Google fixes Android flaws that allow code execution with high system rights

reader comments

45 with 35 posters participating

Share this story

Google has shipped security patches for dozens of vulnerabilities in its Android mobile operating system, two of which could allow hackers to remotely execute malicious code with extremely high system rights.

In some cases, the malware could run with highly elevated privileges, a possibility that raises the severity of the bugs. That’s because the bugs, located in the Android System component, could enable a specially crafted transmission to execute arbitrary code within the context of a privileged process. In all, Google released patches for at least 34 security flaws, although some of the vulnerabilities were present only in devices available from manufacturer Qualcomm.

Anyone with a mobile device should check to see if fixes are available for their device. Methods differ by device model, but one common method involves either checking the notification screen or clicking Settings > Security > Security update. Unfortunately, patches aren’t available for many devices.

Two vulnerabilities ranked as critical in Google’s June security bulletin are indexed as CVE-2020-0117 and

CVE-2020-8597. They’re among four System flaws located in the Android system (the other two are ranked with a severity of high). The critical vulnerabilities reside in Android versions 8 through the most recent release of 11.

“These vulnerabilities could be exploited through multiple methods such as email, web browsing, and MMS when processing media files,” an advisory from the Department of Homeland Security-funded Multi-State-Information Sharing and Analysis Center said. “Depending on the privileges associated with the application, an attacker could then install programs; view, change, or delete data; or create new accounts with full user rights.”

Vulnerabilities with a severity rating of high affected the Android media framework, the Android framework, and the Android kernel. Other vulnerabilities were contained in components shipped in devices from Qualcomm. The two Qualcomm-specific critical flaws reside in closed source components. The severity of other Qualcomm flaws were rated as high.

https://arstechnica.com/?p=1680681

Apple fixes bug that could have given hackers full access to user accounts

reader comments

23 with 20 posters participating

Share this story

Sign in with Apple—a privacy-enhancing tool that lets users log into third-party apps without revealing their email addresses—just fixed a bug that made it possible for attackers to gain unauthorized access to those same accounts.

“In the month of April, I found a zero-day in Sign in with Apple that affected third-party applications which were using it and didn’t implement their own additional security measures,” app developer Bhavuk Jain wrote on Sunday. “This bug could have resulted in a full account takeover of user accounts on that third party application irrespective of a victim having a valid Apple ID or not.”

Jain privately reported the flaw to Apple under the company’s bug bounty program and received a hefty $100,000 payout. The developer shared details after Apple updated the sign-in service to patch the vulnerability.

Sign in with Apple debuted in October as an easier and more secure and private way to sign into apps and websites. Faced with a mandate that many third-party iOS and iPadOS apps offer the option to sign in with Apple, a host of high-profile services entrusted with huge amounts of sensitive user data use adopted it.

Instead of using a social media account or email address, filling out Web forms, and choosing an account-specific password, iPhone and iPad users can tap an button and sign in with Face ID, Touch ID, or a device passcode. The bug opened users to the possibility their third-party accounts would be completely hijacked.

The sign-in service, which works similarly to the OAuth 2.0

standard, logs in users by using either a JWT—short for JSON Web Token—or a code generated by an Apple server. In the latter case, the code is then used to generate a JWT. Apple gives users the option of sharing the Apple email ID with the third party or keeping the ID hidden. When users hide the ID, Apple creates a JWT that contains a user-specific relay ID.

“I found I could request JWTs for any Email ID from Apple and when the signature of these tokens was verified using Apple’s public key, they showed as valid,” Jain wrote. “This means an attacker could forge a JWT by linking any Email ID to it and gaining access to the victim’s account.”

There’s no indication the bug was ever actively exploited.

https://arstechnica.com/?p=1680535

Attackers exploit 0-day code-execution flaw in the Sophos firewall

reader comments

43 with 31 posters participating

Share this story

Users of a widely used firewall from Sophos have been under a zero-day attack that was designed to steal usernames,

cryptographically protected passwords, and other sensitive data, officials with the security firm said on Sunday.

