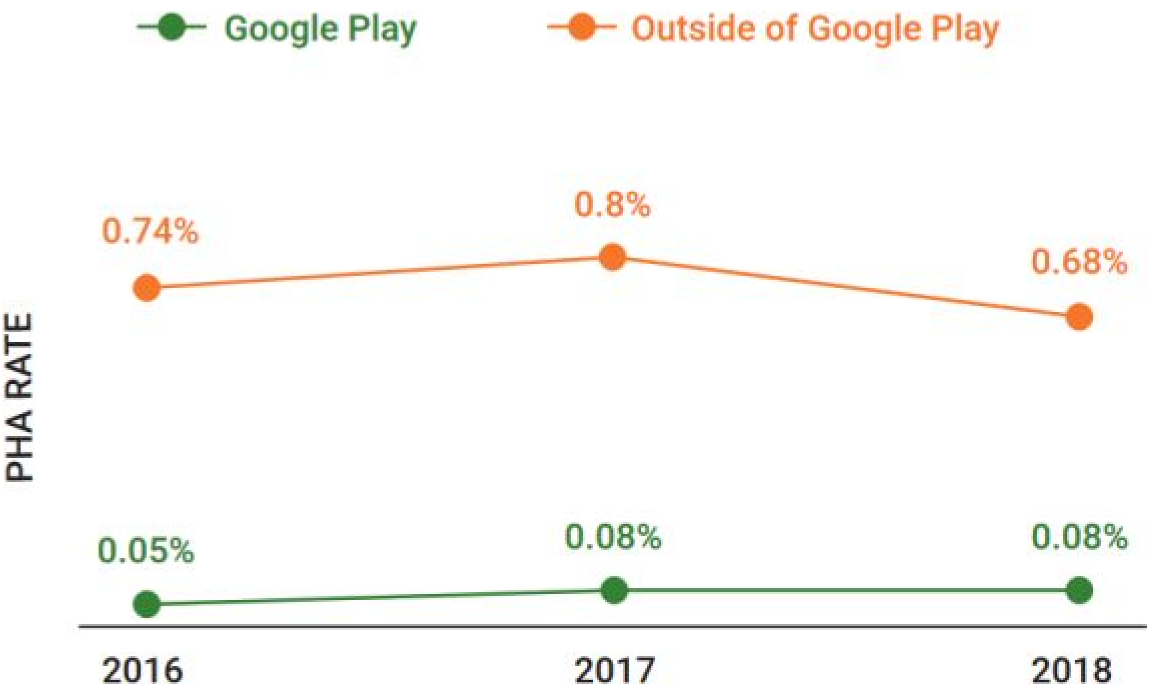
Security Now! #708 - 04-02-19
Android Security

This week on Security Now!

This week we are primarily going to share Google's well-deserved self-congratulatory, but also very honest update on the status of Android Security at its 10th birthday. But before that we're going to share some of the continuing news of the WinRAR vulnerability, some really interesting data on Russian GPS hacking, Android's April Fools Day patches, Tesla autopilot spoofing, some follow-up on the ASUS “ShadowHammer” attack and the targeted MAC addresses, the final release of the Windows 10 (last) October 2018 update, a VMware update, a SQRL question, two bits of listener feedback, a SpinRite development question... then we take a look at the state of Android 10 years in.



Percentage of devices with **Potentially Harmful Applications** installed, 2016-2018

Security News

FireEye weighs in on the WinRAR vulnerability

https://www.fireeye.com/blog/threat-research/2019/03/winrar-zero-day-abused-in-multiple-campaigns.html

Title: "WinRAR Zero-day Abused in Multiple Campaigns"
I'm unsure that this counts as a 0-day any longer, but it's interesting and educational to see how much focus and abuse an exploitable flaw in a widely used archive expansion flaw can obtain.

<quote> WinRAR, an over 20-year-old file archival utility used by over 500 million users worldwide, recently acknowledged a long-standing vulnerability in its code-base. A recently published path traversal zero-day vulnerability, disclosed in CVE-2018-20250 by Check Point Research, enables attackers to specify arbitrary destinations during file extraction of ‘ACE’ formatted files, regardless of user input. Attackers can easily achieve persistence and code execution by creating malicious archives that extract files to sensitive locations, like the Windows “Startup” Start Menu folder. While this vulnerability has been fixed in the latest version of WinRAR (5.70), WinRAR itself does not contain auto-update features, increasing the likelihood that many existing users remain running out-of-date versions.

FireEye has observed multiple campaigns leveraging this vulnerability, in addition to those already discussed by 360 Threat Intelligence Center. Below we will look into some campaigns we came across that used customized and interesting decoy documents with a variety of payloads including ones which we have not seen before and the ones that used off-the-shelf tools like PowerShell Empire. </quote>

Campaign 1: Impersonating an Educational Accreditation Council letter

When the ACE file Scan_Letter_of_Approval.rar is extracted with vulnerable WinRAR versions lower than 5.70, it creates a file named winSrvHost.vbs in the Windows Startup folder without the user’s consent. The VBScript file is executed the next time Windows starts up.

To avoid user suspicion, the ACE file contains a decoy document, “Letter of Approval.pdf”, which purports to be from CSWE, the Council on Social Work Education. This seems to be copied from CSWE website.

The VBS file in the Startup folder will be executed by wscript.exe when Windows starts up. The VBS code first derives an ID for the victim using custom logic based on a combination of the ComputerName, Processor_identifier and Username. It obtains these from environment strings. The malware reaches out to the C2 server at 185.162.131.92 via an HTTP request. The actual communication is via the HTTP query's Authorization header where the VBS backdoor places the base64-encoded data, including the victim ID and the ComputerName. The backdoor than extracts the base64-encoded data from the Authorization header of the HTTP response from the C2 server and decodes it. The decoded data starts with the instruction code from the C2 server, followed with additional parameters...

Upon decoding, the commands are found to be “ok ok”, which we believe is the default C2 command. After some C2 communication, the C2 server responded with instructions to download the payload from http://185.49.71.101/i/pwi_crs.exe, which is a Netwire RAT. (The Netwire RAT is a widely and popular Remote Access Trojan which was first observed in 2012.)

Commands Supported by VBS Backdoor:

- d - Delete the VBS file and exit process
- Pr - Download a file from a URL and execute it
- Hw - Retrieve hardware info
- av - Look for antivirus installed from a predefined list.

Campaign 2: Attack on Israeli Military Industry

Based on the email uploaded to VirusTotal, the attacker sends a spoofed email to the victim with an ACE file named SysAid-Documentation.rar as an attachment. Based on the VirusTotal uploader and the email headers, we believe this is an attack on an Israeli military company.

