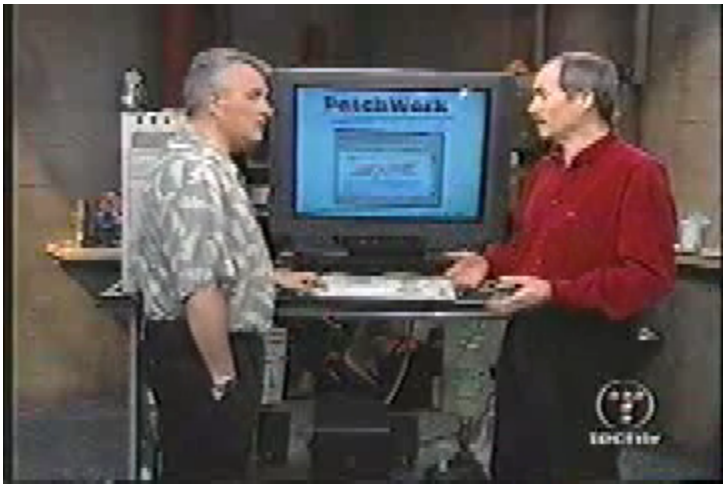
# Security Now! #842 - 10-26-21

# The More Things Change...

## This week on Security Now!

This week we share some welcome news about Windows 11. Leo gets his wish about REvil. Microsoft improves vulnerability report management, attempts to explain their policy regarding the expiration of security updates and prepares for the imminent release of the next big feature update to Windows 10, 21H2. Zerodium publicly solicits vulnerabilities in three top VPN providers. Three researchers disclose their new and devastating "Gummy Browser" attack, which I'll debunk. Another massively popular JavaScript NPM package has been maliciously compromised and then widely downloaded. We close the loop by looking at "Nubeva's" claims of having solved the ransomware problem. We touch on a new annoyance that spreading across websites, and also briefly touch on four SciFi events: Dune, Foundation, Arrival and Invasion. I briefly update on SpinRite. Then we'll take a look back to share and discuss a conversation Leo and I had more than 20 years ago. What's surprising is the degree to which "The More Things Change..." how little (like nothing) actually has.

20+ years ago, Leo and Steve
chatted about security...




It's a bit shocking to hear how very
little has changed in two decades.

# Windows 11 News

**A sneak peak at November 9th upcoming Win11 fixes**

The good news is that Microsoft has jumped right on those pesky Windows 11 problems and will reportedly and hopefully have them fixed up two weeks from now, on November's patch Tuesday. However, anyone who's beset with the problems that next Patch Tuesday's updates are slated to fix may jump the gun and pre-install the update without waiting:

It's packaged as KB5006746 and Googling that magic incantation: "KB5006746" will take you right to it. And I also have a link to it in the show notes:

https://support.microsoft.com/en-us/topic/october-21-2021-kb5006746-os-build-22000-282-preview-03190705-0960-4ba4-9ee8-af40bef057d3

I was going to enumerate the list of things that it fixes, but oh my god... it looks more like the list of things they didn't get in to Windows 11 earlier this month because someone on high said "We shall ship on schedule."

I'm not kidding. The page's list is split into Highlights and Improvements and Fixes. There are 13 highlights listed and 64 improvements and fixes.

I scanned the list and it felt like home. It appeared that everything we've been grumbling about for the past month or two is present and accounted for there. So this fix should go a long way toward dealing with lo those many edge cases.

Specifically, it appears that Microsoft believes that printing is finally fixed. They've confirmed that KB5006746 fixes Windows 11 known issues causing printer installation fails and prompts for admin credentials before every attempt to print on systems commonly found in enterprise environments as I enumerated last week. The trouble with HTTP connections, installation over IPP protocol, and the inability to display custom printing properties should now all be fixed.

The troubles with gaming on AMD chips, which reportedly got worse after this month's patches, are believed to be resolved. And the problems with slow responding Bluetooth mice and keyboards have also been resolved.

Normally I wouldn't recommend that our listeners jump the gun by installing a preview update. But if you already jumped the gun by installing Windows 11, you're obviously afraid of nothing! And once you've scanned through the list of the 77 things that have been highlighted and fixed, this might be another gun worth jumping! And besides, it feels like someone just pressed the "Ship It!" button a bit too soon on Windows 11 and this is just the stuff that didn't make through the door before it slammed closed.

# Ransomware News

**Leo gets his wish!! REvil WAS recently re-taken down by Law Enforcement!**

Last Thursday, Reuters news service exclusively reported that, according to three private sector cyber experts working with the United States and one former official, the ransomware group

**REvil was itself hacked and re-forced offline in a multi-country operation!**

Tom Kellermann, VMWare's head of cybersecurity strategy, and an adviser to the U.S. Secret Service on cybercrime investigations, said law enforcement and intelligence personnel stopped the group from victimizing additional companies. He said: *"The FBI, in conjunction with Cyber Command, the Secret Service and like-minded countries, have truly engaged in significant disruptive actions against these groups. REvil was top of the list."*

As we detailed last week, the new apparent leader of REvil, calling himself "0_neday" who had helped restart the group's operations after an earlier shutdown, said REvil's servers had been hacked by an unnamed party. Recorded Future reported that in a Russian posting they had translated, 0_neday wrote: *"The server was compromised, and they were looking for me. Good luck, everyone; I'm off."*

We now also learn a bit more about what was behind the FBI's deliberate and questionable at the time withholding of the universal decryption key for victims of the Kaseya attacks. Following the attack on Kaseya, the FBI did obtain a universal decryption key that allowed those infected through the Kaseya vulnerability to recover their files without paying a ransom.

