Security Now! #698 - 01-22-19

Which Mobile VPN Client?

This week on Security Now!

This week we examine a very worrisome WiFi bug affecting billions of devices, a new fun category for the forthcoming Pwn2Own, Russia's ongoing, failing and flailing efforts to control the Internet, the return of the Anubis Android banking malware, Google's changing policy for phone ans SMS App access, Tim Cook's note in TIME Magazine, news of a nice Facebook Ad auditing page, another Cisco default password nightmare in widely used lower-end devices, some errata, miscellany, listener feedback and then we answer the age-old, and apparently quite confusing question... which is the right VPN client for Android?

Oscar's Latest DEC PDP resurrection!

Oscar Vermeulen: http://obsolescence.wixsite.com/obsolescence/pidp-11

"Price is $250 plus shipping. I apologise, because that is almost $90 more than the PiDP-8. But if you knew the upfront cost of making that injection-molded case, you would understand it's not because I am Turning Commercial! This is still very much a hobby project, although one that got slightly out of hand."

A few words about the PDP-11...

Security News

An incredibly widespread WiFi firmware bug affects billions of devices.
https://embedi.org/blog/remotely-compromise-devices-by-using-bugs-in-marvell-avastar-wi-fi-from-zero-knowledge-to-zero-click-rce/#the-most-interesting-bug-to-be-exploited

The title of Embedi's disclosure is: "Remotely compromise devices by using bugs in Marvell Avastar Wi-Fi: from zero knowledge to zero-click RCE."

One of the most popular WiFi chipsets on the market, the Marvell Avastar 88W8897, was chosen for use by many high-volume consumer devices including the Sony PlayStation 4, the Xbox One, Microsoft's Surface laptops, Samsung's Chromebooks, the Samsung Galaxy J1 smartphones, Valve SteamLink cast devices, some laptops, several routers other embedded devices, and network access hardware. This chipset uses the most popular of the embedded real-time operating systems, known as "ThreadX".

Late last spring a researcher, Denis Selianin (sell-ee-ah-nin) who is with the embedded security firm Embedi was experimenting with "fuzzing" that highly-popular WiFi chipset... and discovered four problems, two of which are critical. And the problem is, this firmware is currently embedded into billions of devices, many if not most of which may not be subject to updating.

The any of these WiFi-enabled devices are powered up, these bugs allow malicious attackers to force them to execute arbitrary code without requiring any action on the part of the device owner or user. In other words, this problem could not be any worse.

The attack is triggered whenever an affected device searches for available Wifi networks, which, as we know, happens automatically and continuously in the background.

The root of the problem, as I noted above, lies in ThreadX. The four vulnerabilities exploit a memory corruption bug referred to as a “block pool overflow” to introduce malicious code onto a device.

One of these bugs is specific to the very widely used Marvell Avastar 88W8897 Wifi controller, but the others can affect any device based on ThreadX using the same techniques. Embedi cites ThreadX's own website as the source of its statement that over six billion devices have been deployed running this firmware.

Because affected Wi-Fi devices are set to scan for new networks every five minutes, regardless of whether or not they are already connected to a Wi-Fi network, this bug, writes Embedi: “provides an opportunity to exploit devices with zero-click interaction at any state of wireless connection.” Once malicious code is introduced onto the Wifi controller, other techniques can be exploited to send data to the device's application processor... resulting in a complete takeover and compromise of the affected device.

An attacker would not need to know a target's Wi-Fi SSID name or password. The target device only needs to be turned on with the attacker broadcasting malicious packets within radio signal range of the target device.

Embedi tested the vulnerabilities using a Valve Steamlink game streaming device, which is based on Linux features an ARM SoC and the affected Marvell Wifi controller. This device was chosen because it allowed for research tools to run without breaking DRM restrictions.

The vulnerabilities were disclosed to Marvell in early May 2018. Denis presented his findings and a proof of concept at the Zero Nights security conference in late November 2018 and they have only just published all of his research, including a video showing the attack in progress.

At the moment no fixes have been issued. But according to the November presentation, work was in progress.

Denis wrote:

"I've managed to identify ~4 total memory corruption issues in some parts of the firmware. Due to AFL being able to mutate input in a way that can't be passed to the fuzzed function (for example, because of some sanitization checks before the fuzzed function) it is hard to investigate a potential impact that may be caused by these issues. I also tried to reproduce these bugs on different versions of the firmware and different versions of wireless SoCs and it looks like bugs are present in many of them.