The ACE file contains a collection of decoy files related to documentation for SysAid, a help desk service based in Israel.

One of the files "Thumbs.db.lnk" is a valid windows shell link file pointing to 'C:\Users\john\Desktop\100m.bat'. But the icon for this link is remotely hosted on one of the C2 servers. This can be used to steal NT Lan Manager hashes

Upon extraction, the WinRAR flaw causes a previously unknown payload FireEye named "SappyCache" to be copied into the user's Startup folder with the file name 'ekrnview.exe'. The payload will be executed the next time Windows starts up.

SappyCache attempts to fetch the next-stage payload using three approaches:

1) Decrypting a File: The malware tries to read the file at %temp%\..\GuiCache.db. If it is successful, it tries to decrypt it using RC4 to get the C2 URLs.

2) Decrypting a Resource: If it is not successful in retrieving the C2 URL using the previous method, the malware next tries to retrieve the encrypted C2 URLs from an executable file's resource section. If it is successful, it will decrypt the C2 URLs using RC4.

3) Retrieving From C2: If it is not successful in retrieving the C2 URLs using those previous two methods, the malware tries to retrieve the payload from four different hardcoded URLs. The malware creates the HTTP request using the computer name from the "GetComputerName" Windows API call, the Windows OS name, retrieved by querying the ProductName value from the registry key SOFTWARE\Microsoft\Windows NT\CurrentVersion and uses it as the HTTP parameter 'key'. And the malware's module name retrieved using the GetModuleFileName API call, returning it as the HTTP parameter 'page'.

Then the system's entire list of running processes and their module names is retrieved with standard Windows process enumeration functions and placed into the HTTP parameter 'session_data'.

If any of the aforementioned methods is successful, the malware sends the query then attempts to execute the decrypted payload received in any reply. By the time FireEye was performing this analysis the C2 server did not respond with a next-level payload.

Campaign 3: Potential Attack in Ukraine with Empire Backdoor

The ACE file named zakon.rar is propagated using a malicious URL. The researchers at 360 Threat Intelligence Center had also encountered this malicious campaign.

The ACE file contains a file named Ukraine.pdf, which contains a message on the law of Ukraine about public-private partnerships that purports to be a message from the former president of Ukraine.

Based on the decoy PDF name, the decoy PDF content and the VirusTotal uploader, we believe this is an attack on a specific individual in Ukraine. Thus a classic targeted attack.

When the file's contents are extracted, WinRAR drops a .bat file named mssconf.bat in the Startup folder. The batch file contains commands that invoke base64-encoded PowerShell commands. After decoding, the PowerShell commands invoked are found to be the Empire backdoor. FireEye reported that they did not observe any additional payloads at the time of their analysis.

Campaign 4: Credential and Credit Card Dumps as Decoys

This campaign uses credential dumps and likely stolen credit card dumps as decoy documents to distribute different types of RATs and password stealers. (Apparently there really is no honor among thieves!)

One file named 'leaks copy.rar', used text files that contained stolen email IDs and passwords as decoys for an unwitting victim. Another file, 'cc.rar', used a text file containing stolen credit card information as a decoy.

This campaign used payloads from different malware families including one well known RAT named: QuasarRAT. In another case the decompilation of the .NET-based payload revealed that much of the code is written in Chinese. It was later identified as 'Buzy.' The other payloads have similar keylogging, password stealing and standard RAT capabilities. The VirusTotal submissions show the use of different malware families in this campaign and a wide range of targeting.

FireEye concludes their report and analysis by writing:

We have seen how various threat actors are abusing the recently disclosed WinRAR vulnerability using customized decoys and payloads, and by using different propagation techniques such as email and HTTP queries. Because of the huge WinRAR customer-base, lack of auto-update feature and the ease of exploitation of this vulnerability, we believe this will be used by more threat actors in the coming days.

Traditional AV solutions will have a hard time providing proactive zero-day detection for unknown malware families. It's also worth noting that this vulnerability allows the malicious ACE file to write a payload to any path if WinRAR has sufficient permissions, so although the exploits that we have seen so far chose to write the payload to startup folder, a more involved threat actor can come up with a different file path to achieve code execution so that any behavior based rules looking for WinRAR writing to the startup folder can be bypassed. Enterprises should consider blocking vulnerable WinRAR versions and mandate updating WinRAR to the latest version.

Win-Rar responds with an offer of MalwareBytes for free!

Julia D. Seymour <seymour@win-rar.com>
Subject: Update WinRAR and get Malwarebytes Premium FREE!

Dear Customer,

Greetings from WinRAR!

You may ask yourself why we are contacting you at this particular point in time. We wouldn't usually contact our users individually, but these are extraordinary circumstances

We have recently released the new version of WinRAR 5.70, following the discovery of a potential security vulnerability within the UNACEV2.DLL . For more information please check here

Here at win.rar GmbH, we believe in full transparency, which is why we are contacting our users personally to explain and offer advice regarding their continued safe use of WinRAR. We always recommend that users update to the latest version, to remain risk-free and to have full access to all of the current improvements, additions, and bug-fixes.

As a WinRAR license holder, we wanted to make sure that you are aware that you can upgrade to the latest version free-of-charge. You will find the latest version of WinRAR 5.70 in your desired language here: www.win-rar.com/download-winrar.html.

In addition to that, we are offering you Malwarebytes Premium for free*.

Download the most current version from here www.malwarebytes.com/mwb-download/thankyou and insert the license key by clicking the "Activate License" button in the top menu bar of the software.

Here is your Malwarebytes registration key: .....

WINRAR Upgrade:

Upgrading is quick and easy. No complicated uninstalling of the previous version is necessary. Do not delete your existing "WinRAR" program folder. Your registration information and WinRAR settings will be kept then. Close all open WinRAR archives and exit WinRAR before installing. Then you can just install the new version of WinRAR over your old installation by double-clicking on the .exe file you have downloaded. Now you are ready to continue using the best compression tool around!

We would also recommend that you sign up to receive our newsletter; keeping you up-to-date with all of the latest WinRAR developments and improvements. Please feel free to sign up here

If you have any questions regarding the new version of WinRAR, please do not hesitate to get in touch and we will be happy to help.

* Using the provided license key, you will receive Malwarebytes PREMIUM for free for a period of 3 months.

Russia has been messing with Global Positioning Systems
https://sophosnews.files.wordpress.com/2019/04/a8543-aboveusonlystars.pdf

"Above Us Only Stars" / Exposing GPS Spoofing in Russia and Syria

A 66-page detailed extremely compelling analysis of signals intelligence collected from fixed and in-orbit assets.