The FBI later acknowledged that law enforcement officials had initially withheld the key as they quietly pursued REvil's staff. According to three people familiar with the matter, law enforcement and intelligence cyber specialists were able to hack REvil's computer network infrastructure, obtaining control of at least some of their servers.

Then UNKN disappeared and later 0_neday and a few remaining team members returned and restored those websites from a backup last month. But, in doing so they unwittingly restored and restarted some internal systems that were already controlled by law enforcement.

Oleg Skulkin, the deputy head of the forensics lab at the Russian-led security company Group-IB said: *"The REvil ransomware gang restored the infrastructure from backups assuming that they had not been compromised. Ironically, the gang's own favorite tactic of compromising backups was turned against them."*

Officials have repeatedly declined to comment on the record. A spokesperson for the White House National Security Council declined to comment on the operation specifically. The FBI also declined to comment. But one person familiar with the events said that an unnamed foreign partner of the U.S. government carried out the hacking operation that penetrated REvil's computer infrastructure. And a former U.S. official, who spoke on condition of anonymity, said the operation is still active.

VMWare's Tom Kellermann said that *"The success stems from a determination by U.S. Deputy Attorney General Lisa Monaco that ransomware attacks on critical infrastructure should be treated as a national security issue akin to terrorism."*

In June, Principal Associate Deputy Attorney General John Carlin told Reuters the Justice Department was elevating investigations of ransomware attacks to a similar priority. Tom Kellermann explained that "These actions gave the Justice Department and other agencies a legal basis to get help from U.S. intelligence agencies and the Department of Defense."

"Before [this]", Tom explained, "you couldn't hack into these forums, and the military didn't want to have anything to do with it. Since then, the gloves have come off." In other words, the U.S. military was engaged and apparently getting into the hacker's inner sanctum wasn't so difficult.

So, score one for the big lumbering U.S. bureaucracy. As we have observed before, these crypto cretins need to keep their heads down and under the radar. They made a big mistake when they poked the bear with a sharp stick.

# Security News

**Microsoft: "We're Excited to Announce the Launch of Comms Hub!"**
https://msrc-blog.microsoft.com/2021/10/25/comms-hub/

Yesterday, Microsoft's MSRC blog posted the news of a new Comms Hub, a vulnerability reporoting and researcher portal. Paraphrasing for content, they wrote:

*We are excited to announce the launch of Comms Hub to the Researcher Portal submission experience! With this launch, security researchers will be able to streamline communication with MSRC case SPMs (case managers), attach additional files, track case and bug bounty status all in the Researcher Portal.*

*Currently, security researchers who submit via the portal communicate with MSRC via email. To create a better user experience for the security researcher, we're excited to introduce the Comms Hub feature to the Researcher Portal. With Comms Hub, you will be able to streamline communication with your case SPM, view case status, add file attachments, and track the lifecycle of your case. Comms Hub provides chat functionality allowing asynchronous communications between researchers and the SPM with all the relevant case data readily available all within the Researcher Portal.*

*Please sign in or create an account at MSRC Researcher Portal to submit a vulnerability report. After you have submitted a report, you can use Comms Hub to track and communicate with your SPM in the Researcher Portal. Once you create an account on the MSRC Portal, with the email you used for submission, you will be able to see all of your case submissions.*

They then enumerate a few features of the new Comms Hub...

- When you click on an individual line item from the "All Reports Page" it will take you to the "View Case Details" Page. This displays the timeline of the case, relevant case status information, and allows you to communicate with your case SPM. In this view, you will be able to send in additional POC data, ask and respond to questions to your case manager. Each question will display as individual threads and you will be able to respond to your case SPM's questions.

- The "All Reports Page" shows all the submitted vulnerability reports in a list view. You will be able to view additional case details and communicate to your case SPM when you click into an individual case.

- The “Case Update Glimmer”: When the status of a case is changed and unread, a “glimmer” will show at the top left of the case (row) on the All Reports page and will also be bolded. The “glimmer” + bold will only disappear after the case has been read (opened).

- When you receive a new message from your case SPM, you will be notified via email to visit the MSRC Researcher Portal. When you visit the Researcher Portal, a blue bubble with a white number will be displayed next to the reports with new messages in the ‘My Vulnerability Reports’ page.

