Security Now! #574 - 08-23-16

Routers & Micro Kernels

This week on Security Now!

- Did the Shadow Brokers hack the NSA's Equation Group?
- Apple's bug bounty gets quickly outbid.
- A critical flaw discovered in the RNG of GnuPG.
- The EFF weighs in on Windows 10.
- Chrome browser is frightening people unnecessarily.
- A Johns Hopkins team of cryptographers, including Matthew Green, disclose a weakness in Apple's iMessage technology.
- Some discussion of surprisingly and sadly unused router hardware capabilities, and
- What's a "Micro Kernel"??
- And a Security Now! listener puzzler to ponder for a week...

Security News

Shadow Brokers claims to have hacked NSA-tied hackers, posts exploits as proof

- http://arstechnica.com/security/2016/08/group-claims-to-hack-nsa-tied-hackers-posts-exploits-as-proof/
- "Shadow Brokers" posts 256 megabytes of compressed data -- predominantly batch scripts and unimpressively coded Python -- which is claims to have obtained from "The Equation Group".
- What we know is that it is the data from some advanced hacking entity.
- The Shadow Brokers posting: (as I read exactly what they wrote, see if this doesn't sound like a native English speaker attempting (poorly) to sound like a non-English speaker)
- How much you pay for enemies cyber weapons? Not malware you find in networks. Both sides, RAT + LP, full state sponsor tool set? We find cyber weapons made by creators of stuxnet, duqu, flame. Kaspersky calls Equation Group. We follow Equation Group traffic. We find Equation Group source range. We hack Equation Group. We find many many Equation Group cyber weapons. You see pictures. We give you some Equation Group files, you see. This is good proof, no? You enjoy??? You break many things. You find many intrusions. You write many words. But not all, we are auction the best files.
- The recently dated files are from 2013
- They contain implants, exploits, and other tools for controlling routers and firewalls, including those from Cisco Systems, Juniper, Fortigate, and Chinese manufacturer Topsec.
- A separate analysis from firm Risk Based Security noted that an IP address in an exploit labeled "ESPL: ESCALATE PLOWMAN" contained an IP address belonging to the US Department of Defense.
- Monday’s post of 256 MB of data was offered as a small "taste" of what the Shadow Brokers claim to have acquired.
- The Shadow Brokers’ post offered to auction off the stolen data in exchange for a payment reaching one million Bitcoins ($582 USD/BTC)
- Researchers are skeptical that the group has any hope of selling the data. Those experts speculate the true aim of Shadow Brokers is to discredit and embarrass the US government and its intelligence apparatus.

Apple's "Bug Bounty" program immediately outbid by Exodus Intelligence

- https://www.exodusintel.com/
- As we know, Apple is now offering security researchers up to $200,000 if they privately disclose serious, critical holes in software rather than take such vulnerabilities and exploits elsewhere. However, Exodus Intelligence upped the game last Tuesday by raising Apple's bid, luring researchers with rewards of up to half a million dollars for valid Apple software bugs.
- "Exodus Intelligence", a commercial exploit broker, has launched a "hit list" of the hottest, most wanted exploits for software including Apple iOS, Google Chrome, Microsoft Edge and Adobe Flash. The company will pay $500,000 for the most dangerous bugs in Apple iOS 9.3 and above -- and researchers can choose to take a lump sum or smaller payments which continue to roll in as long as the exploit is still alive.

- Logan Brown, president of Exodus Intelligence, said: "Exodus is excited to be engaging the global research community in our mission to provide the highest quality of vulnerability intelligence in the industry/"
- How can Exodus support such bounties? Exodus offers $200,000/year subscriptions to other companies for comprehensive reports, proof-of-concept, demos, packet captures, and full documentation of these acquired bugs.

Critical RNG Flaw Fixed in GnuPG

- https://lists.gnupg.org/pipermail/gnupg-announce/2016q3/000395.html
- Security researchers discovered a critical vulnerability in the random number generator inside GnuPG and Libgcrypt encryption apps which has been around since 1998 and is present in all version since then. This vulnerability allows an attacker who can arrange to obtain 4640 bits of entropy from the RNG to trivially predict the next 160 bits of output. output from the software's random number generator under some conditions. The mistake was found in the "mixing functions" used by the systems RNG.
- GnuPG v2.1.15 is the fully patched version, thus all Libgcrypt and GnuPG versions released before August 17th, 2016 are affected on all platforms.
- The researchers indicated that although the vulnerability is critical, users should not immediately start revoking private keys created with vulnerable versions.
- <quote> "A first analysis on the impact of this bug in GnuPG shows that existing RSA keys are not weakened. For DSA and Elgamal keys it is also unlikely that the private key can be predicted from other public information. This needs more research and I would suggest _not to_ overhasty revoke keys."