Executive Summary

GPS and other Global Navigation Satellite Systems (GNSS) are used in everything from cellular communication networks, to basic consumer goods, high-end military systems, and stock trading inputs. But these systems are vulnerable: by attacking positioning, navigational, and timing (PNT) data through electronic warfare (EW) capabilities, state and non-state actors can cause significant damage to modern militaries, major economies, and everyday consumers alike.[1][2][3] With recent technological advances, the tools and methodologies for conducting this interference are now at a high risk for proliferation. GNSS attacks are emerging as a viable, disruptive strategic threat.

In this report, we present findings from a year-long investigation ending in November 2018 on an emerging subset of EW activity: the ability to mimic, or “spoof,” legitimate GNSS signals in order to manipulate PNT data. Using publicly available data and commercial technologies, we detect and analyze patterns of GNSS spoofing in the Russian Federation, Crimea, and Syria that demonstrate the Russian Federation is growing a comparative advantage in the targeted use and development of GNSS spoofing capabilities to achieve tactical and strategic objectives at home and abroad. We profile different use cases of current Russian state activity to trace the activity back to basing locations and systems in use.

- In Section One, we examine GNSS spoofing events across the entire Russian Federation, its occupied territories, and overseas military facilities. We identify 9,883 suspected instances across 10 locations that affected 1,311 civilian vessel navigation systems since February 2016. We demonstrate that these activities are much larger in scope, more diverse in geography, and longer in duration than any public reporting suggests to date.

- In Section Two, we examine the role of Russian GNSS spoofing for very important person (VIP) protection. We find a close correlation between movements of the Russian head of state and GNSS spoofing events. We believe the Russian Federal Protective Service (FSO) operates mobile systems to support this activity. Through a review of Russian procurement data, we identify one possible mobile system, manufactured by a company closely connected to the FSO.

- In Section Three, we profile the use of Russian GNSS spoofing for strategic facilities protection. We identify potential technology in use for facility protection in Moscow. We also highlight spoofing activities taking place in proximity to protected facilities on the coast of Russia and Crimea in the Black Sea. Through a line of sight analysis, we judge the most likely placement for a GNSS spoofing transmitter on the Black Sea to be at a multi-million dollar “palace,” formerly owned by reported family members of senior FSO officers and previously reported to be built for President Putin.

- Finally, in Section Four, we expose the use of GPS spoofing in active Russian combat zones, particularly Syria, for airspace denial purposes. This is a capability scarcely reported in the public domain. Using data from a scientific sensor on the International Space Station (ISS), we are able to identify ongoing activity that poses significant threats to civilian airline GPS systems in the region. We pinpoint the most likely location for the system to the northwestern quadrant of Khmeimim airbase, and identify potential military-grade EW systems in use through publicly available information.

The Russian Federation has a comparative advantage in the targeted use and development of GNSS spoofing capabilities. However, the low cost, commercial availability, and ease of deployment of these technologies will empower not only states, but also insurgents, terrorists, and criminals in a wide range of destabilizing state-sponsored and non-state illicit networks. GNSS spoofing activities endanger everything from global navigational safety to civilian finance, logistics, and communication systems.

Android was patched on April Fools Day

The patches fix a pair of critical remote code execution (RCE) vulnerabilities and nine high severity elevation of privileges (EoP) and information disclosure (ID) vulnerabilities.

The security issues tracked as CVE-2019-2027 and CVE-2019-2028 as part of the 2019-04-01 security patch level are critical vulnerabilities impacting the Media framework which could allow potential remote attackers to make use of specially crafted files "to execute arbitrary code within the context of a privileged process."

CVE	Type	Severity	Updated AOSP versions
CVE-2019-2027	RCE	Critical	7.0, 7.1.1, 7.1.2, 8.0, 8.1, 9
CVE-2019-2028	RCE	Critical	7.0, 7.1.1, 7.1.2, 8.0, 8.1, 9

The two critical vulnerabilities impact all Android 7.0 or later devices, but users should be safe against attacks after applying the latest Android security patch.

According to Google, there were no "reports of active customer exploitation or abuse of any of these newly reported issues" and the severity assessment of the security issues patched in this month's security update are based on the effect their possible exploitation WOULD have on compromised devices.

Google also says that all Android partners were alerted of all issues disclosed in this update at least a month prior to today's public disclosure.

Tesla Autopilot Spoofing:
The attention-grabbing headline was: "Researchers Trick Tesla to Drive into Oncoming Traffic" Unfortunately, in this case the hack appears to have been quite simple to pull off.

The 40-page research paper was published by researchers at the Tencent Keen Security Lab. Their paper was titled: "Experimental Security Research of Tesla Autopilot"

https://keenlab.tencent.com/en/whitepapers/Experimental_Security_Research_of_Tesla_Autopilot.pdf

Abstract
Keen Security Lab has maintained the security research work on Tesla vehicle and shared our research results on Black Hat USA 2017[1] and 2018[2] in a row. Based on the ROOT privilege of the APE (Tesla Autopilot ECU, software version 18.6.1), we did some further interesting research work on this module. We analyzed the CAN messaging functions of APE, and successfully got remote control of the steering system in a contact-less way. We used an improved optimization algorithm to generate adversarial examples of the features (autowipers and lane recognition) which make decisions purely based on camera data, and successfully achieved the adversarial example attack in the physical world. In addition, we also found a potential high-risk design weakness of the lane recognition when the vehicle is in Autosteer mode. The whole article is divided into four parts: first a brief introduction of Autopilot, after that we will introduce how to send control commands from APE to control the steering system when the car is driving. In the last two sections, we will introduce the implementation details of the autowipers and lane recognition features, as well as our adversarial example attacking methods in the physical world. In our research, we believe that we made three creative contributions:

1. We proved that we can remotely gain the root privilege of APE and control the steering system.
2. We proved that we can disturb the autowipers function by using adversarial examples in the physical world.
3. We proved that we can mislead the Tesla car into the reverse lane with minor changes on the road.

They deeply examined and reverse engineered the algorithms used by the Tesla’s Autopilot and other systems.

They:

- Took over steering with a bluetooth-connected gamepad controller
- They spoofed rain...
- Most worrisomely, by understanding the autopilot's lane-recognition algorithm, they were able to induce a lane change to the left -- which would have been into oncoming traffic -- by strategically placing three large dots on the road ahead of the car.

This does NOT mean that a Tesla would turn into oncoming traffic, but rather that in this carefully crafted and isolated instance they were able to induce the car to redirect into the lane to the left. I would be shocked if the AI didn't have multiple other sensors and input to expressly forbid it from actually turning into oncoming traffic. So the headlines do Tesla a BIG disservice.