We will continue to release updates and new features to better the Comms Hub experience. In the coming months, these are some feature improvements coming to the Researcher Portal:

- Additional notification for case updates in the Researcher Portal
- Comms Hub chat functionality improvements

I think it's all good that Microsoft is working to improve upon and better manage their communication with security researchers who find and report problems with their software offerings. We can also assume, or at least hope, that this new Comms Hub is the public-facing surface of a deeper and significant mechanism for organizing and being responsive to reports of serious defects. For Microsoft, the story of 2021 was, more than anything else, an indictment over their horrifyingly poor response to the known security shortfalls of their products which enabled successful attacks upon many of their own customers. Perhaps there were some serious meetings last spring, after the mishandling of the Exchange Server flaws had become so apparent, with the result being that these new systems have been put in place to prevent a repeat in the future.  Let’s hope.

**Microsoft: “Windows update expiration policy explained”**
https://techcommunity.microsoft.com/t5/windows-it-pro-blog/windows-update-expiration-policy-explained/ba-p/2860928

I encountered an interesting post by Microsoft regarding their policy surrounding the expiration of old updates.

I should preface this by noting that I’ve never had any idea how Microsoft manages the incremental updating of this operating system as well as they do... or at all, for that matter. As someone who builds projects from a large number of smaller files, I'm quite familiar with the idea of dependency trees and dependency resolution. But Windows has become so mind blowingly sprawling that I can't even imagine how they keep the dependency definitions straight.

In any event, it has always seemed to me that there's no point in installing a Windows update if a subsequent update is going to be replacing what the earlier update updated. On the other hand, if you don't wind up installing the subsequent update, or need to later back out of it, that earlier obsoleted update starts looking pretty good.

And speaking of backing out of updates gone wrong, it's one thing to install these things sequentially. That's conceivable. And to later back out of them in strict reverse order. But if you really want your mind blown, think about reaching in and removing some arbitrary update from

the middle of a larger batch... which Windows has always allowed. Watching Windows Update run, I've often noted that the system's mass storage drive spends a lot of time not being in use. In other words, Windows is quite busy thinking. So perhaps individual Windows clients are spending a lot of their own time working out for themselves what to discard and what to roll back. That wouldn't surprise me.

But back to Microsoft's attempt to clarify this. They wrote:

*Microsoft produces two to three updates per supported Windows platform monthly. This results in a backlog of updates and potentially increases the size of update packages. Many of these updates, however, are cumulative and include all earlier updates that have been published for that platform. That means, when older packages expire, you still receive the updates contained in those packages by installing the cumulative update.*

*By expiring older, redundant packages, you get better performance, shorter scan times, a faster user experience, and reduced risk of deploying older updates which have been superseded with newer, more secure ones. Here are answers to common questions we receive about our Windows update expiration policy.* [ It's no surprise that there are questions! ]

***How often are update packages expired?***
*Our published packages are evaluated for expiration on a regular basis. Once a large enough quantity of candidates have been found, an expiration will take place.* [ Which doesn't answer the question they asked themselves. So I suppose the answer is: as often as needed. ]

***Why aren't older updates expired?***
*Some older packages may not yet have been evaluated or may not have met the criteria for expiration.* [ Whatever that means. ] *It is also possible that they have not yet expired because of existing dependencies on that specific update.* [ In other words, we expire older update packages when we want to, and can. ]

***Are there any packages that cannot be expired?***
*Security-only update packages for Windows 8.1, Windows Server 2012 SP2, Windows Server 2012, Windows 7 SP1, Windows Server 2008 R2, and Windows Server 2008 SP2 do not expire as they are not cumulative and hold only one month worth of fixes. Additionally, if a more recent update package has a dependency on an older package, the older package will not expire until it has been superseded by a newer package.* [ So that makes sense. They're saying that until Windows 10, monthly security update packages were not cumulative. They only contained the changes for the current month, thus all previous updates always needed to be installed first. It wasn't until Windows 10 that any single month's security update package could bring any system current. ]

***How can I find out if my update has expired?***
*If an update has expired, you will see the word "EXPIRED" appended to the title of the release note article associated with that specific update on support.microsoft.com.*

**And while we're on the subject of Windows Updates...**
Those who have chosen to remain with Windows 10 will probably be interested in knowing that the next big feature release, known as 21H2, will be rolling out in a few weeks. It's now available to Windows Insiders in the Release Preview Channel.

Microsoft's John Cable, Vice President, Program Management, Windows Servicing, and Delivery explained that "Windows 10, version 21H2 will have a scoped set of features focused on productivity and security, prioritized to meet our customers' needs based on feedback." He said that 21H2 would include:

- WPA3 H2E standards support for enhanced Wi-Fi security.

- Windows Hello for Business introduces a new deployment method called cloud trust to support simplified passwordless deployments and achieve a deploy-to-run state within a few minutes.

- GPU compute support in the Windows Subsystem for Linux (WSL) and Azure IoT Edge for Linux on Windows (EFLOW) deployments for machine learning and other compute intensive workflows.

However, Microsoft recently stated that they were still finalizing the Windows Hello for Business cloud trust deployment method and that it would be subsequently launched in a monthly update.