The most interesting bug to be exploited

One of the discovered vulnerabilities was a special case of ThreadX block pool overflow. This vulnerability can be triggered without user interaction during the scanning for available networks. This procedure is launched every 5 minutes regardless of a device being connected to some Wifi network or not. That's why this bug is so cool and provides an opportunity to exploit devices literally with zero-click interaction at any state of wireless connection (even when a device isn't connected to any network). For example, one can do RCE in just powered-on Samsung Chromebook. So just to summarize:

- *It doesn't require any user interaction.*
- *It can be triggered every 5 minutes in case of GNU/Linux operating system.*
- *It doesn't require the knowledge of a Wi-Fi network name or passphrase/key.*
- *It can be triggered even when a device isn't connected to any Wifi network, just powered on.*

Here, I will describe how to achieve arbitrary code execution on WiFi SoC. Details on escalation techniques will be presented further in this article.

(((After explaining the details of block-based linking and memory management, Denis writes:)))

So, we have 2 techniques to exploit ThreadX block pool overflow. One is generic and can be applied to any ThreadX-based firmware (in case it has a block pool overflow bug, and the next block is free). The second technique is specific to the implementation of Marvell Wi-Fi firmware and works if the next block is busy. In other words, by combining them together we can achieve reliable exploitation.

(((He wraps this up by saying:)))

Some important things may be learned from this story:

- *Wireless devices expose HUGE attack surface.*
- *Usually, there's no exploitation mitigation on wireless SoC.*
- *Device drivers may expose WIDE attack surface for escalation from a device to host application processor even in cases when a device doesn't have direct access to host memory.*

Demonstration PoC video: https://youtu.be/syWIn62M72Y

The chip is a closed system. There may not be any way for its firmware to be updated. Or no programmatic way without hooking up a JTAG programmer. And since the attack is INSIDE the WiFi subsystem, it is independent of the operating system and cannot be blocked by the OS.

Compared with all other attacks, the only thing that would make this worse was if it was an over-the-Internet attack that could be launched from anywhere. But this promises to seriously whet the appetites of higher-end hackers. They'll do the heavy lifting and produce turn-key exploit kits that less capable hackers will use.

I'm SURE Marvell has already fixed this for all new sales of this very nice chipset. But firmware updates to existing embedded systems seem very unlikely. And we know that even if such updates were forthcoming, many devices would never be updated. So it seems quite unlikely that we haven't heard the last of this.

Pwn2Own adds a Tesla Model 3 to its hacking lineup!

There's MONEY in them thar hacks! March's forthcoming CanSecWest to be in Vancouver will, for the first time, add automobiles to its hacking target -- and cash prize -- lineup. In fact, more than $900,000 worth of prizes will be awarded (or at least available) for successful attacks that subvert a variety of a Tesla Model 3's onboard systems.

The biggest prizes will be $250,000 for hacks that execute code on the car's three primary systems: The gateway, the autopilot, and the VCSEC. The gateway is the central hub that interconnects the car's powertrain, chassis, and other components and processes the data they send. As we know, the autopilot is a driver assistance feature that helps control lane changing, parking, and other driving functions. The VCSEC the Vehicle Controller Secondary. It's responsible for security functions, including the alarm.

These three systems represent the most critical parts of a Tesla, so it's clear why hacks successfully targeting them would be eligible for big payouts. To qualify, the exploits must force the gateway, autopilot, or VCSEC to communicate with a rogue base station or other malicious entity. Meanwhile, a denial-of-service attack that takes out the car's autopilot will pay $50,000.

Pwn2Own will pay $100,000 for hacks that attack the Tesla's key fob or Phone-as-Key either by achieving code execution, unlocking the vehicle, or starting the engine without using the key. The competition will also pay a $100,000 add-on prize for winning hacks in another category that attack the car's controller area network, or CAN bus. This system allows microcontrollers and devices to communicate with each other.

Hacks targeting the car's infotainment system is on the menu for $35,000, and hacks which escape the security sandbox or escalate privileges to root or access the OS kernel will fetch $85,000.

WiFi or Bluetooth hacks will pay $60,000. A separate add-on payment of $50,000 will be paid for winning hacks that achieve persistence, which means they maintain root access even after a reboot.

It's going to fun and interesting to see how the Tesla fares against the world's top hackers!

And, as always, the secure of things without wheels will also be tested: The category of virtualization will award $250,000 for a successful Hyper-V client guest-to-host escalation and $150,000, $70,000, and $35,000 for hacks of VMware ESXi, VMware Workstation, and Oracle VirtualBox, respectively. The Web-browser attack category will pay $80,000 for hacks of Chrome and Microsoft Edge with a Windows Defender Application Guard-specific escape. A Firefox exploit will net $40,000. The Server Side category is much smaller this year with Microsoft Windows RDP as the only target. Most of our server side targets moved to Trend Micro's Targeted Incentive Program, so they no longer need to be included in Pwn2Own. Still, a successful RDP exploit will bag $150,000 for the contestant.

The Hacking competition will be taking place March 20-22 and I'm sure we'll be covering the results on Tuesday the 26th (the day I turn 64!).