But there IS an important message here nevertheless: A car's autopilot really is an extremely complex interpreter of many sensory inputs. And how many times have we talked about how "interpreters" can be deliberately fooled by malicious actors who have access to the interpreter's internals? That's exactly what happened here. Within an isolated environment with preset conditions, the Tesla's autopilot was fooled by something that would never have fooled a human driver. We would think: "Huh… I wonder why there are three big weird dots on the road?"... whereas the Tesla AI seeing the same thing apparently thinks: "Oh! Time to change lanes to the left!"

A WONDERFUL CLASSIC HACK!...

ASUS "ShadowHammer" MAC addresses published

For whatever reason, Kaspersky chose not to publish their full list of MAC addresses.

Instead, they attempted to hide them inside a downloadable application by using a salted hash function to create a one-way test of a user's submitted MAC address. The application's code would then identically salt the hash of any user-submitted MAC address and check to see whether the result matched any of the hashes which were bundled with the app.

Hiding the MAC addresses in this way apparently annoyed the guys at the Australian security firm Skylight Cyber.

https://skylightcyber.com/2019/03/28/unleash-the-hash-shadowhammer-mac-list/

They wrote:

The question of who did this and why is intriguing, but not one we were trying to answer in this case. First thing's first - if information regarding targets exists, it should be made publicly available to the security community so we can better protect ourselves. Kaspersky have released an online tool that allows you to check your MAC address against a DB of victim MAC addresses (which is hidden). Good on Kaspersky on one hand, but on the other hand, this is highly inefficient, and does not really serve the security community.

So, we thought it would be a good idea to extract the list and make it public so that every security practitioner would be able to bulk compare them to known machines in their domain. If you are interested in the list it can be downloaded here or here for the extended list.

They also felt that having a simple list of targeted victim MAC addresses would be far more useful for large enterprises with many hundreds of thousands of systems where the stakes were pretty high -- were talking about the reliable installation of a Trojan Backdoor by unknown actors.

So how do we solve this problem? It's a variation of the classic brute force password cracking problem. Though it's significantly simplified because we know that every test MAC address is a 48-bit binary input to the cracking hash function.

The Skylight Cyber guys calculated that their own fastest computer would require about 162.5 days to extract all of the MAC addresses from Kaspersky's offline checker tool... And that was using a custom-tweaked build of HashCat to duplicate the customer SHA256 algorithm Kaspersky used, and with taking advantage of all the tricks they could think of, such as only checking the 24-bits of vendor information in the MAC that corresponded with known or believed possibilities.

So... they finally decided to hire out... afteer which they did the entire job in a bit less than an hour by renting an Amazon AWS p3.16xlarge instance which comes equipped with eight NVIDIA V100 Tesla 16GB GPUs.

The result is a lovely numerically ordered list of 583 MAC addresses...

https://skylightcyber.com/2019/03/28/unleash-the-hash-shadowhammer-mac-list/list.txt

(Some of the total of 619 hashes were absent and uncrackable, presumably because of untested vendor MACs.)

The ill fated and long awaited Windows 10 [last] October 2018 Update
Officially designated as ready in April of 2019.

Last Thursday, Microsoft announced that they are now designating Windows 10 October 2018 Update Build 1809 to be ready for broad, rather than targeted, deployment.

Microsoft's "Windows as a Service" evangelist John Wilcox said: "Based on the data and the feedback we’ve received from consumers, OEMs, ISVs, partners, and commercial customers, Windows 10, version 1809 has transitioned to broad deployment. With this, the Windows 10 release information page will now reflect Semi-Annual Channel (SAC) for version 1809. We will continue to communicate for future releases the transition from targeted to broad deployment status."

A quick note that VMware has fixed a bunch of things...
… including closing the holes that were revealed in the recent Vancouver PwntoOwn competition.

Android Security 10 Years In…

Last Friday, Google released their:
Android Security & Privacy 2018 Year In Review" / Celebrating 10 years of Android

https://source.android.com/security/reports/Google_Android_Security_2018_Report_Final.pdf

This 31-page report examined and shared Android's Platform Security, Google Play Protect, Ecosystem Data and PHA Families (Potentially Harmful Applications).

The Android security team's mission is to protect every one of the more than two billion Android users. We do this through massive investment and continuous improvement in our security and privacy technology and operations.

The report did share some interesting and impressive stats. For example:

The broadest statistic for measuring device hygiene is how frequently a full-device scan detects Potentially Harmful Applications (PHAs). Google Play Protect, Android's built-in defense mechanism, is incredibly effective at keeping PHAs out of Google Play, but malicious apps can still be downloaded from other sources. These apps endanger not only the device but also threaten the sanctity of the Android environment. This is why Google Play Protect scans all apps installed on a device regardless of the source.

In 2018 only 0.08% of devices that used Google Play exclusively for app downloads were affected by PHAs. In contrast, devices that installed apps from outside of Google Play were affected by PHAs eight times more often.Compared to the previous year, even those devices saw a 15% reduction in malware due to the vigilance of Google Play Protect.

Android's security saw a strengthened application sandbox in 2018 along with hardened developer APIs with features like Biometric Prompt and an updated target API level for apps. The Android Security team continued their investment in hardware backed security through discrete tamper-resistant secure elements that enable the use of industry-first security APIs, such as Protected Confirmation and Strongbox.

In 2018 we surpassed $3 million in total reward program payouts.

Our Android security rewards programs allow us to work with top researchers from around the world to improve the security of the Android ecosystem. These programs offer some of the highest priced rewards in the industry.

Through a combination of platform improvements like Treble , new original equipment manufacturer (OEM) agreements, and partner programs such as Android Enterprise Recommended, the Android ecosystem has made significant progress in releasing security updates.

In the 4th quarter of 2018 we had 84% more devices receiving a security update than in the same quarter the prior year.

Android platform security and privacy
Improving Android's security with every major Android release and monthly security updates is critical. However, in order to be even more effective, we must work to continuously increase security without putting a burden on our end users. A layered security model is part of our fundamental design principle and is a foundation of Android's architecture. The Android platform controls how the operating system works and how apps interact with other apps, device hardware, and other services. Supported by Google Play Protect, Android is protected around the clock. The following table lists some of these protections that are designed to provide better platform-level security.

The following table lists the platform security feature and the protection it provides:

- Encryption: Protects data from unauthorized access.

- Hardware-backed security: Strengthens key storage and cryptographic services and enables strong remote authentication.

- Kernel self-protections: Protects the kernel against memory corruption vulnerabilities and other security flaws in kernel components and drivers.