Once it's out, 21H2 will receive 18 months of support for Home and Pro editions and 30 months for Enterprise and Education editions.

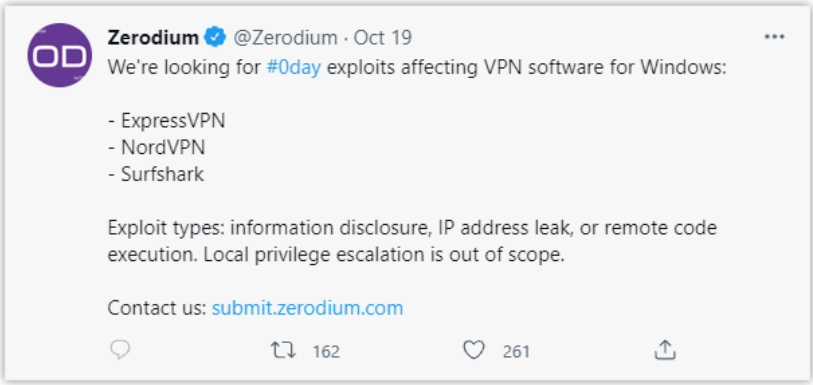
I recall Paul and even MaryJo "Oh-humming" this 21H2 update and being anything but excited. Even to the point of MaryJo asking Paul if there was any there, there? I'm sure everyone who wants to remain current with Windows 10 will want it. But there doesn't appear to be anything new and exciting. And its corners are still quite pointy.

**Windows XP's 20th Anniversary**
At the end of today's podcast we're going to share and discuss a conversation Leo and I had a little more than twenty years ago. That conversation took place on Monday, April 9th, 2001. And it was a little over five months later, exactly 20 years ago yesterday, on September 25th, 2001 that Windows XP was released to the public.

So, Happy Birthday WinXP. And it occurs to me that I've always been grumpy about Windows being changed. I've always wanted Microsoft to please just leave it alone. Fix it, yes; but stop constantly changing it just for the sake of having something new to sell. I recall complaining at the time that they had just taken the very utilitarian and extremely functional Windows 2000 and added a thick candy colored sugar coating to Win 2000's UI.

Oh... and StatCounter agrees with the industry's appraisal of the number of WinXP systems still in use somewhere in the world: 0.59% of all Windows desktops are still running WinXP. That's actually a rather large number. 0.6% or 1 in every 167 desktops — though they may be in use keeping ATMs and kiosks more or less alive.

**Last Tuesday the 19th, Zerodium tweeted:**




We should note that one of these three, ExpressVPN, is currently a sponsor of the TWiT network.

As we know, Zerodium is in the business of reselling software vulnerabilities. They appeared in 2015, headquartered in Washington, DC and we've been following their "exploits" (if you'll pardon the pun) ever since. Their sleazy business model is to purchase exploits for freshly discovered and unknown 0-day vulnerabilities in high profile and often targeted applications — as is the case here — and then compile, catalog and resell those exploits to government and law enforcement agencies.

And what do we imagine these governments and agencies do with those exploits?

On this podcast, we spend a lot of time focusing upon the good guy hackers who participate in public Pwn2Own competitions or who responsibly report their valuable vulnerability findings to a Bug Bounty program, either an independent clearinghouse, or directly to the affected company. All of the major companies pay to learn of responsibly disclosed vulnerabilities in their software. It's become part of what a security-responsible company does.

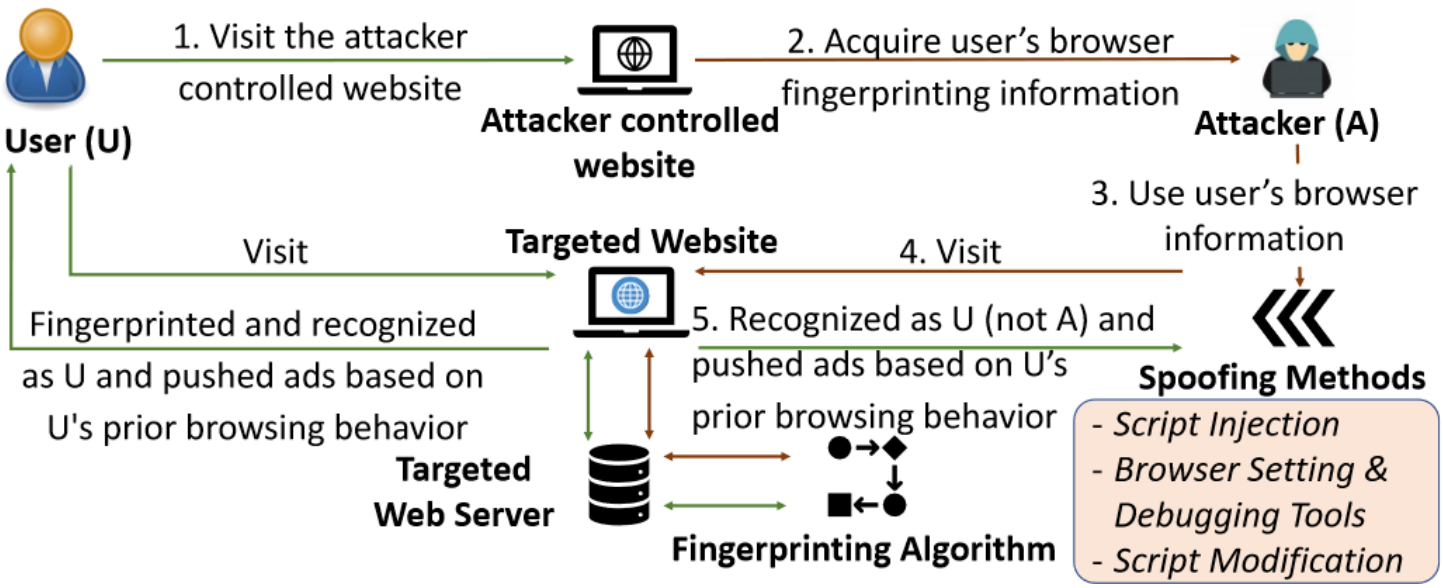
And then there's Zerodium. The fly in the ointment. And they do pay big. Security researchers are encouraged to sell their exploits for up to $2.5 million, depending upon the type and target of their discovery. And, from time to time, Zerodium has launched limited time "bug acquisition drives," during which they express their desire to purchase 0-day exploits in non-standard software. Some previous acquisition drives have targeted routers, cloud services, mobile IM clients, and even something as niche as the Pidgin app — popular with cybercrime organizations.