Russia's not doing so well with blocking Internet services they dislike

As we discussed at the time, back in April of 2018 the Russian agency responsible for censoring Russians' access to the Internet -- Roskomnadzor -- attempted to block Telegram after Telegram ignores their threats of blocking the service. Recall that after the initial block, Telegram moved their servers into the cloud network space of Amazon and Google... which resulted in the blocking of many more critical services than just Telegram. It turns out, it's easier said than done.

And for their part, Telegram users also evaded the blocking by using VPNs and various available proxy services. Russia again countered by expanding its block list and ended up blocking even more.

Reuters later reported in August of 2018 that Russia then reportedly started testing "more precise technology to block individual online services" after the attempt to block Telegram failed, but "Moscow has yet to find a way to shut it down without hitting other traffic." In 2016, Russia had also attempted to block LinkedIn with limited success.

So why are we talking about this now? Because Russia has now set their sites on Facebook and Twitter: Roskomnadzor (the Russian government agency responsible for censorship on the Internet) last December 17th sent letters to both Facebook and Twitter has accusing them of failing to comply with a law requiring all servers that store personal data to be located in Russia. The letters gave each company 30 days to provide "a legally valid response." Well, that time was up last week with neither company bothering to reply. So The Wall Street Journal reports that

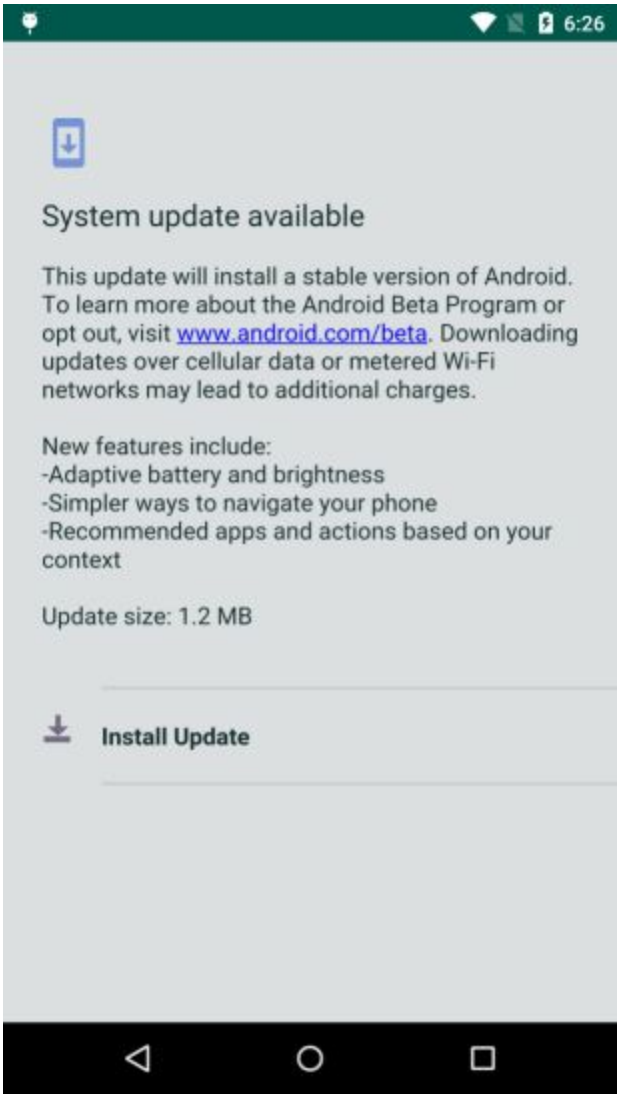
today, "Roskomnadzor begins administrative proceedings against both companies."

The Russian censorship agency, "said the social-media networks hadn't submitted any formal and specific plans or submitted an acceptable explanation of when they would meet the country's requirements that all servers used to store Russians' personal data be located in Russia."

Russia had previously threatened to block Facebook over its non-compliance with the data-storage law in both 2017 and 2018.

So... This should be interesting. I wouldn't be surprised to learn that these large global mega-service companies had deliberately adopted networking architectures that made their blocking difficult.

Anubis: The banking malware you really don't want in your Android smartphone
Trend Micro writes: "Google Play Apps Drop Anubis Banking Malware, Use Motion-based Evasion Tactics"



The Anubis Trojan has been observed attacking 377 different bank applications from 93 countries around the world, with banks like Santander, RBS, Natwest, and Citibank, as well as non-banking apps such as Amazon, eBay, and PayPal among a great many others. It is an aggressive and capable banking malware Trojan. Trend Micro recently discovered it hiding inside two Android Google Play Store apps with dozens of fake five-star ratings and thousands of installations onto Android devices.

The two apps are: Currency Converter and BatterySaverMobi. And what sets these apart from other malware-carrying Android apps is their use of their host's motion sensors to detect whether they've been installed in a malware analysis sandbox... in which case their malicious behavior is suppressed and the apps behave themselves. In other words, they use motion sensors to determine whether or not they're under analysis.