The well-researched and developed attack exploited a SQL injection flaw in fully patched versions of the Sophos XG Firewall. With that toehold in systems, it downloaded and installed a series of scripts that ultimately executed code intended to make off with users' names, usernames, the cryptographically hashed form of the passwords, and the salted SHA256 hash of the administrator account's password. Sophos has delivered a hotfix that mitigates the vulnerability.

Other data targeted by the attack included a list of the IP address allocation permissions for firewall users; the version of the custom operating system running; the type of CPU; the amount of memory that was present on the device; how long it had been running since the last reboot; the output of the ifconfig, a command-line tool; and ARP tables used to translate IP addresses into domain names.

"This malware's primary task appeared to be data theft, which it could perform by retrieving the contents of various database tables stored in the firewall, as well as by running some operating system commands," Sophos researchers wrote in Sunday's disclosure. "At each step, the malware collected information and then concatenated it to a file it stored temporarily on the firewall with the name Info.xg."

The exploits also downloaded the malware from domains that appeared to be legitimate. To evade detection, some of the malware deleted underlying files that executed it and ran solely in memory. The malicious code uses a creative and roundabout method to ensure it's executed each time firewalls are started. Those characteristics strongly suggest that the threat actors spent weeks or months laying the groundwork for the attacks.

Sophistication

The attack demonstrated that the attackers had a detailed knowledge of the Firewall that could only come from someone who had access to the software, which likely required a license. From there, the attackers carefully studied the Firewall to find inner workings that allowed the downloading and installation of malware that used names that closely resembled names of legitimate files and processes.

The data the malware was designed to exfiltrate suggests the attack was designed to give attackers the means to further penetrate the organizations that used the firewall through phishing attacks and unauthorized access to user accounts, and it potentially exploits targeting the firewalls or end users. The Sophos post said there was no evidence the data exfiltrations were successful, but it also didn’t rule out that possibility.

The zero-day vulnerability that made the attacks possible was a pre-authentication SQL injection flaw found in the custom operating system that runs the firewall. Sophos provided no additional details about the vulnerability. SQL injection exploit flaws that execute malicious code through strings that are entered into forms contained on a vulnerable website. The flaws are the result of a failure to filter out commands. Pre-authentication means the attacker didn’t need to provide any credentials to carry execute code.

Users of vulnerable firewalls should install the hotfix as soon as possible and then examine their systems for signs of compromise published on the previously mentioned post here.

https://arstechnica.com/?p=1671396

That no-click iOS 0-day reported to be under exploit doesn’t exist, Apple says

reader comments

37 with 30 posters participating

Share this story

Apple is disputing the accuracy of this week’s report that found attackers have been exploiting an unpatched iOS bug that allowed them to take full control of iPhones.

San Francisco-based security firm ZecOps said on Wednesday that attackers had used the zero-day exploit against at least six targets over a span of at least two years. In the now-disputed report, ZecOps had said the critical flaw was located in the Mail app and could be triggered be sending specially manipulated emails that required no interaction on the part of users.

Apple declined to comment on the report at the time. Late on Thursday night, however, Apple pushed back on ZecOps’ findings that (a) the bug posed a threat to iPhone and iPad users and (b) there had been any active exploit at all. In a statement, officials wrote:

> *Apple takes all reports of security threats seriously. We have thoroughly investigated the researcher’s report and, based on the information provided, have concluded these issues do not pose an immediate risk to our users. The researcher identified three issues in Mail, but alone they*

> *are insufficient to bypass iPhone and iPad security protections, and we have found no evidence they were used against customers. These potential issues will be addressed in a software update soon. We value our collaboration with security researchers to help keep our users safe and will be crediting the researcher for their assistance.*

A fair number of independent researchers have also questioned the ZecOps conclusion. Generally, the critics said that the evidence ZecOps based its findings on wasn't persuasive. The disputed findings were based on evidence that the malicious emails were deleted, presumably to hide attacks, but that data that remained in logs indicated the deletions and crashes were the result of an exploit.

The critics said if the exploit was able to delete the emails ,it would have been able to delete the crash log data as well. The critics said that failure and some technical details contained in the ZecOps report strongly suggested the flaw was a more benign bug that was triggered by certain types of emails. Also skeptical, the critics said, is that an advanced exploit would cause a crash at all. Those doubts have continued ever since.