Major VPN providers, such as the three now being targeted by Zerodium, manage networks of thousands of VPN servers across the globe, rerouting their customers' web traffic to mask their users' physical location, under the premise that where someone is is no one else's business. What's interesting is that these VPN services work with VPN clients residing on any OS platform — Windows, macOS, Linux, Unix, iOS or Android. But Zerodium's solicitation plainly stated that they were only interested in exploits targeting Windows clients, and specifically exploits that can disclose a VPN user's personal information, that can reveal the user's real-world IP address, or exploits that allow remote code execution on the user's computer.

This suggests that there's market among governments and law enforcement for the targeted penetration and determination of the identities of VPN users who are proactively protecting their privacy and identity using the services of these major VPN providers. We know that not everyone who uses a VPN does so merely to geo-relocate themselves for the purpose of accessing locally embargoed media content, or to keep their nosey ISP out of their business. It's certainly the case that criminals also use VPN services to evade law enforcement.

But something still feels very slimy about having an agent of the government and other three-letter agencies — which is exactly what Zerodium is — actively soliciting vulnerabilities in products designed to protect their users.

**The "Devastating" Gummy Browsers attack!**




In preparing each week's podcast, I survey the news of the past week, selecting those items that I think are important and that our listeners should be informed of and/or would enjoy. And I typically skip over dumb things don't merit our time. But in this case I was caught off guard by the exaggerated descriptions of this new and reportedly devastating attack. One of the things that heightened my expectations was that the story was widely picked up across the tech press. So it was with some anticipation that I turned my attention to it to see what was going on for the podcast.

The paper was authored by three researchers, two from Texas A&M University and the other from the University of Florida and it's titled: *"Gummy Browsers: Targeted Browser Spoofing against State-of-the-Art Fingerprinting Techniques"*

Its Abstract reads:

> *We present a simple yet potentially devastating and hard to detect threat called Gummy Browsers whereby the browser fingerprinting information can be collected and spoofed without the victim's awareness, thereby compromising the privacy and security of any application that uses browser fingerprinting. The idea is that the attacker A first makes the user U connect to*

*his website or to a well-known site the attacker controls and transparently collects the information from U that is used for fingerprinting purposes just like any fingerprinting website W collects this information then A orchestrates a browser on his own machine to replicate and transmit the same fingerprinting information when connecting to W fooling W to think that U is the one requesting the service rather than A. As a consequence, if W populates targeted ads for U based on only browser fingerprints A can now start seeing the same or similar ads on his browser as U would see. This will allow the attacker to profile U and compromise U's privacy.*

Okay...

In other words, **if** a website uses advertisers who only employ browser fingerprinting rather than cookies to identify their advertising targets (which is, of course, only very fuzzy identification; it is certainly not unique identification), then it **would** be possible to capture a victim's browser fingerprint by causing their browser to request any asset from an attacker-controlled web server. Then, that attacker-controlled web server could query the original website while deliberately echoing and presenting all of the features of the original browser's query which are fingerprinted by that site's advertisers (assuming that the attacker was also querying and reproducing the identical set of browser fingerprintable features — which is unknowable). And in this way the attacker would be spoofing the website's advertisers into believing that the attacker is actually the user... which, and this is the great headline grabbing concern of these researchers, would then allow them to "profile them" (and that's in air quotes) by seeing which advertisements they are served.

I don't know, Leo. I thought that listening to the audio in a remote room by bouncing a beam of light off a vibrating bag of potato chips was of questionable value. But this one might actually be even less useful. In the words of the authors: "*We present a simple yet potentially devastating and hard to detect threat called Gummy Browsers.*" Devastating? Not so much.