- Sandboxing: Keeps each app in a separate space, protecting its data and processing from other apps.

- SELinux: Provides an auditable definition of—and enforcement of—security boundaries on all operating system and app components above the kernel.

- Userspace hardening: Protects the operating system and apps against memory corruption vulnerabilities and other security flaws; includes address space layout randomization (ASLR), data execution prevention (DEP), and Control Flow Integrity (CFI).

- Verified boot: Cryptographically verifies that the operating system starts in a known good state.

With Android 9, we added a myriad of great security features. We strengthened the application sandbox and hardened the developer APIs. We continued to invest in hardware-backed security via the trusted execution environment (TEE) and on select devices through discrete tamper-resistant hardware. We also layered a set of privacy preserving enhancements and adopted more anti-exploitation techniques so that bugs don't turn into exploitable vulnerabilities.

On the topic of: Security research competitions and zero day vulnerabilities...

The Android Security & Privacy team participated in a number of external vulnerability discovery and disclosure competitions, including Mobile Pwn2Own, which took place at the PacSec conference in Tokyo, Japan. At this event, researchers were rewarded for demonstrating working exploits against major mobile operating systems. Exploits against Google Pixel devices were categorized in the top reward category along with other devices such as the iPhone. No exploits successfully compromised Google Pixel devices and none of the exploits demonstrated against devices running Android utilized a security vulnerability in the Android operating system. Further, in 2018, no critical security vulnerabilities affecting the Android platform were publicly disclosed without a security update or mitigation available for Android devices.

On: Android security updates program...

Google mitigates security vulnerabilities discovered through the Android Security Rewards program and additional engagements through regular Android security updates. In 2018, we continued to work with Android device manufacturers, mobile network operators, and system-on-chip (SOC) vendors to increase the number of devices receiving regular security updates. Through our combined efforts, which include platform improvements, new OEM agreements, and partner programs such as Android One and Android Enterprise Recommended, we've made significant progress in releasing the latest Android security updates. In fact, in the 4th quarter of 2018 we had 84% more devices receiving a security update than in the same quarter the prior year.

As of December 2018, over 95% of deployed Google Pixel 3 and Pixel 3 XL devices were running a security update from the last 90 days.

System image scanning:

In the Android Security 2017 Year in Review report, we announced that we had begun scanning for pre-installed PHAs across many software builds for devices with Google services. In 2018, we expanded this program and launched it as Build Test Suite (BTS) for partner OEMs.

BTS is similar to the Compatibility Test Suite (CTS). OEMs submit their new or updated build images to BTS. BTS then runs a series of tests that look for security issues on the system image. One of these security tests scans for pre-installed PHAs included in the system image. If we find a PHA on the build, we work with the OEM partner to remediate and remove the PHA from the build before it can be offered to users.

During its first calendar year, BTS prevented 242 builds with PHAs from entering the ecosystem.

Anytime BTS detects an issue we work with our OEM partners to remediate and understand how the application was included in the build. This teamwork has allowed us to identify and mitigate systemic threats to the ecosystem. Through the BTS program, we discovered, analyzed, and remediated PHA families such as Chamois and EagerFonts, which are described in detail in the PHA Families section. In 2019, we're continuing our commitment to vetting approved Android devices for security issues.

PHAs - Potentially Harmful Applications

Potentially Harmful Applications (PHAs) are apps that could put users, user data, or devices at risk. Common PHA categories include trojans, spyware, and phishing apps. In 2018, we started tracking click fraud as a PHA category. Click fraud apps simulate clicks on advertisements without user consent.User-wanted PHAsSome apps with attractive features also weaken Android's built-in security. When users try to install these apps, Google Play Protect warns users about potential hazards so that they can make informed decisions. Our statistics separate these from classic "malware" PHAs. For example, Google Play Protect warns users about apps that disable Android security features, such as SELinux, or root the device with disclosure and user consent. Google Play Protect discourages changes that lower Android’s built-in security protections, but allows individuals to choose the risks that they are willing to take with their devices.A warning message is displayed to the user anytime a PHA installation is detected. If they decide to ignore this warning and proceed with the installation, they will not receive further security warnings about that app. Interrupting the Android user experience with constant warnings would make Google Play Protect more annoying than useful.In 2018, user-wanted PHAs comprised 0.11% of app installations downloaded outside of Google Play (Google Play doesn’t allow any security-breaking apps even if they are user-wanted).

Mobile unwanted software (MUwS)

Google defines unwanted software (UwS) as apps that aren’t strictly malware, but are harmful to the software ecosystem. Mobile unwanted software (MUwS) impersonates other apps or collects at least one of the following without user consent:

- Device phone number
- Primary email address
- Information about installed apps
- Information about third-party accounts

Google Play policies prohibit MUwS, but users who decide to install software from outside of Google Play can still be affected by MUwS. To combat this issue, in 2018 Google Play began warning users about MUwS if they initiate a download from outside of Google Play. With this change, the total number of install attempts coming from MUwS apps declined from 2.09% in 2017 to 0.75% in 2018.

Changes in methodology

In 2018, we changed some of our methodology, which led to some variances in numbers in this report compared to last year. These changes include:

- Added click fraud to our PHAs definitions

- Introduced the concept of user-wanted PHAs. These apps are classified as PHAs, but are intentionally installed by users who want them for their unique system capabilities. For example, power users install apps to root their device or disable security settings, such as SELinux

- Introduced the don’t-warn-again concept. If Google Play Protect flags apps as PHAs, users receive a warning at the time of install, allowing them to make an informed decision to continue or cancel the installation. If they proceed, this specific installation is removed from the metrics for PHAs used for calculating device hygiene.

Threat landscape changes

In 2018, there were two notable changes to the Android threat landscape: an increase in pre-installed PHAs and backdoored SDKs (software development kits).

Pre-installed PHAs: Malicious actors increased their efforts to embed PHAs into the supply chain using two main entry points: new devices sold with pre-installed PHAs and over the air (OTA) updates that bundle legitimate system updates with PHAs. Neither entry point requires action from users, making them difficult to defend against.

There are three possible reasons for an increase in the number of pre-installed PHAs.

1. First, the developers of pre-installed PHAs only need to deceive the device manufacturer or another company in the supply chain instead of large numbers of users, so it’s easier to achieve large-scale distribution. Even a less popular device model can compromise hundreds of thousands of users through one pre-installed harmful application.

2. Second, pre-installed PHAs can gain more privileged access to the device, so it’s easier to execute malicious behavior that would usually be blocked by Android’s security model. At the same time, these additional privileges allow PHAs to defend against security tools or removal attempts by users.