HD Moore, vice president of research and development at Atredis Partners and an expert in software exploitation, told me on Friday:

> *It looks like ZecOps identified a crash report, found a way to reproduce the crashes, and based on circumstantial evidence assumed this was being used for malicious purposes. It sounds like after he reported it to Apple, Apple investigated, found out these were just crash bugs, and that shuts the door on this being actually in-the-wild-exploitation of a new iOS zero-day.*
>
> *It could be Apple is wrong, but given their sensitivity to this stuff, they probably did a decent job of investigating*

> *it. Through the grapevine I heard that the internal security team that handled this investigation at Apple was pissed off about it, since ZecOps went straight to press before they had a chance to review.*

Other critics have delivered their critiques on Twitter.

"Looks like you have a real vuln but the evidence of exploitation looks weak… and no info in your post on post-exploitation chaining to lead to info disclosure or code execution," researcher Rich Mogul wrote. "Any update you can share? Pretty big claim of a no-click mail 0-day being used."

> *Looks like you have a real vuln but the evidence of exploitation looks weak… and no info in your post on post-exploitation chaining to lead to info disclosure or code execution. Any update you can share? Pretty big claim of a no-click mail 0-day being used. https://t.co/xrWbXTPndQ*
>
> *— Rich Mogull (@rmogull) April 22, 2020*

While Mogul left open the possibility of a real-world exploitation of a vulnerability, he said ZecOps didn't provide adequate proof to rule out an intentional bug crash. Another criticism is here.

ZecOps, meanwhile, appeared to stand by its report, saying on Twitter:

> *According to ZecOps data, there were triggers in-the-wild for this vulnerability on a few organizations. We want to thank Apple for working on a patch, and we're looking forward to updating our devices once it's available. ZecOps will release more information and POCs once a patch is available.*

ZecOps said that based on the data collected on iPhones it believes were exploited, company researchers were able to

write a proof-of-concept exploit that took full control of fully updated devices. ZecOps has declined to publish the exploit or other data until Apple releases a fix for the bug. Apple has already released the patch for a beta version of the upcoming 13.4.5, and as Thursday night's statement said, the company plans make it generally available soon.

The controversy, Apple's denial, and the rarity of zero-click vulnerabilities in iOS are certainly reasons for skepticism. It will be worth reviewing the additional information ZecOps has pledged to publish once Apple releases a fix.

https://arstechnica.com/?p=1671007

A critical iPhone and iPad bug that lurked for 8 years may be under active attack

reader comments

36 with 22 posters participating

Share this story

A critical bug that has lurked in iPhones and iPads for eight years appears to be under active attack by sophisticated hackers to hack the devices of high-profile targets, a security firm reported on Wednesday.

The exploit is triggered by sending booby-trapped emails that, in some cases, require no interaction at all and, in other cases, require only that a user open the message, researchers from ZecOps said in a post. The malicious emails allow attackers to run code in the context of the default mail apps, which make it possible to read, modify, or delete messages. The researchers suspect the attackers are combining the zero-day with a separate exploit that gives full control over the device. The vulnerability dates back to iOS 6 released in 2012. Attackers have been exploiting the bug since 2018 and possibly earlier.

Enormous scope

“With very limited data we were able to see that at least six organizations were impacted by this vulnerability– and the full scope of abuse of this vulnerability is enormous,” ZecOps researchers wrote. “We are confident that a patch must be provided for such issues with public triggers ASAP.”

Targets from the six organizations include:

- Individuals from a Fortune 500 organization in North America
- An executive from a carrier in Japan
- A VIP from Germany
- Managed security services providers in Saudi Arabia and Israel
- A journalist in Europe
- Suspected: An executive from a Swiss enterprise

Zerodays, or vulnerabilities that are known to attackers but not the manufacturer or the general public, are rarely exploited in the wild against against users of iPhones and iPads. Some of the only known incidents a 2016 attack that installed spyware on the phone of a dissident in the United Arab Emirates, a WhatsApp exploit in May of last year that was transmitted with a simple phone call, and attacks that Google

disclosed last August.

Apple has currently patched the flaw in the beta for iOS 13.4.5. At the time this post went live, a fix in the general release had not yet been released.