**User-Agent Parser NPM package maliciously altered**

One by one, successive chunks of the technology the world has created for the benefit of everyone, are falling to abuse by bad actors. The trouble is, security is difficult and it's not automatic. Attacks on the software industry's software module supply chain are extremely worrisome because that supply chain was never really secured and there are all manner of ways for bad guys to get their malicious code into the systems of unsuspecting end users and packaging developers. We've discussed a few of these various means of subversion (if you'll pardon the pun) in the past. Like posting something malicious under the name of something real but having a higher version number than the latest real version. Insecure package managers may encounter that higher-numbered malicious package, believe that they no longer have the latest and greatest, and so download the malware for incorporation into their next builds. It's a mess.

So, in that vein, we learned last Friday that a massively popular browser user-agent header parser, named UAParser.js, which is packaged as a JavaScript NPM and is routinely downloaded 6 to 7 **million** times per week, was compromised and, yes, then massively downloaded.

The compromise installs a password stealer, a cryptocurrency miner, and worse on systems where the compromised versions were used. According to the official UAParser.js official site, the library is used by companies such as Facebook, Apple, Amazon, Microsoft, Slack, IBM, HPE, Dell, Oracle, Mozilla, Shopify, Reddit, and many of Silicon Valley's elites.

Compromised versions were: 0.7.29, 0.8.0, 1.0.0
Patched versions: 0.7.30, 0.8.1, 1.0.1

The library's author, Faisal Salman, wrote: *"I believe someone was hijacking my npm account and published some compromised packages (0.7.29, 0.8.0, 1.0.0) which will probably install malware."* Yeah. Probably, indeed. A few hours after discovering the hack, Salman pulled the compromised library versions to prevent users from accidentally infecting themselves and he replaced them with clean copies.

Subsequent analysis of the malicious code revealed extra scripts that would download and execute binaries from a remote server. The binaries were provided for both Linux and Windows platforms. On Friday, a GitHub user said: "From the command-line arguments, one of them looks like a cryptominer, but that might be just for camouflage."

But on Windows systems, the scripts also download and execute an infostealer Trojan, which might be a version of the Danabot malware. According to another GitHub user's findings it contains the capabilities to export browser cookies (thus hijacking logged-on sessions), browser passwords, and OS credentials,

Because of the large number of downloads and the big-name corporations that relied on the library, the US Cybersecurity and Infrastructure Security Agency (CISA) published a security alert late Friday night about the incident, urging developers to update to the safe versions.

GitHub's security team also took note of the incident and issued its own advisory, urging immediate password resets and token rotations from systems where the library was used as part of development processes:

> *Any computer that has this package installed or running should be considered fully compromised. All secrets and keys stored on that computer should be rotated immediately from a different computer. The package should be removed, but as full control of the computer may have been given to an outside entity, there is no guarantee that removing the package will remove all malicious software resulting from installing it.*

And just for the record, this was the 4th malicious npm package found this week. On Wednesday, Sonatype also found three newly-released npm libraries that contained similar malicious code, intended to download and install a cryptocurrency miner, targeting Linux and Windows systems alike.

So, Houston... we have a problem. As I said, we built so much of our world under the assumption that sharing and collaboration would make us all stronger. Without any doubt, it does. But it also opens us to infiltration by those wishing to take advantage of the trust that's inherent in online collaborative efforts.

# Closing the Loop

**Tom Andreas Mannerud / @TAMannerud**
Steve, have you heard of Nubeva before? I just came across them recently and was fascinated by their technology. Seems they are turning the ransomware game on its head. Their technology intercepts and stores the encryption keys used by ransomware so that you can restore your files without needing to pay ransom or restore from backup. https://info.nubeva.com/ransomless_decryption I would be curious to hear what you think?

Okay, so... Interesting. I went looking to see what was up: https://www.nubeva.com/

Nubeva's claim to fame appears to be what they call "TLS Decryption Evolved." The banner on their homepage says: *"Nubeva's patented SKI (Session Key Intercept) software technology delivers a breakthrough solution for modern TLS decryption. SKI decrypts any TLS, with trailblazing price-performance and ease of use. Nubeva licenses SKI to fill growing capability gaps of legacy decryption and simplify operations for inline and passive Cybersecurity and Application Monitoring solutions."*

So, this claim raises all manner of questions, because the entire point of TLS is explicitly to prevent any third party from being able to obtain the communication's session keys. Digging a bit deeper, under their homepage headline of *"See Into Any Session"*, they explain: *"To inspect TLS, each session's shared encryption keys are needed to decrypt. With SKI (Session Key Intercept), Nubeva delivers a reliable, secure, scalable, and non-disruptive means to learn and extract session secrets from servers or clients, at the time of creation via the handshake, and transport for use on authorized decryption functions. After use, keys are destroyed, thus ensuring the highest levels of security. Nubeva licenses software to get keys, securely handle keys, and use keys to decrypt."*

Ah, so now it becomes clear: a *"...means to learn and extract session secrets from servers or clients, at the time of creation via the handshake..."* So, this patented Nubeva technology is not a man in the middle, it's a man deeply embedded into one of the endpoints. From that vantage point it watches the TLS handshake and captures the symmetric encryption key once it's been determined.