I wanted to note these just to shake my head and share another example of the cat and mouse game the bad guys are playing. Creating a secure system which is also powerful and flexible is something which has, so far, eluded the best brains in computer science. We could have a system that's closed, like the original Apple iPhone. But people want apps. We want open. I certainly do. And we want capable apps. But, so far, with app capability comes app liability and exploitability. We haven't yet figured out how to get the one we want without inviting the others we don't want.

And then there's social engineering. The previous page shows the dialog that Anubis uses to obtain the administrative rights it needs by displaying an entirely believable dialog of the sort smartphone users see all the time. Whom among us would think twice before clicking "Install Update." We CLEARLY need our platform to protect us, not only from others, but also from ourselves.

Google is cracking down on Android Phone- and SMS-using Apps
https://android-developers.googleblog.com/2019/01/reminder-smscall-log-policy-changes.html

Posted last Monday to the Android Developers Blog, Paul Bankhead, the Director of Product Management for Google Play wrote:

Reminder SMS/Call Log Policy Changes / 14 January 2019

TLDR; As previously announced and directly communicated to developers via email, we'll be removing apps from the Google Play Store that ask for SMS or Call Log permission and have not submitted a Permissions Declaration Form. If you have not submitted a permissions declaration form and your app is removed, see below for next steps.

We take access to sensitive data and permissions very seriously. This is especially true with SMS and Call Log permissions, which were designed to allow users to pick their favorite dialer or messaging app, but have also been used to enable many other experiences that might not require that same level of access. In an effort to improve users' control over their data, last October we announced we would be restricting developer access to SMS and Call Log permissions.

Our new policy is designed to ensure that apps asking for these permissions need full and ongoing access to the sensitive data in order to accomplish the app's primary use case, and that users will understand why this data would be required for the app to function.

Developers whose apps used these permissions prior to our announcement were notified by email and given 90 days to either remove the permissions, or submit a permissions declaration form to enable further review.

More about app reviews
We take this review process seriously and understand it's a change for many developers. We apply the same criteria to all developers, including dozens of Google apps. We added to the list of approved use cases over the last few months as we evaluated feedback from developers.

Our global teams carefully review each submission. During the review process, we consider the following:

- Likelihood that an average user would understand why this type of app needs full access to the data.
- User benefit of the feature.
- Importance of the permission relative to the core functionality of the app.
- Risks presented by all apps with this use case having access to this sensitive data.
- Availability of more narrow alternatives for enabling the feature.

With this change, some uses cases will no longer be allowed. However, many of the apps we reviewed with one of these permissions can rely on narrower APIs, reducing the scope of access while accomplishing similar functionality. For example, developers using SMS for account verification can alternatively use the SMS Retriever API, and apps that want to share content using SMS can prepopulate a message and trigger the default SMS app to show via intents. Tens of thousands of developers have already resubmitted their apps to support the new policy or have submitted a form. Thank you! Developers who submitted a form received a compliance extension until March 9th.

Next steps

Over the next few weeks, we will be removing apps from the Play Store that ask for SMS or Call Log permission and have not submitted a permission declaration form. If your app is removed and you would like to have it republished, you can do one of the following in the Play Console:

- submit a new version without these permissions, or
- submit a new version of your app that retains the permissions.
  Doing so will require you to complete a permissions declaration form inside the Play Console (coming soon) and will give you an extension until March 9th to remove the permissions or receive approval for your use case.

Keeping our overall Android ecosystem healthy is very important, and protection of user data is vital to the long term health of all developers. We know these changes have required significant work from you and we appreciate your efforts to create innovative experiences while protecting user's privacy.

TIME: "You Deserve Privacy Online. Here's How You Could Actually Get It"
Tim Cook / Last Wednesday, January 16th <quoting Tim's statement>

Last week, Time Magazine printed a statement by Apple's Tim Cook which took aim at the largely hidden data brokerage industry that has quietly sprung up over the past decade. We've touched on them a bit from time to time. Sometimes I'll encounter one of their chilling websites touting everything that they know about us and I'll share that on this podcast. So I think Tim raises some important points. Because of the accepted Apple vs Facebook and Google profit models, anything Tim says tends to be viewed through the lens of "we don't profit from the collection of your data, but they do." However that's not what he wrote because what he wrote was platform neutral, important (if not earth shaking), and exactly right. He said:

"We all deserve control over our digital lives. That's why we must rein in the data brokers.

In 2019, it's time to stand up for the right to privacy—yours, mine, all of ours. Consumers shouldn't have to tolerate another year of companies irresponsibly amassing huge user profiles, data breaches that seem out of control and the vanishing ability to control our own digital lives.

This problem is solvable—it isn't too big, too challenging or too late. Innovation, breakthrough ideas and great features can go hand in hand with user privacy—and they must. Realizing technology's potential depends on it.