3. Third, large families of PHAs used exploits to root devices, but this is increasingly more difficult due to Android’s constantly improving security model which blocks privilege escalation exploits to achieve similar privileges and defense levels for regular apps. Developers of these apps know that it is easier to compromise the supply chain of device manufacturers than to attack the Android platform security model.

To combat the problem of pre-installed PHAs, the Android Security team launched a security program in 2017 as part of the Android device certification process. We expanded the program in 2018 and now every new Android-certified device goes through the same app scanning process as apps on Google Play. Additionally, our security scanner looks for other common security and privacy issues and denies device certification until device manufacturers fix these problems.

Backdoored SDKs, apps, and other code:
Some SDKs appear legitimate, but include behaviors and functionality that the app developer may not have known about when they included the SDK. This functionality may compromise user data, device integrity, or experience. It may also be used as a part of a larger initiative, such as committing click fraud, mining cryptocurrency, or app install attribution fraud.

Here are some approaches developers use to include malicious code in legitimate SDKs:

- Backdoored SDKs with legitimate functionality
- Backdoored Android system code that injects the malicious code into every app on the device
- Modified Google apps with backdoor code injected
- Modified SDKs rehosted with similar names to confuse legitimate developers into accidentally downloading them.

Hundreds of apps have been affected by backdoored code. We have been working with impacted developers to educate them about this new threat and to publish updated versions of their apps without the backdoor code.

Device and ecosystem hygiene:
The broadest statistic for measuring device hygiene is how frequently a routine full-device scan detects PHAs. Since we began to measure device hygiene in late 2014, an average of less than 1% of devices have PHAs installed at any point. This trend remained steady in 2018.

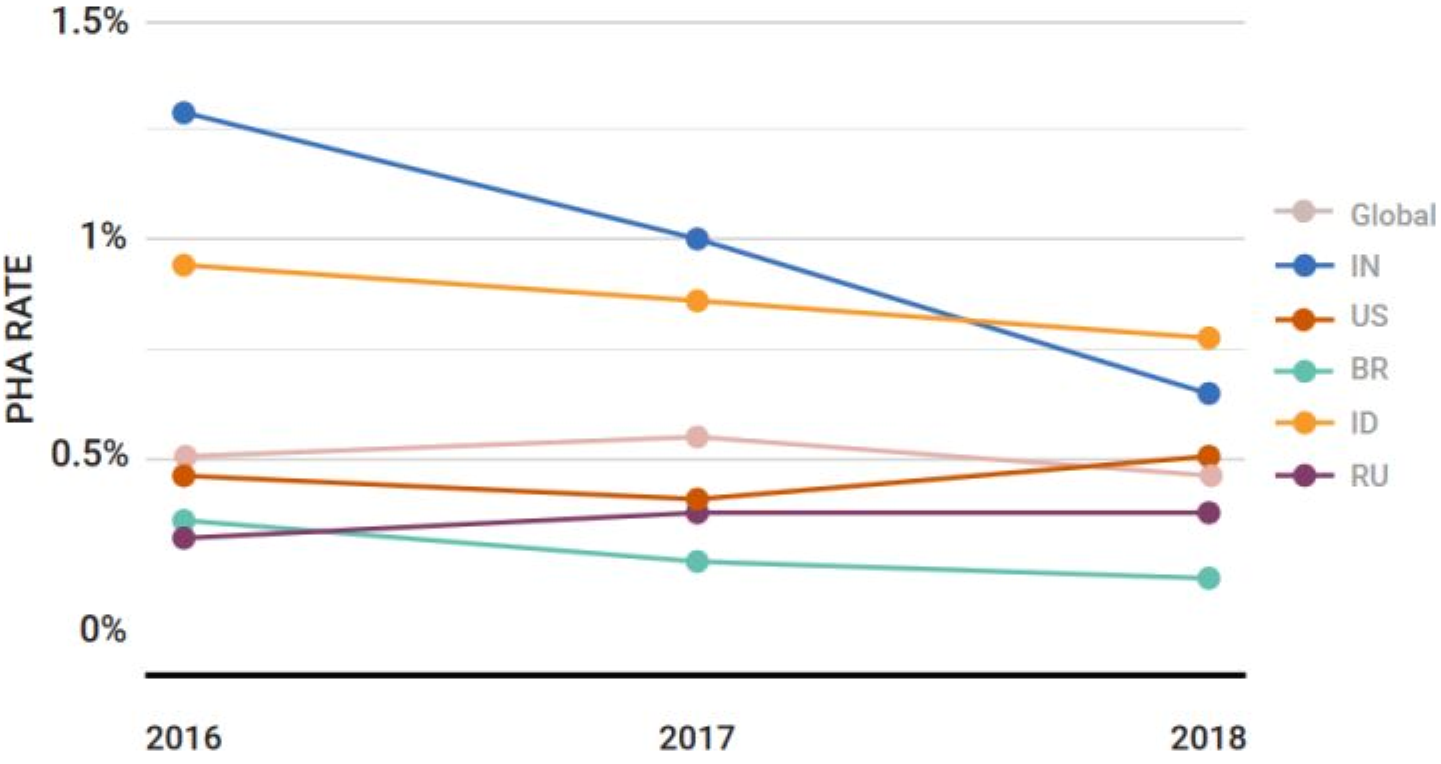
In 2018, 0.45% of all Android devices running Google Play Protect had installed PHAs, compared to 0.56% of PHA-affected devices in 2017. This equates to a 20% year-over-year improvement to the health of the Android ecosystem.

Apps available on Google Play must adhere to published policies and are reviewed to verify their compliance. Of the millions of apps published on Google Play, only 0.08% of devices that exclusively used Google Play had one or more PHAs installed (unchanged from last year).

In contrast, 0.68% of devices that installed apps from outside of Google Play were affected by one or more PHAs in 2018. While this number is 8 times higher than devices that exclusively used Google Play, it’s a noticeable improvement from 0.80% in 2017.

Devices that installed apps only from Google Play were 8 times less likely to be affected by PHAs.

Device hygiene in the largest Android markets
While Google Play and the overall Android ecosystem became safer in 2018, there is some variance based on the country of the device. Of the five largest Android markets, three (India, Indonesia, Brazil) became cleaner year-over-year, one (Russia) stayed at the same level, and for one the numbers went up (USA). These markets have unique challenges for threat severity and distribution of PHAs, so each is discussed in more detail in this section.