Malicious mails that trigger the flaw work by consuming device memory and then exploiting a heap overflow, which is a type of buffer overflow that exploits an allocation flaw in memory reserved for dynamic operations. By filling the heap with junk data, the exploit is able to inject malicious code that then gets executed. The code triggers strings that include 4141…41, which are commonly used by exploit developers. The researchers believe the exploit then deletes the mail.

A protection known as address space layout randomization prevents attackers from knowing the memory location of this code and thus executing in a way that takes control of the device. As a result, the device or application merely crashes. To overcome this security measure, attackers must exploit a separate bug that reveals the hidden memory location.

Little or no sign of attack

The malicious mails need not be prohibitively large. Normal-size emails can consume enough RAM using rich text format documents, multi-part content, or other methods. Other than a temporary device slowdown, targets running iOS 13 aren’t likely to notice any signs that they’re under attack. In the event that the exploit fails on a device running iOS 12, meanwhile, the device will show a message that says “This message has no content.”

ZecOps said the attacks are narrowly targeted but provided only limited clues about the hackers carrying them out or targets who were on the receiving end.

“We believe that these attacks are correlative with at least

one nation-state threat operator or a nation-state that purchased the exploit from a third-party researcher in a Proof of Concept (POC) grade and used ‘as-is’ or with minor modifications (hence the 4141..41 strings),” ZecOps researchers wrote. “While ZecOps refrain from attributing these attacks to a specific threat actor, we are aware that at least one ‘hackers-for-hire’ organization is selling exploits using vulnerabilities that leverage email addresses as a main identifier.”

The most visible third-party organization selling advanced smartphone exploits is Israel-based NSO Group, whose iOS and Android exploits over the past year have been found being used against activists, Facebook users, and undisclosed targets. NSO Group has come under sharp criticism for selling its wares in countries with poor human-rights records. In recent months, the company has vowed to serve only organizations with better track records.

It’s generally against security community norms to disclose vulnerabilities without giving manufacturers time to release security patches. ZecOps said it released its research ahead of a general release fix because the zeroday alone isn’t enough to infect phones, the bugs had already been mentioned in the beta release, and the urgency created by the six organizations the firm believes are under active attack

To prevent attacks until Apple releases a general-availability patch, users can either install the beta 13.4.5 or use an alternate email app such as Gmail or Outlook. Apple representatives didn’t respond to an email seeking comment for this post.

https://arstechnica.com/?p=1670281

Microsoft patches 3 Windows 0days under active exploit

Enlarge / A man looks at the home screen for the “new” Windows 7 platform when it was launched in October 2009. Microsoft has ended support, but the OS lives on.

reader comments

7 with 6 posters participating

Share this story

Microsoft has patched three actively exploited vulnerabilities that allow attackers to execute malicious code or elevate system privileges on devices that run Windows.

Two of the security flaws—tracked as CVE-2020-1020 and CVE-2020-0938—reside in the Adobe Type Manager Library, a Windows DLL file that a wide variety of apps use to manage and render fonts available from Adobe Systems. On supported operating systems other than Windows 10, attackers who successfully exploit the vulnerabilities can remotely execute code. On Windows 10, attackers can run code inside an AppContainer sandbox. The measure limits the system privileges malicious code has, but even then, attackers can use it to create accounts with full user rights, install programs, and view, change, or delete data.

Attackers can exploit the flaws by convincing a target to open a booby-trapped document or viewing it in the Windows preview pane. Tuesday’s advisories said that Microsoft is “aware of limited, targeted attacks that attempt to leverage” both vulnerabilities. Microsoft revealed last month that one of the

bugs was being exploited in limited attacks against Windows 7 machines.

While installing the newly available patches is the best way to protect vulnerable systems, temporary workarounds for those who need to buy more time include:

- Disabling the Preview Pane and Details Pane in Windows Explorer
- Disabling the WebClient service
- Rename ATMFD.DLL (on Windows 10 systems that have a file by that name), or alternatively, disable the file from the registry

These are the same mitigations that Microsoft recommended in its March advisory. Once the patches are installed, users can roll back the mitigations.