That's why I and others are calling on the U.S. Congress to pass comprehensive federal privacy legislation—a landmark package of reforms that protect and empower the consumer. Last year, before a global body of privacy regulators, I laid out four principles that I believe should guide legislation:

- *First, the right to have personal data minimized. Companies should challenge themselves to strip identifying information from customer data or avoid collecting it in the first place.*
- *Second, the right to knowledge—to know what data is being collected and why.*
- *Third, the right to access. Companies should make it easy for you to access, correct and delete your personal data. And,*
- *Fourth, the right to data security, without which trust is impossible.*

But laws alone aren't enough to ensure that individuals can make use of their privacy rights. We also need to give people tools that they can use to take action. To that end, here's an idea that could make a real difference.

One of the biggest challenges in protecting privacy is that many of the violations are invisible. For example, you might have bought a product from an online retailer—something most of us have done. But what the retailer doesn't tell you is that it then turned around and sold or transferred information about your purchase to a "data broker"—a company that exists purely to collect your information, package it and sell it to yet another buyer.

The trail disappears before you even know there is a trail. Right now, all of these secondary markets for your information exist in a shadow economy that's largely unchecked—out of sight of consumers, regulators and lawmakers.

Let's be clear: you never signed up for that. We think every user should have the chance to say, "Wait a minute. That's my information that you're selling, and I didn't consent."

Meaningful, comprehensive federal privacy legislation should not only aim to put consumers in control of their data, it should also shine a light on actors trafficking in your data behind the scenes. Some state laws are looking to accomplish just that, but right now there is no federal standard protecting Americans from these practices. That's why we believe the Federal Trade Commission should establish a data-broker clearinghouse, requiring all data brokers to register, enabling consumers to track the transactions that have bundled and sold their data from place to place, and giving users the power to delete their data on demand, freely, easily and online, once and for all.

As this debate kicks off, there will be plenty of proposals and competing interests for policymakers to consider. We cannot lose sight of the most important constituency: individuals trying to win back their right to privacy. Technology has the potential to keep changing the world for the better, but it will never achieve that potential without the full faith and confidence of the people who use it."

Once again, I find myself feeling as though we're still in the very early days of this explosion in processing power, collapse in the cost of mass storage, and the connectivity created by the Internet. And, not surprisingly, the regulatory framework that's needed to govern the implications of these changes is lagging far far behind. Few of those who would presume to create the framework we need have any idea how any of this stuff works, or of what's even possible. What an interesting time!

And this brings me to a story -- that a sponsor of this show -- Sophos ran...
"Did you know you can see the ad boxes Facebook sorts us into?"
https://nakedsecurity.sophos.com/2019/01/18/did-you-know-you-can-see-the-ad-boxes-facebook-sorts-us-into/

https://www.facebook.com/ads/preferences/?entry_product=ad_settings_screen#_=_

http://bit.ly/snfbads (sn fb ads)

"Fitbit? Pollination? Jaguars? Snakes? Mason jars?

OK, fine, Facebook, I'm not surprised that I've clicked on those things. But when did I ever click on anything related to Star Trek: Voyager? Or Cattle?!

My "this feels weird" reaction makes me one of the 51% of Facebook users who report that they're not comfortable that the ad-driven company creates a list that assigns each of us categories based on our real-life interests.

It's called "Your ad preferences." You can view yours here. If you drill down, you can see where Facebook gets its categorization ideas from, including the things we click on or like, what our relationship status is, who employs us, and far more."

Your ad preferences

Learn what influences the ads you see and take control over your ad experience.

Learn about Facebook Ads

- **Your interests**
- **Advertisers**
- **Your information**
- **Ad settings**
- **Hide ad topics**
- **How Facebook ads work**

Ad settings

Close

Ads based on data from partners

To show you better ads, we use data that advertisers and other partners provide us about your activity off Facebook Company Products.

Allowed

Ads based on your activity on Facebook Company Products that you see elsewhere

When we show you ads off Facebook Company Products, such as on websites, apps and devices that use our advertising services, we use data about your activity on Facebook Company Products to make them more relevant.

Allowed

Ads that include your social actions

We may include your social actions on ads, such as liking the Page that's running the ad. Who can see this info?

No One

Was the **ad settings** section helpful for you? Yes No

Cisco's small business switches have SERIOUS problem...
https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-20181107-sbsw-privacc

If you, your organization, or anyone you know are using Cisco 200 or 250 Series Smart Switches, 300 or 350 Series Managed Switches, Cisco 350X, 500 or 500X Series Stackable Managed Switches, there's a REAL problem...

Cisco explains:

Summary

A vulnerability in the Cisco Small Business Switches software could allow an unauthenticated, remote attacker to bypass the user authentication mechanism of an affected device.

The vulnerability exists because under specific circumstances, the affected software enables a privileged user account without notifying administrators of the system. An attacker could exploit this vulnerability by using this account to log in to an affected device and execute commands with full admin rights.