In India, which is by far the biggest Android market, the number of devices affected by PHAs has decreased each year. In 2018, 0.65% of all Indian devices were affected by PHAs at any time, a 35% drop from the previous year. For the first time, India didn't have the highest device rate of PHAs among the top Android markets. In India, most PHAs were Trojans, backdoors, or hostile downloaders that downloaded more PHAs onto devices. These apps were introduced to users through supply chain attacks either in the form of pre-installed apps on new devices or OTA updates handled by untrustworthy OTA companies. Pre-installed apps from the EagerFonts, Snowfox, and Chamois families were the most common. For more details on these families, see PHA Families. Two apps outside of this category are versions of a popular video player that mines cryptocurrency in the background without user consent. Mobile devices have been damaged by cryptocurrency mining in the past, so we flag apps with this behavior as PHAs.

As in previous years, **the USA** is the second biggest Android market. In 2018, the number of impacted devices rose from 0.4% to 0.5% due to the introduction of click fraud as a PHA category. However, compared to India, the USA's context for PHAs is different and less severe. Eight of the top ten PHAs in the USA are wanted by users or don't significantly impact them directly. Of these eight, four are power-user tools for rooting devices or for circumventing other security settings and four are click fraud apps (of the CardinalFall and Snowfox families) that may harm advertising networks but not users. As a reminder, click fraud wasn't considered a PHA before 2018 and was treated only as a Google Play policy violation. Thus, the increase is significantly impacted by the inclusion of click fraud as a PHA category. Only two of the top ten are hostile downloaders or Trojans. In the USA, Of the top ten PHAs, only one was distributed pre-installed on the device, and that was limited to a single device type from one carrier.

Brazil continued to be the third-largest Android market in 2018. In 2017, and again in 2018, Brazilian Android devices were the cleanest among major markets with a 0.23% PHAs device rate (about half the global average). While the number of affected devices was lower than in other markets, the data suggests that Brazil, like India, still faces challenges in ensuring device integrity through the supply chain process. Four of the top ten PHAs in Brazil were pre-installed PHAs that were shipped on devices of a single Brazilian original equipment manufacturer (OEM). Two others were OEM-specific third-party stores from a different OEM that offered high numbers of PHAs to users for download. The remaining four PHAs on the top ten list were power user tools that disabled security settings to enable app piracy.

Indonesia is the fourth-largest Android market and the major market most affected by PHAs. Its overall landscape of PHAs looks similar to India's, likely because the Android OEMs and distributors targeting these countries are similar. That means Indonesia's Android PHAs are hostile and dangerous to users, and many PHAs come pre-installed on devices sold and used in Indonesia without end users' knowledge. Of the top ten PHAs in Indonesia in 2018, four were pre-installed Trojans posing as system settings apps, font manager apps, or quick search apps. Two others were Trojanized utility apps (a flashlight app and a camera app) for which the distribution vectors are unknown.

The last of the five largest Android markets is **Russia**. While below the world-wide average for PHA infections, Russia's infections are user-hostile and often pre-installed on devices. Seven of the top ten PHAs in Russia in 2018 were pre-installed Trojans. Several of those have been known to us since 2016 (Chamois family), which hints at a greater device longevity in Russia compared to the other surveyed countries. Two more of the top ten apps were problematic third-party app stores that either offer many PHAs for download or tamper with device security settings without user consent.

Device hygiene by Android version
Newer versions of Android are less affected by PHAs.
Android 8 and Android 9 have PHA rates that are at 0.19% and 0.18% respectively.

We attribute this trend to advancements in platform security. In particular, newer Android versions are more resilient to privilege escalation attacks that previously allowed PHAs to gain persistence on devices and protect themselves against removal attempts. On newer versions, GPP is effectively cleaning PHAs. In conjunction with platform changes, GPP is preventing PHAs from protecting themselves from removal or disabling.

PHA install rates from Google Play increased from 0.02% in 2017 to 0.04% in 2018 (+100% YoY). This increase is due to the inclusion of click fraud as a PHA category. As mentioned earlier, before 2018 click fraud wasn’t considered a PHA and was treated only as a Google Play policy violation. If we remove the numbers for click fraud from these stats, the data shows that PHAs on Google Play declined by 31% year-over-year.

In Google Play, we can remove any app that shows PHA behavior. For apps outside of Google Play, this isn’t the case. Because of this difference, in addition to actual installs, we also track installation attempts for apps installed from outside of Google Play. Not all installation attempts result in actual installs: if users heed the warning that an app is a type of PHA, they won’t install the app. In particularly harmful cases, such as ransomware or banking phishing apps, Google Play Protect blocks the installation in addition to warning users.

In 2018 the total number of app installations in Google Play grew 16% compared to 2017.

The sideloaded PHA install rate also showed a drastic reduction from 1.48% in 2017 to 0.92% in 2018 (-38% YoY).

Outside of Google Play, PHA installation attempts in 2018 fell by 20% from the previous year. Google Play Protect stopped 73% of PHA installations from outside of Google Play in 2018 compared to 71% in 2017 and 59% in 2016. The other 27% were a combination of apps that were installed before we identified them as PHAs and users who ignored Google Play Protect's warnings.

Google Play Protect prevented 1.6 billion PHA installation attempts from outside of Google Play in 2018.

Outside of Google Play: Backdoors
In 2018, backdoors were the most prevalent PHA category outside of Google Play where they make up 28.0% of all PHA installs and 0.26% of all app installs (up from 0.22% in 2017). Last year, backdoor apps were mainly targeting devices in Russia, Brazil, Mexico, and Vietnam.

As the prevalence of Trojans and hostile downloaders decreased in 2018, backdoors took the top spot. However, the spread of backdoor PHAs is attributed to a specific PHA family—Chamois—that we previously discussed publicly. Chamois apps are pre-installed on popular devices from different OEMs that didn't carefully scan for malware. As a consequence, users are buying compromised systems. When users start up their new devices, the pre-installed Chamois apps (usually disguised as system apps) download and install PHAs and other apps in the background.

Like hostile downloaders, Trojans have many variations and aren't from a particular PHA family. The only noteworthy Trojans are those that mine cryptocurrency without user consent. As cryptocurrency prices rose dramatically at the end of 2017 and early 2018, the number of malicious actors also rose. Google Play Protect started warning users about the potential problems of cryptocurrency mining on their devices. In 2018, 4 of the top 11 Trojans were cryptocurrency miners, all embedded in a popular video player app (previously mentioned as the most prevalent PHA in India).

PHA Families
Google Play Protect removes all apps it detects from Google Play that are part of any of the PHA families listed below.

Chamois

Chamois was one of the most impactful PHA families in Android in 2018 with more than 199 million installs. It originally emerged in late 2016 and again in early 2017; Google detected and disrupted the first two variants. After an eight month hiatus, Chamois re-emerged in November 2017 outside of Google Play. Chamois uses a variety of distribution mechanisms including being pre-installed, added as an advertising SDK, and injected into popular sideloaded applications. The Android security team implemented detection and remediation techniques across these channels, leading to a sharp decline in installs in 2018.