The last zeroday exploit targets CVE-2020-1027, an elevation of privilege flaw in the way that the Windows kernel handles objects in memory. Attackers who already have limited system rights on a vulnerable machine can use the exploit to execute malicious code. To exploit the vulnerability, a locally authenticated attacker could run a specially crafted application.

Microsoft didn’t provide any details about the attacks that are underway against the latter two flaws.

Threat analysis group gets credit

The software maker credited discovery of the zero-day exploits to Google’s threat analysis group, which tracks government-backed hack attacks against the company’s users.

Google’s threat analysis group reported the attacks against the Adobe Type Manager flaws on March 23 and, per the company’s disclosure policy for actively exploited bugs, gave Microsoft seven days to fix or disclose the flaw. Google later

gave Microsoft an extension to accommodate work slowdowns caused by the novel coronavirus pandemic. Group members plan to issue a report that details the Adobe flaws in the next month or so. It’s not clear when the threat analysis group will provide details about the other two vulnerabilities.

Typically, Windows devices in home and smaller-office settings receive patches automatically within 24 hours. It’s always a good idea to make sure updates are installed within that time frame. Administrators in larger organizations face the sometimes-difficult task of testing patches before deploying them to ensure they’re compatible with systems already in place. That task is likely to be harder this month, with the work disruptions caused by COVID-19 infections sweeping the globe.

Post updated to correct the number of 0days. It’s three.

https://arstechnica.com/?p=1668426

Attackers can use Zoom to steal users’ Windows credentials with no warning

reader comments

93 with 67 posters participating, including story author

Share this story

Users of Zoom for Windows beware: the widely used software has a vulnerability that allows attackers to steal your operating system credentials, researchers said.

Discovery of the currently unpatched vulnerability comes as Zoom usage has soared in the wake of the coronavirus pandemic. With massive numbers of people working from home, they rely on Zoom to connect with co-workers, customers, and partners. Many of these home users are connecting to sensitive work networks through temporary or improvised means that don't have the benefit of enterprise-grade firewalls found on-premises.

Embed network location here

Attacks work by using the Zoom chat window to send targets a string of text that represents the network location on the Windows device they're using. The Zoom app for Windows automatically converts these so-called universal naming convention strings—such as \\attacker.example.com/C$—into clickable links. In the event that targets click on those links on networks that aren't fully locked down, Zoom will send the Windows usernames and the corresponding NTLM hashes to the address contained in the link.

Attackers can then use the credentials to access shared network resources, such as Outlook servers and storage devices. Typically, resources on a Windows network will accept the NTLM hash when authenticating a device. That leaves the networks open to so-called pass-the-hash attacks that don't require a cracking technique to convert the hash to its corresponding plain-text password.

"It's quite a shortcoming from Zoom," Matthew Hickey, cofounder of the security boutique Hacker House, told me. "It's a very trivial bug. With more of us working from home now, it's even easier to exploit that bug."

More than 24 hours after receiving a request to comment and nine hours after this post went live, a Zoom representative emailed with the follow statement: “At Zoom, ensuring the privacy and security of our users and their data is paramount. We are aware of the UNC issue and are working to address it.”

The vulnerability was first described last week by a researcher who uses the Twitter handle @_g0dmode. He wrote: “#Zoom chat allows you to post links such as \\x.x.x.x\xyz to attempt to capture Net-NTLM hashes if clicked by other users.

> *#Zoom chat allows you to post links such as \\x.x.x.x\xyz to attempt to capture Net-NTLM hashes if clicked by other users.*
>
> *— Mitch (@_g0dmode) March 23, 2020*

On Tuesday, Hickey expanded on the discovery. He showed in one tweet how the Zoom Windows client exposed the credentials that could be used to access restricted parts of a Windows network.

“Hi @zoom_us & @NCSC,” Hickey wrote. “Here is an example of exploiting the Zoom Windows client using UNC path injection to expose credentials for use in SMBRelay attacks. The screen shot below shows an example UNC path link and the credentials being exposed (redacted).”

The screenshot shows the Windows username as Bluemoon/HackerFantastic. Immediately below, the NTLM hash appears, although Hickey redacted most of it in the image he posted.