Cisco has not released software updates that address this vulnerability. This advisory will be updated with fixed software information once fixed software becomes available. There is a workaround to address this vulnerability.

What's really going on? Unbelievably, it's another of those Cisco default built-in password nightmares...

The vulnerability (CVE-2018-15439), which has a critical base CVSS severity rating of 9.8, exists because the default configuration of these devices includes a privileged user account that is used for the initial login and cannot be removed from the system. An administrator may disable this account by configuring other user accounts with access privilege set to level 15. However, if no user-configured privilege level 15 accounts exist in the device configuration, the default privileged user account is enabled without notifying administrators of the system.

Cisco says: "Under these circumstances, an attacker can use this account to log in to an affected device and execute commands with full admin rights. It could allow an unauthenticated, remote attacker to bypass the user-authentication mechanism of an affected device."

Actually… no. It allows anyone in the world to USE the user-authentication mechanism since the credentials are documented in Cisco's "Getting Started" documentation.

Naturally, since these network switches are used to manage LANs, this means that a remote attacker would gain access to all local network security functions such as firewalls and the management interface for administering voice, data and wireless connectivity for network devices.

The biggest problem is… as we know, most of these already-deployed devices will never be updated after leaving the factory, leaving their users permanently vulnerable. For those in the know, although there's no workaround yet, just adding at least one user account with access privilege set to level 15 in the device configuration will disable the default admin login credentials.

Miscellany

SyncThing:
Looks like the right solution. I'll know more after I dig into it and I'll shre what I find.

SQRL presentation this coming Sunday to a neat Ethical Hacking group.
Normally, they would welcome more of our listeners who are in the area. But many are already listeners, since the interest expressed by those wishing to attend is currently at 180% of the meeting space capacity, so I'm going to wait until next week to talk about the meeting in more detail. It will be video recorded so I'll be able to share a link once it's ready. And, apparently, based upon the postings to the meeting's sign-up board, I'll be signing many copies of SpinRite!

To prepare, I've been updating the presentation slides that I first created for the DigiCert Security Summit, then updated to demonstrate SQRL to Stina and her technical colleagues at Yubico. The latest slides now add the news of the SQRL Service Provider API, about which I am increasingly excited as I understand how much it will mean for SQRL.

I had never really thought about the server-side implementation until I considered what someone who didn't know SQRL at all would need to know. And while on one hand it's not rocket science, it is all unfamiliar and there are many important subtleties needed to get it right. So the SQRL Service Provider API encapsulates ALL of that to provide a very clean and straightforward RESTful style HTTP query/reply system. And this hides ALL of the details of SQRL's operation to make implementation with the API very straightforward.

The developer in Denmark who knows the XenForo forum software we're using, quickly and easily added SQRL support using the API, and now he's adding the finishing UI touches. And one of the guys in the GRC newsgroups who knows the SQRL protocol inside and out has my assembly language source code for the API and is reimplementing it in portable open source 'C'. As we know, it's one thing for us to be all excited about how it works and all the problems that it solves. But after that, what we most need is for it to actually be adopted. The creation of this SSP API abstracts out ALL of the SQRL protocol management. This means that adding SQRL to any existing website should be a matter of, at most, a few programmer-days work.

Errata

OMG! Pushing past a certificate warning does NOT mean an unencrypted plaintext connection! However is DOES open one to enabled MITM attacks.

Gary Foard / England
Subject: 30 gig
Date: 21 Jan 2019 01:07:10
:
Dear Steve

Love the show and Spinrite....but

Last week (697) you mentioned again as before in 696 about filemail.com and that you have a 30Gig upstream connection. Really? Bloody hell! (In English accent). I get along with 6Mbps downstream and 1 Mbps upstream. Some people have 100Mbps ds which is good. I think I've heard Leo brag of his business connection being 10 Gig but you say 30 Gig Upstream. I don't know how much Spinrite you sell but it ain't that big that you need 30 gig up. You're a nice guy and deserve it but maybe you mean 30 Mbps that would still be impressive. You've said it twice now and it's bugging me. Have a nice day.

SpinRite

Ben asks about having a dedicated SpinRite machine? Wat?
Date: 17 Jan 2019 21:40:09
:
Hello Steve,

Long time listener of Security Now, and all Gibson goodness wherever it is available. Crawler of grc.com. Owner of SpinRite, and recommend it to everyone on the train. Thank you for all of those. I came across an interesting use-case, and wondered how to best do it.

A friend of mine works at a 'photo' shop (that's what they're called here). They do a variety of photo services, as well as a few other digital services. One of them is restoration. Restoring deleted content and/or corrupted media. I visited him at his workplace, and saw that they're running Recuva. Knowing what the answers will be, I asked him what the success rate is. Expectedly, it's so-so. They use several other such utilities, with about the same results. I recommended they buy SpinRite. My friend trusts me, and he convinced the boss to buy it. Success rates are, unsurprisingly, FAR better.