Under product details they say: *"Supporting a growing list of platforms and OS including containers, Kubernetes DaemonSet, Windows service, or Native Linux Daemon."* Ah. So, it's not a universal solution. For example under Windows, they've reverse engineered Windows crypto library, where TLS is always negotiated (unless a pesky third party like Firefox brought along their own crypto library). But in any event, they've built a set of hooks into Windows' crypto library so that they can snapshot its working memory to identify and capture any negotiated keys on-the-fly. And they've done something similar for Linux and Kubernetes with <quote> *"a growing list of platforms."* I suppose that's useful, though it's not entirely clear exactly how, since the interception, such as it is, occurs at an endpoint where you also inherently have the pre-encrypted and post-decrypted plaintext directly available. It seems like the hard way to skin that particular cat. This has nothing to do with Ransomware... but it gives us a starting point for understanding their next set of claims: https://info.nubeva.com/ransomless_decryption

On the page that Tom linked to, Nubeva states: *"**NUBEVA for Ransomware** — Universal RansomLess Decryption"* (Sounds Great!! Where do I get some of that??) Then they continue:

> *The RansomLess Decryption is a product development effort by Nubeva to build a revolutionary and systemic solution to this worldwide threat. Not another defense system, not another backup solution, Nubeva enables the reversal of ransomware's encryption with RansomLESS decryption and recovery.*
>
> [ Wow! Sign me up!! ]
>
> *Our solution is an adaptation of our patented Session Key Intercept (SKI) technology, in which Nubeva has perfected the ability to reliably learn and extract the symmetric keys used in Ransomware to encrypt files.*
>
> [ Wait. They've "perfected the ability to reliably learn and extract the symmetric keys used in Ransomware to encrypt files." Hmmm. ]
>
> *We can reliably get keys copies of the keys* [okay, a little typo on their page, there] *right at the moment of encryption, before they are locked with the asymmetric encryption process or exported to command and control servers. And with the file encryption keys available, decryption is simple and immediate. And we can do this not just for old and extinct ransomware variations like many tools on the internet, we can get keys for all modern and historic crypto-ransomware, thus delivering a universal solution.*
>
> [Boy! ... if only only that were true. ]
>
> *When the attackers get through defenses, there is no need for lengthy recovery processes from backups (provided you have them, they are current, and weren't turned off or corrupted by the ransomware too). Instead, simply decrypt and restore without paying, with Nubeva's RansomLess decryption and recovery.*

Their page then shows us three videos, one for REvil, one for AraranLocker/Venus and one for Zariza. In each case they have their solution installed in the system when the ransomware is triggered. And, sure enough, their system captures the encryption keys at the time of encryption.

The end of this page declares: *"Not Another Defense System. Not More Storage or System Back-Up Services. Decrypt Without Paying. You Already Have the Keys."*

*We Get The Keys! Nubeva's core intellectual property, Session Key Intercept, is software that can reliably, efficiently, and securely discover symmetric encryption keys of application processes and services running on computer systems that are used for bulk, symmetric encryption. We have mastered this ability for TLS session keys to enable better, faster, and easier full packet inspection and protection of network traffic. We have proven we can do this for SMB(v3) file-sharing traffic. Now, Nubeva has successfully applied this IP to Ransomware and is working to bring it to market in partnership with the select leaders in security solutions, business, and government.*

Okay. What I believe they have actually shown is that within a specifically constrained environment such as TLS or SMBv3, on specific platforms for which they are designed, they are able to capture the symmetric keys of those processes that they know. So this leaves us with two big questions:

First: *"How generic can this really be?"* It would be entirely possible to reverse engineer a specific piece of ransomware for the purpose of building an interceptor for that specific ransomware. Then, assuming that this Nubeva agent was already running in a system which was subsequently the victim of exactly that same ransomware which Nubeva had been previously trained to observe and intercept, then yeah... it would be just like those videos. So the question is, as I said, *"how generic can this be?"* I think that's to be determined and proven, and there's where I'm exceedingly skeptical. They're claiming that this is an extension of their existing proven and, oh yes, very patented technology. But calling it an extension, I think, stretches the meaning of the term, probably past the point of breaking.

The second big question is: If you have a software agent that's running in a machine which is already paying attention to what's going on and is able to see that some ransomware running in a given process has just generated a symmetric encryption key in preparation for encrypting a file, why not just immediately terminate that process, with extreme prejudice, send an emergency note to corporate IT headquarters, and shutdown the machine? Why would you patiently sit there watching the ransomware go about its dastardly business, frantically recording all of the keys it's using to encrypt, and letting it do so?" (And where are all those keys stored, anyway? Hopefully they don't get encrypted!)