> *Hi @zoom_us & @NCSC — here is an example of exploiting the Zoom Windows client using UNC path injection to expose credentials for use in SMBRelay attacks. The screen shot below shows an example UNC path link and the credentials being exposed (redacted). pic.twitter.com/gjWXas7TMO*
>
> *— Hacker Fantastic (@hackerfantastic) March 31, 2020*

Attacks can be mounted by people posing as a legitimate meeting participant or during so-called Zoom bombing raids, in which trolls access a meeting not secured by a password and bombard everyone else with offensive or harassing images.

Protect yourself

While the attack works only against Windows users, Hickey said attacks can be launched using any form of Zoom, again, by sending targets a UNC location in a text message. When Windows users click on the link while they're connected to certain unsecured machines or networks, the Zoom app will send the credentials over port 445, which is used to transmit traffic related to Windows SMB and Active Directory services.

In the event that port 445 is closed to the Internet—either by a device or network firewall or through an ISP that blocks it—the attack won't work. But it's hardly a given that this egress will be closed on many Zoom users' networks. The events of the past month have left millions of people working from home without the same levels of IT and security support they get when working on premises. That makes it more likely that port 445 is open, either because of an oversight or because the port is needed to connect to enterprise resources.

Zoom's statement didn't indicate when a fix will be in place. Until then, Windows users should be extra suspicious of chat messages that contain links. When possible, users should also ensure that port 445 is either blocked or can access only trusted addresses on the Internet.

https://arstechnica.com/?p=1664925

Researchers find serious flaws in WordPress plugins used on 400k sites

reader comments

49 with 33 posters participating

Share this story

Serious vulnerabilities have recently come to light in three WordPress plugins that have been installed on a combined 400,000 websites, researchers said. InfiniteWP, WP Time Capsule, and WP Database Reset are all affected.

The highest-impact flaw is an authentication bypass vulnerability in the InfiniteWP Client, a plugin installed on more than 300,000 websites. It allows administrators to manage multiple websites from a single server. The flaw lets anyone log in to an administrative account with no credentials at all. From there, attackers can delete contents, add new accounts, and carry out a wide range of other malicious tasks.

People exploiting the vulnerability need only know the user name of a valid account and include a malicious payload in a POST request that’s sent to a vulnerable site. According to Web application firewall provider Wordfence, the vulnerability stems from a feature that allows legitimate users to automatically log in as an administrator without providing a password.

“Logical vulnerabilities like the ones seen in this recent disclosure can result in severe issues for Web applications

and components," Marc-Alexandre Montpas, a researcher at Web security firm Sucuri, wrote in a post. "These flaws can be exploited to bypass authentication controls—and in this case, log in to an administrator account without a password."

Anyone running InfiniteWP Client version 1.9.4.4 or earlier should update to 1.9.4.5 immediately.

The critical flaw in WP Time Capsule also leads to an authentication bypass that allows unauthenticated attackers to log in as an administrator. WP Time Capsule, which runs on about 20,000 sites, is designed to make backing up website data easier. By including a string in a POST request, attackers can obtain a list of all administrative accounts and automatically log in to the first one. The bug has been fixed in version 1.21.16. Sites running earlier versions should update right away. Web security firm WebARX has more details.

The last vulnerable plugin is WP Database Reset, which is installed on about 80,000 sites. One flaw allows any unauthenticated person to reset any table in the database to its original WordPress state. The bug is caused by reset functions that aren't secured by the standard capability checks or security nonces. Exploits can result in the complete loss of data or a site reset to the default WordPress settings.

A second security flaw in WP Database Reset causes a privilege-escalation vulnerability that allows any authenticated user—even those with minimal system rights—to gain administrative rights and lock out all other users. All site administrators using this plugin should update to version 3.15, which patches both vulnerabilities. Wordfence has more details about both flaws here.

There's no evidence that any of the three vulnerable plugins are being actively exploited in the wild.

https://arstechnica.com/?p=1645061

Critical Windows 10 vulnerability used to Rickroll the NSA and Github

Enlarge / Chrome on Windows 10 as it Rickrolls the NSA.

reader comments

96 with 64 posters participating, including story author

Share this story

Less than a day after Microsoft disclosed one of the most critical Windows vulnerabilities ever, a security researcher has demonstrated how attackers can exploit it to cryptographically impersonate any website or server on the Internet.