This got me thinking: They could use a dedicated machine for Spinrite. Since SR consumes the whole machine, and they do this often enough, then a machine dedicated for SR makes sense. So, what hardware would this machine feature? From my own experience with SR, I know that it can become very CPU intensive, depending on what it's dealing with on the target media. So, CPU. Lots of CPU power. Cores vs. clock speed? What else? Machine will of course be disk-less (for itself). Is there anything to avoid?

While in "normal" circumstances the machine's hardware wouldn't be a big focus (you start SR and let it run overnight), here it's different. The machine cannot do anything else, and SR does one thing at a time. So giving it the resources to complete its work quicker is the whole point.

I recall that you did say that one can try running SR in a VM and connecting the media to the VM, but you also said that that narrows down SR's range of functionality.
So, what's optimal for SpinRite? If SR can consume the universe, what would it consume? If one had no other considerations, what would that machine look like?

I'd love if you can bring this up in an SN episode.

Would also like to thank to Leo and the heroic team at TWiT as well. Special thanks to Elaine. Her text transcripts come in very handy.

Closing The Loop

Steve Fosmire in Denton, Texas
Subject: Java Update offers to remove itself!
Date: 16 Jan 2019 17:16:19
:
I just got a Java update notification that popped up, so I clicked on update.

This is the text that appeared:

Please remove unused versions of Java
It appears that you have not used Java on your system in over 6 months. We recommend that you uninstall it by clicking the Remove button below. If you later decide you need Java, you can re-install it from java.com.

If you wish to keep Java on your system, please update it by clicking the Update button below.

[Update] [Remove>]

How about that? I don't remember what thing I needed Java for, but to have it tell me "go ahead and remove it" is a new thing entirely. Thought you would want to know. I did take a screen shot of the Java Update window, so if you decide you want to put this in the SN show, I could send it to you. Thanks for over a decade's worth of free security knowledge goodness, Steve.

P.S. proud Spinrite site license (4) owner from my previous IT consultant company (closed up shop a year ago and went to work full time as a network admin at my biggest client.)

Elie in the Bay Area
Subject: Regarding the Chromecast UPnP story
Date: 10 Jan 2019 19:34:33
:
Hi Steve, long time SN listener (you rock, etc., etc.). I run pfsense at home with UPnP enabled on my LAN and IoT VLANs since I'm not too paranoid about anything attacking from the LAN side.

The story about Chromecasts being taken over via UPnP gave me a bit of a scare, admittedly, so I did some additional reading.

It seems that the security community is in agreement that the Chromecasts were NOT commandeered via UPnP, since Chromecasts don't utilize UPnP (Google's [infuriating] support page says that UPnP == multicast...ugh).

Anyway, pfsense shows you a table of UPnP ports that have been opened, and I confirmed that my Chromecast hasn't requested that any ports be opened. From what I read on twitter, it seems that the majority of the vulnerable Chromecasts were located in a specific country/countries, so it could be a certain ISP's routers come with this terrible misconfiguration.

What seems pretty clear though is that UPnP was NOT the way the hackers commandeered the Chromecasts. Cheers!

Which Mobile VPN Client?

"Malware, User Privacy Failures Found in Top Free VPN Android Apps"
https://www.bleepingcomputer.com/news/security/malware-user-privacy-failures-found-in-top-free-vpn-android-apps/

<quote> One in five apps from the top 150 free VPN Android apps in Google's Play Store was flagged as a potential source of malware, while a quarter of them come with user privacy breaking bugs such as DNS leaks which expose user DNS queries to their ISPs.

As found by Simon Migliano, Metric Labs' Head of Research, the company behind the Top10VPN service, these VPN Android applications have already been installed approximately 260 million times according to the numbers reported by Google's official store.

The issues found in the top ten free VPN apps (most installs) on the Google Play store:

App (Installs)	Risky Permissions	DNS Leaks	Risky Functions	Virus / Malware
Hotspot Shield Free (50M)	Detected	No leaks	Detected	No
SuperVPN (50M)	Detected	Leaks	Detected	No
Hi VPN (10M)	Detected	Leaks	Detected	No
Hotspot Shield Basic (10M)	Detected	No leaks	Not detected	No
Psiphon Pro (10M)	Detected	No leaks	Detected	No
Turbo VPN (10M)	Not detected	Leaks	Not detected	No
VPN Master (10M)	Not detected	Leaks	Detected	No
Snap VPN (10M)	Detected	Leaks	Detected	No
Hola (10M)	Detected	Leaks	Detected	No
SpeedVPN (10M)	Detected	No leaks	Detected	No