Chamois is a well-engineered, sophisticated piece of malware. As of November 2018, there were five known variants of the Chamois botnet family, three of which emerged after November 2017. These variants are comprised of four or five stages with anti-analysis features and a command-and-control infrastructure for deploying their payload. Google Play Protect classifies Chamois as a backdoor due to the remote command-and-control capabilities it has. The payloads for Chamois range from a variety of ad fraud payloads to SMS fraud to dynamic code loading.

Snowfox

Snowfox is an advertising SDK with two variants; one variant steals OAuth tokens from a device and the other injects JavaScript for click fraud into WebViews with loaded ads. The Snowfox campaign began in late 2017 and peaked in March 2018. Over the course of 2018, apps with the malicious Snowfox SDK were installed more than 16 million times.

Snowfox is predominantly distributed outside of Google Play by apps including the SDK. However there are some distribution mechanisms where an application dynamically downloads the Snowfox SDK as a plugin, probably to bypass static analysis.

Cosiloon

Cosiloon is a family of hostile downloader PHAs that was pre-installed on uncertified Android devices. Cosiloon apps are two-stage PHAs with the first stage pre-installed on the device. There are two variants of the first stage: a standalone, pre-installed application and a backdoored System UI application. The first stage downloads and installs the second stage, which shows ads and installs other PHAs.

Cosiloon was initially detected in November 2017. In early 2018, Avast and Google Play Protect collaboratedon a threat analysis. Google Play Protect then deployed two remediation solutions to protect users. By March 2018, Cosiloon was largely eradicated from the Android ecosystem. Because it was pre-installed, Cosiloon was exclusively found outside of Google Play.

BreadSMS

BreadSMS is a large PHA family that Google Play Protect began tracking in the beginning of 2017. BreadSMS evolved rapidly in 2018, accumulating over 11 million installs with approximately 98% of those occurring on Google Play.

In 2018, BreadSMS added cloaking and obfuscation techniques to evade detection. For example, when some BreadSMS apps detect that they are being analyzed, a disclosure and consent dialog opens for premium SMS messages. The dialog doesn’t appear when running on users’ devices. In 2018, BreadSMS added WAP fraud, subscribing users to services that charge their mobile bills without their knowledge or consent.

In 2018, 52.6% of BreadSMS installs occurred in Thailand and 19.1% occurred in Malaysia.

View SDK

View SDK is a monetization SDK that uses JavaScript to perform ad click fraud. View SDK was originally discovered by Google Play Protect in December 2017. However, Google Play Protect didn't begin treating click fraud as a PHA category until March 2018. During 2018, there were approximately 5.2 million app installs containing View SDK. All of these installs were sideloaded; none occurred from Google Play.

Affected apps drop a JAR file containing View SDK during execution. View SDK then downloads JavaScript from a remote server and injects it into a WebView showing ads and triggering fake ad clicks without user intervention.

Triada

Triada software was first documented by Kaspersky in 2016 as a rooting Trojan. In mid-2017, Dr. Web documented a pre-installed backdoor variant where the log function in the Android framework of the device’s firmware had been backdoored. Any application related to Triada can then communicate and perform commands using the backdoored log method. This variant also injects code into other processes, such as browsers or Google Play. The purpose of this backdoor was to show ads and install other applications.

During 2018, Google identified all Triada variants, including new ones, and all devices infected with Triada. Based on this information Google reached out to OEMs to remediate. OEMs provided users with system updates that removed the Triada backdoor.

CardinalFall

CardinalFall is a large PHA family with an SDK that implements click fraud and, in some cases, dynamic code loading. The CardinalFall SDK uses encrypted communication with a number of command-and-control servers to evade detection. The servers provide CardinalFall apps with specific ad IDs (including AdMob and Facebook ad IDs) that are used to serve ads to the affected devices. When an ad is fetched, the CardinalFall app uses the click API on a WebView to automatically trigger fake clicks on the loaded ad without user intervention.

More than 90% of all CardinalFall apps have been found on Google Play. The countries most significantly affected by this PHA family in 2018 were India, the US, and Pakistan; although users in more than 10 other countries were also impacted.

FlashingPuma

FlashingPuma is a large click fraud family discovered in 2018. More than 75% of these apps are from outside of Google Play. The FlashingPuma SDK fetches ads and generates fake clicks without user intervention, crediting the fraudsters.

FlashingPuma apps were propagated worldwide, but India was the most affected country with 30% of all installs. Enforcement fluctuated during 2018, as new variants were being produced and identified. Now Google Play Protect is detecting all apps that belong to this family, whose footprint is significantly reduced.

EagerFonts

EagerFonts is an SDK embedded in the Fonts apps that come pre-installed on some Android devices. The EagerFonts PHA family was discovered by Google in mid-2018 and at its peak was present in approximately 12 million devices globally across hundreds of OEMs. Google Play Protect treats EagerFonts as a backdoor because it uses a remote server to dynamically download and run fake plugins; these are known PHAs, including Chamois and Snowfox. Unlike other hostile downloaders, EagerFonts didn't install an app, but loaded and ran the fake plugins dynamically in the original app’s process.

Google has worked with the affected OEMs to remediate the EagerFonts infections. Upon discovering this family, Google immediately stemmed the infection by halting the shipment of new builds containing the PHAs.

Idle Coconut

Idle Coconut is an SDK that developers include in their apps for monetization. The apps double as end points of a certain commercial VPN that routes traffic through affected Android devices. The SDKs use a websocket for communications with a command-and-control (CnC) server and then connect to hosts that the CnC commanded over “normal” sockets. None of this behavior is disclosed and the user’s device unknowingly becomes used in a proxy network. Thus, Google flags these apps as Trojans.

About 60% of the installs of Idle Coconut in 2018 came from Google Play and approximately 40% were from sideloaded applications. India had the highest rate of installs (approximately 25%) during 2018.

After the twelve and a half years of this podcast, I would imagine that all of our listeners likely have a realistic and sober appreciation for the difficulty of the task Google has willfully undertaken by shepherding Android. It’s a challenge I would never want to face. They are placing a highly capable and powerful open source operating system running atop equally powerful hardware, sourced very indirectly through partners far and wide and largely out of their control, into the hands of inherently trusting and non-computer-savvy consumers… and for the most part, in an openly hostile environment and against quite unfavorable odds, the damn thing is working.

So I say Bravo Google. And thank you.

~30~