Researcher Saleem Rashid on Wednesday tweeted images of the video “Never Gonna Give You Up,” by 1980s heart-throb Rick Astley, playing on Github.com and NSA.gov. The digital sleight of hand is known as Rickrolling and is often used as a humorous and benign way to demonstrate serious security flaws. In this case, Rashid’s exploit causes both the Edge and Chrome browsers to spoof the HTTPS verified websites of Github and the National Security Agency. Brave and other Chrome derivatives, as well as Internet Explorer, are also likely to fall to the same trick. (There’s no indication Firefox is affected.)

Enlarge / The same exploit used to Rickroll Github on Edge.
Rashid's simulated attack exploits CVE-2020-0601, the critical vulnerability that Microsoft patched on Tuesday after receiving a private tipoff from the NSA. As Ars reported, the flaw can completely break certificate validation for websites, software updates, VPNs, and other security-critical computer uses. It affects Windows 10 systems, including server versions Windows Server 2016 and Windows Server 2019. Other versions of Windows are unaffected.

Rashid told me his exploit uses about 100 lines of code but that he could compress it down to 10 lines if he wanted to remove a "few useful tricks" his attack has. While there are constraints and several potentially difficult requirements in getting the exploit to work in real-world, adversarial conditions (more about that later), Wednesday's proof-of-concept attack demonstrates why the NSA assesses the vulnerability as "severe" and said sophisticated hackers could understand how to exploit it "quickly."

"Fairly terrifying"

Other researchers shared the NSA's sense of urgency.

"What Saleem just demonstrated is: with [a short] script you can generate a cert for any website, and it's fully trusted on IE and Edge with just the default settings for Windows," Kenn White, a researcher and security principal at MongoDB, said. "That's fairly horrifying. It affects VPN gateways, VoIP,

basically anything that uses network communications.” (I spoke with White before Rashid had demonstrated the attack against Chrome.)

The flaw involves the way the new versions of Windows check the validity of certificates that use elliptic-curve cryptography. While the vulnerable Windows versions check three ECC parameters, they fail to verify a fourth, crucial one, which is known as a base point generator and is often represented in algorithms as ‘G.’ This failure is a result of Microsoft’s implementation of ECC rather than any flaw or weakness in the ECC algorithms themselves.

Attackers can exploit the flaw by extracting the public key of a root certificate that ships by default in Windows. These certificates are described as root because they belong to big certificate authorities that either issue their own TLS certificates or validate intermediate certificate authorities that sell certificates on the root CA’s behalf. Any root certificate will work, as long as it’s signed with an ECC algorithm. Rashid’s attack started with a root certificate from Sectigo, the Internet’s biggest CA, which previously used the name Comodo. The researcher later modified his attack to use a GlobalSign root certificate. His code made the switch automatic.

The attacker examines the specific ECC algorithm used to generate the root-certificate public key and proceeds to craft a private key that copies all of the certificate parameters for that algorithm except for the point generator. Because vulnerable Windows versions fail to check that parameter, they accept the private key as valid. With that, the attacker has spoofed a Windows-trusted root certificate that can be used to mint any individual certificate used for authentication of websites, software, and other sensitive properties.

The behavior is tantamount to a law enforcement officer who checks someone’s ID to make sure it properly describes the

person’s height, address, birthday, and face but fails to notice that the weight is listed as 250 pounds when the person clearly weighs less than half that.

“It’s such a strange bug, because it’s like they’re only halfway checking something that is at the root of the entire trust system,” White said. “It’s a core part of the whole chain of trust.”

For more detailed technical explanations of the bug, see posts here and here and the Twitter thread here.

The caveats

As noted earlier, there are several requirements and constraints that significantly raise the bar for Rashid’s attack to work in real-world uses by an adversary. The first is that it most likely requires an active man-in-the-middle attack. These types of attacks, which modify data as it passes through networks, may be difficult to carry out. An alternative to an active MitM is to convince a target to click on a fake URL. This method is much easier, but it also requires some targeting. (It wouldn’t apply to attacks against websites or other servers that require a certificate from the connecting client.)

