go where the money is, where they will have success getting in.

Remedy

Several different protections should help with this example, depending on how the email might penetrate your environment.

- **Anti-spam and anti-malware at the gateway**—Based on its knowledge of the message itself, the reputation of its source (the IP address), or its payload (if this PowerPoint attachment has been flagged already), anti-spam and anti-malware running at the gateway should be able to note and block this message before it enters the email server or reaches the client
- **Anti-malware on the client**—In case the malware is not recognized as spam or is somehow coming from within the network, perhaps from a compromised client on the LAN, you should have anti-malware (anti-virus and anti-spyware) installed on the system itself. If DAT downloads occur regularly (and most solutions download at least daily), they should blacklist, or block, based on the signature for the known vulnerability.
- **Host intrusion prevention on the client**—As a final line of defense, your host intrusion prevention system should also have received a signature for the vulnerability and be able to block the attempted installation of the backdoor
- **Application whitelisting on legacy and fixed function systems**—If you have an older system that is no longer receiving patches, you can't patch it. In addition, if you have a custom-built or limited function device, you may not be able to run anti-malware or intrusion prevention protections within its resource envelope or guarantee the connectivity for ongoing updates to keep it protected. For these situations, we recommend application whitelisting, where approved good code is the only code allowed to run. This type of protection can block unauthorized changes to applications, effectively locking down system configurations, so it would block the attempt to drop an executable into PowerPoint.

Bogey Number Three: Browser-based Exploits

Web 2.0 technologies have made websites risky through interactive, user-contributed content. Criminals take advantage of insecure websites—often not the Amazons and the Bank of England, but the second tier and all-volunteer sites with fewer security professionals on staff—and plant malware in dynamically loadable components, scripts, libraries, and attractive content, such as movies, pictures, and documents. Users can pick up this malicious content in a drive by, without noticing or accepting a download.

Making protection more challenging, sophisticated criminals now disguise or encrypt their code and parallelize their approach adopting the business continuity model of enterprise IT: if the first tactic doesn't succeed, they have a back-up plan.

Conficker was one of these multi-vector threats, propagating over the network through a Microsoft vulnerability, through email, by accessing unique web domains, and using a peer-to-peer protocol, as well as spreading directly at the endpoint through USB drives. A new multi-vector threat, eventually known as the DirectShow exploit, was detected in early July by McAfee researchers.

Situation: DirectShow exploit in Internet Explorer

Perhaps expecting U.S.-based threat researchers to be on vacation during the American Fourth of July holiday weekend, Chinese hackers hijacked innocent websites all over China and redirected visitors to sites they had compromised with malicious code. Their malicious Trojan downloaded silently, and then pulled down several different exploits. One took advantage of a zero-day vulnerability in Microsoft Direct Show, one hedged their bet by targeting a known vulnerability in another media player, RealPlayer, and others used known bugs in Internet Explorer 6 and 7 as well as an IE toolbar. McAfee researchers noted that the Trojan protected itself somewhat by checking if the visitor came from a Chinese government or education site.⁷

If the DirectShow exploit succeeded, it would gain the privileges of the user, often an administrator (the default for most Windows machines), allowing it to take over the machine or use these admin credentials to access other resources on the network. This sort of approach would help someone build a network, a botnet, of remotely controlled “zombie” computers.

Later that week, a similar vulnerability was disclosed, one that used an ActiveX control vulnerability and was distributed in compromised online ads on gaming sites.⁸

Remedy

Again, given the range of techniques used in this example, we recommend several reinforcing protections to help you shield your systems, users, and data.

- **Host intrusion prevention on the client**—Buffer overflow protection should have protected the use of memory by the browser, blocking the attempt to overwrite the memory; once installed, the signature would also provide protection as described in the previous section, beneficial in both host and network intrusion prevention systems
- **Anti-malware at the web gateway and on the client**—Behavioral, reputational, and other forms of real-time analysis would detect the spike in activity and the malicious pattern, elevating the risk assessment for the code and recommending a “block” to anti-malware systems;⁹ the DAT file, once installed, would also provide protection as described in the previous section; defenses layered at the gateway and on the client would protect against inbound transmission as well as re-distribution from an infected source within the network
- **Application whitelisting**—An additional strategy for blocking browser exploits is to explicitly allow only good code, your authorized applications, to run. When you specify which versions of which plug-ins, scripts, ActiveX controls, Java applets, and other web components your browser can run, you reduce the variables available for exploit. You can also tune configurations: you might block RealPlayer or DirectShow when you hear about this new exploit, at least until a patch or signature is available.

7. <http://www.avertlabs.com/research/blog/index.php/2009/07/06/new-attacks-against-internet-explorer/>

8. http://www.theregister.co.uk/2009/07/09/microsoft_security_delayed/

9. <http://www.trustedsource.org/blog/265/An-Artemis-View-of-Zero-Day-Attacks>

White Paper A Tale of Three Bogeys: Zero-day Vulnerabilities,
Unpatched Client Applications, and Browser Exploits

Conclusion

The web and email are not IT's friends. They create risk and increase your workload, because they distribute and prolong the payoff of known and unknown threats in zero-day vulnerabilities, client-side vulnerabilities, and Web 2.0 exploits. Layer the new and traditional techniques discussed here, including anti-malware, anti-spam, intrusion prevention, and application whitelisting, throughout your network and systems. You will minimize your risk and minimize your work: the time and cost of cleaning up after bogeys like these.

McAfee Can Help

McAfee has a complete, integrated portfolio of products to help you protect your systems, data, and networked resources.

To improve your systems security today, investigate McAfee Total Protection for Endpoint, which protects you against the known and the unknown using signatures, behavior, and global threat intelligence delivered in real time.

It combats cost and complexity by unifying anti-malware, anti-spam, host intrusion prevention, system firewall, web security, and other important protections within a single agent and single management console.

Consider McAfee Application Control to add application whitelisting for your critical systems, legacy environments, and fixed-function devices including point-of-sale (POS), kiosk, and mobile systems.

Learn more at
www.mcafee.com.