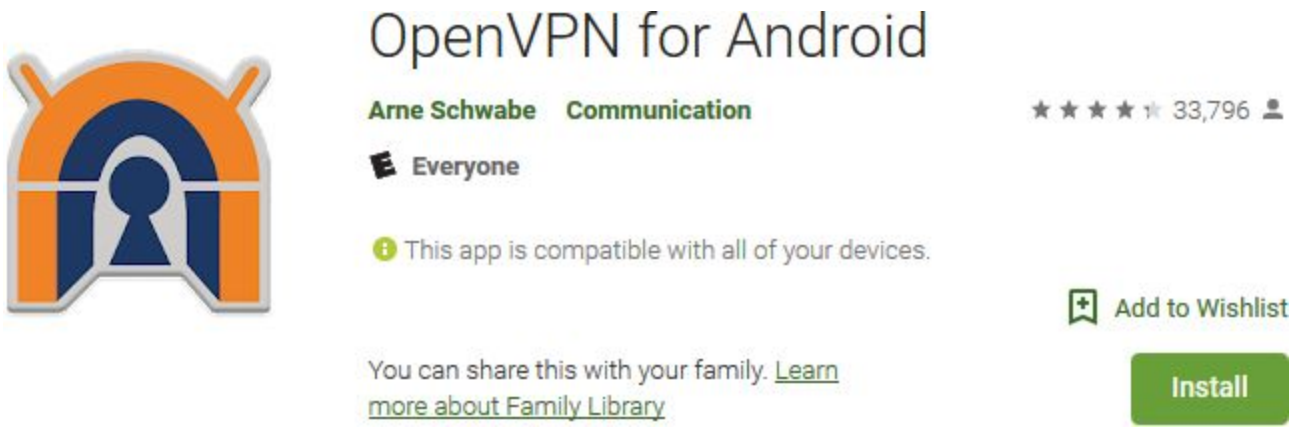
The research team found the following intrusive permissions and user privacy-breaking code:

- Location tracking (25% of apps);
- Access to device status information (38%);
- In smaller numbers: use of camera and microphone and the ability to secretly send SMS.
- Over half (57%) featured code to get a user's last known location.

So… this bring us to the main topic and title of today's podcast:

Which Mobile VPN Client??

OpenVPN for Android
https://play.google.com/store/apps/details?id=de.blinkt.openvpn&hl=en_US



A German developer, Arne Schwabe

5 Million+ downloads

https://github.com/schwabe/ics-openvpn

http://ics-openvpn.blinkt.de/

arne-openvpn@rfc2549.org

https://tools.ietf.org/rfc/rfc2549.txt

- April 1st, 1999 / Category: Experimental
- IP over Avian Carriers with Quality of Service

What's New
- Fix keystore signing on Android 4.1 (yes, really :))
- Fix always visible status from last beta
- Fix default routes not installed with openvpn 3++
- Support ECDSA Certificates from Android keystore, OpenVPN 2.x only
- New OpenSSL Version (1.1.0h)
- Speed test for OpenSSL crypto algorithm
- Experimental support for OpenVPN 3 C++ core
- Implement .api.DisconnectVPN Intent for easy control from other apps
- Better proxy support including Tor(Orbot)
- Bugfixes

Four anecdotal reviews:

- Mike D aka Mike The Nanotecnologist / 3 June 2018
  Fully featured vpn should have dark mode. open vpn connect does. I got a response months ago saying with an update this would come. well... there has been many. i paid for a yr subscription to private tunnel. I am a bit disappointed. Love to see a black or at least a dark theme option. Black out my favorite vpn app please! (black background and white or light grey text would be amazing) saves battery life. cheers

- Sarah Morrigan / 17 January 2019
  This OpenVPN client is very light on system resource consumption, so I would recommend this over all other VPN apps. I've tried several other VPN apps, including the *other* OpenVPN Connect and a few provided by VPN service providers, and I was frustrated by how they terminate itself while I was using a web browser or watching YouTube. Not with this OpenVPN client! I can even multitask and won't lose VPN connection. This is truly a heaven-sent. For those who are looking for free VPN apps here on Google Play, you can use OpenVPN app. Simply download .ovpn files from ProtonVPN (has excellent free service plan), SigaVPN, or VPNBook, import the .ovpn files into this app, and you'll be set to go in a minute.

- Matthew Pottinger / 8 June 2018
  Just set up my first openvpn server on a cloud machine and tried a few other popular opevpn clients and they were super slow for me on the downlink side. from 70mbps down to 30 or even 5mbps... with this client i get pretty much my full internet speed!!! 70 mbps!! awesome! found what i needed

- Some Dude / 5 July 2018
  I'd been using OpenVPN Connect and it just got worse and worse over time. It would cause my phone to slow down over time and eventually become so slow it couldn't recognize my fingerprint in time to unlock the phone before the screen shut off, requiring a hard reboot. It also developed a problem where it would ask me to "continue" or "select certificate" at every connect, which made auto-reconnecting to the VPN after a reboot impossible. I fixed it by installing this client, which is much lower on resources, doesn't kill my phone after a while, and just connects quietly like a VPN client is supposed to do. Good work, please keep it up.