The exploit also requires that the target has recently visited a site with a transport layer security certificate that’s chained to an ECC-signed root certificate. That’s because the root certificate must already be cached by the targeted system. In the event a targeted system doesn’t have the root certificate cached, Rashid said, an attacker could still pull off an exploit by adding JavaScript that accesses a site chained to the root certificate.

Another constraint: Chrome uses a mechanism known as certificate pinning for google.com and a variety of other sensitive websites. Pinning requires that the certificate

authenticating a website contain a specific cryptographic hash, even if the certificate offered is otherwise valid. This measure would prevent exploits from working when they spoofed protected sites.

While installing Tuesday's patch by Microsoft is by far the only reasonable way to prevent attacks, a Google representative said Chrome developers have already distributed a fix in a beta version and will fold the fix into stable versions soon. A word of caution: even with this fix, users of vulnerable Windows versions will still face considerable risk from other attack scenarios.

A matter of time

Despite the requirements and limitations, the vulnerability is serious. As NSA officials put it in the above-linked advisory:

> *The vulnerability places Windows endpoints at risk to a broad range of exploitation vectors. NSA assesses the vulnerability to be severe and that sophisticated cyber actors will understand the underlying flaw very quickly and, if exploited, would render the previously mentioned platforms as fundamentally vulnerable. The consequences of not patching the vulnerability are severe and widespread. Remote exploitation tools will likely be made quickly and widely available. Rapid adoption of the patch is the only known mitigation at this time and should be the primary focus for all network owners.*

The vulnerability may not pose as extreme a threat as those caused by the Heartbleed flaw in 2014 that allowed attackers to steal private keys, passwords, and other highly sensitive data from hundreds of thousands of vulnerable sites. But because of the breadth of security measures foiled by the Microsoft vulnerability, it's worse even than Apple's critical goto fail flaw, which prevented iOS and macOS systems from

detecting invalid TLS certificates served by websites. That makes CVE-2020-0601 one of the most severe vulnerabilities in recent memory.

Windows' automatic update mechanism is likely to have patched vulnerable systems already. For anyone else, fixes for various vulnerable versions are available here. Readers who haven't patched yet should do so immediately.

Post updated to correct description of certificate pinning and discussion of man-in-the-middle attacks.

https://arstechnica.com/?p=1644337

Firefox gets patch for critical zeroday that's being actively exploited

reader comments

31 with 29 posters participating, including story author

Share this story

Mozilla has released a new version of Firefox that fixes an actively exploited zeroday that could allow attackers to take control of users' computers.

In an advisory, Mozilla rated the vulnerability critical and said it was "aware of targeted attacks in the wild abusing

this flaw.” The US Cybersecurity and Infrastructure Security Agency said one or more exploits were “detected in the wild” and warned that attacks could be exploited to “take control of an affected system.” The Mozilla advisory credited researchers at China-based Qihoo 360 with reporting the flaw.

No other details about the attacks were immediately available. Neither Mozilla nor Qihoo 360 responded immediately to emails asking for more information.

CVE-2019-17026, as the vulnerability is indexed, is a type confusion, a potentially critical error that can result in data being written to, or read from, memory locations that are normally off-limits. These out-of-bounds reads may allow attackers to discover memory locations where malicious code is stored, so that protections such as address space layout randomization can be bypassed. Out-of-bounds reads can also cause crashes.

The flaw is fixed in Tuesday’s release of Firefox 72.0.1. The patch came a day after version 72 fixed 11 other vulnerabilities, six of which were rated high. Three of those six bugs might make it possible for attackers to run malicious code on affected computers.

The patching of CVE-2019-17026 comes seven months after Mozilla patched a pair of potent zerodays that attackers exploited in an attempt to install an undetected backdoor on Macs used by cryptocurrency exchange Coinbase.

While details of the new exploits are unavailable, Firefox users should install the patch as soon as practical. The easiest way to do that is use the in-browser update feature, which is available by clicking “About Firefox.” In Windows, it’s available in the menu’s Help section. On Macs, it’s in the menu’s Firefox section.

https://arstechnica.com/?p=1641253