Anyway... Thank you, Tom for pointing us to what really seems like a harebrained idea. It's unclear, even in the case of TLS or SMBv3 interception, what problem is being solved by implanting an agent in an endpoint to capture the negotiated key for the purpose of decryption when that endpoint also contains the plaintext. Why not just get the plaintext?

And it's not at all clear that this technology **can** really be extended beyond that somewhat questionable application and made into a generic symmetric key capture utility. When we had Heartbleed, which was capturing snapshots of RAM and, sure enough, **was** able to discover live server certificates in that RAM, it was able to do so **only** because it was finding certificates. A certificate is rigid and fully-specified highly structured block of data. So it can readily be discovered, with a near zero false positive rate, simply by scanning through RAM. A symmetric key ***has no structure***. As we know, it's just a block of 32 bytes of maximum entropy random binary data that, from the perspective of an outside observer, could be anything. It's only if you know precisely where it is that you could know what it is. Unlike a certificate, nothing identifies it as being a symmetric key. That's why the only way I can see this working is with a highly customized and targeted agent. So, color me skeptical about the ability for this to be made into a generic ransomware solution.

## Miscellany

And completely off topic, Leo... I wanted to take a moment to comment upon a new and annoying behavior I've been encountering on more and more websites and to ask whether you're seeing it too. So, I'm on a site; typically from following a link from a search engine. I look around and don't see what I was looking for. So as I slide my mouse off the page, toward the browser's [BACK] button or toward the page's tab in order to close it, the browser's JavaScript, which has been silently, until now, monitoring my mouse's movements, intercepts my attempt to leave, darkens the screen, and pops-up a "before you leave" or "are you sure you want to leave" intercept. It's happening more and more. It's a bit jarring, and it's really annoying. If it keeps up I wouldn't be surprised to see the browsers start blocking this behavior, or an add-on created to do so.

## SciFi - **Dune / Foundation / Arrival / Invasion**

## SpinRite

I'm heading toward the 5th pre-release of SpinRite to the GRC spinrite.dev newsgroup. I found and fixed the problem I mentioned last week which only affected Intel chipsets, and then only when they were operating in their legacy IDE/ATA mode rather than in their modern AHCI mode. But there was another related problem hiding behind that first one which I'm currently pursuing. Since the bug is intimately timed to specific hardware which I only have here at my primary workplace, in the evenings when I'm not here I've been at work on the improvement to SpinRite's benchmarking which will soon boast a new and very useful feature. So, all's going well!

# The More Things Change...

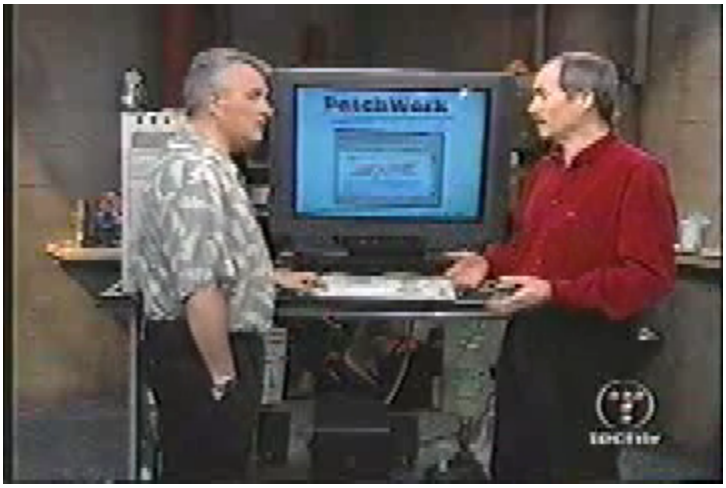
The inspiration for this week's title discussion began with a tweet I received in the late morning last Sunday. Eric, who tweeted publicly from @ELHonline, wrote:

> **Eric / @ELHonline**
> *As a long-time listener of @SGgrc's Security Now podcast, I was delighted to stumble across this gem from 20 years ago! It's so funny how much technology has advanced, and yet so many problems remain the same.*
>
> https://www.youtube.com/watch?v=TqsKyVqYgWc

On the one hand it's easy to say "yeah, there are still security problems." But I hadn't watched that video, which I've also had posted on GRC since the beginning, for many years. Eric's tweet got to spend five and a half minutes watching it again. And when I did, I was frankly astonished to listen to my conversion with Leo, which took place on Monday, April 9th, 2001... and hear just exactly the degree to which we are still right where we were then.

So I want to play this short audio and video into the podcast, to share with our listeners. Then spend a bit of time chatting about the degree to which "the more things change..." The first three minutes, in particular, are surprisingly current, but the whole five and a half minutes will probably be interesting to our listeners, if only to indulge in a bit of nostalgia:

https://media.grc.com/mp4/ss_04_09_01.mp4




Bugs in Internet-connected Windows servers allow Russians to get into those systems to exfiltrate the data they contain, after which those attackers turn around and extort their victims by threatening the disclosure of their customer's personal and private information unless they sign up for the Russian's bogus security services. Sound familiar??