EFF: With Windows 10, Microsoft Blatantly Disregards User Choice and Privacy

- https://www.eff.org/deeplinks/2016/08/windows-10-microsoft-blatantly-disregards-user-choice-and-privacy-deep-dive
- Steve's Tweet: Win10 holdouts (not to mention corporations) would likely appreciate the EFF's take on choice and privacy tradeoffs.

- INTRO: Microsoft had an ambitious goal with the launch of Windows 10: a billion devices running the software by the end of 2018. In its quest to reach that goal, the company aggressively pushed Windows 10 on its users and went so far as to offer free upgrades for a whole year. However, the company's strategy for user adoption has trampled on essential aspects of modern computing: user choice and privacy. We think that's wrong.

  You don't need to search long to come across stories of people who are horrified and amazed at just how far Microsoft has gone in order to increase Windows 10's install base. Sure, there is some misinformation and hyperbole, but there are also some real concerns that current and future users of Windows 10 should be aware of. As the company is currently rolling out its "Anniversary Update" to Windows 10, we think it's an appropriate time to focus on and examine the company's strategy behind deploying Windows 10.

- Disregarding User Choice
  - Chronicles all of the various ways MSFT pushed (hard) to force everyone to Windows 10.

- Disregarding User Privacy
    - INTRO: The trouble with Windows 10 doesn't end with forcing users to download the operating system. Windows 10 sends an unprecedented amount of usage data back to Microsoft, particularly if users opt in to "personalize" the software using the OS assistant called Cortana. Here's a non-exhaustive list of data sent back: location data, text input, voice input, touch input, webpages you visit, and telemetry data regarding your general usage of your computer, including which programs you run and for how long.

    - << talks about the options to reduce the reporting >>

    - Unless you're an enterprise user, no matter what [settings you choose], you have to share at least some of this telemetry data with Microsoft, and there's no way to opt-out of it. Microsoft has tried to explain this lack of choice by saying that Windows Update won't function properly on copies of the operating system with telemetry reporting turned to its lowest level. In other words, Microsoft is claiming that giving ordinary users more privacy by letting them turn telemetry reporting down to its lowest level would risk their security since they would no longer get security updates. (Notably, this is not something many articles about Windows 10 have touched on.)

    - CONCLUDING: There's no doubt that Windows 10 has some great security improvements over previous versions of the operating system. But it's a shame that Microsoft made users choose between having privacy and security.

- Never10: >1,865,000 (4200/day)

Google Chrome's ongoing SHA-1 scare tactics:

- Via Twitter: Steve - I work for a hospital and we have an online bill pay system in place using a third party site. When visiting the payment site in Chrome, and looking at the certificate information, Chrome is telling me that the security on the site is weak. (See image) What are your thoughts on this?
- Chrome was warning of an SHA-1 cert, claiming that the connection might not be secure.
- That's utter nonsense, and Google's security engineers know it.
- This is Google doing the good work of pushing an always reluctant industry forward.
- Unfortunately, nuance works less well than inspiring fear with a blunt instrument.

Dancing on the Lip of the Volcano: Chosen Ciphertext Attacks on Apple iMessage

- https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/garman
- Matthew Green and four other Johns Hopkins researchers
- ABSTRACT:
  Apple's iMessage is one of the most widely-deployed end-to-end encrypted messaging protocols. Despite its broad deployment, the encryption protocols used by iMessage have never been subjected to rigorous cryptanalysis. In this paper, we conduct a thorough analysis of iMessage to determine the security of the protocol against a variety of attacks.

  Our analysis shows that iMessage has significant vulnerabilities that can be exploited by a sophisticated attacker. In particular, we outline a novel chosen ciphertext attack on Huffman compressed data, which allows retrospective decryption of some iMessage payloads in less than 218 queries. The practical implication of these attacks is that any party who gains access to iMessage ciphertexts may potentially decrypt them remotely and after the fact. We additionally describe mitigations that will prevent these attacks on the protocol, without breaking backwards compatibility. Apple has deployed our mitigations in the latest iOS and OS X releases.

- The team DID responsibly disclose, reporting to Apple in November of 2015.
- Apple revised their protocols in backward-compatible fashion in iOS 9.3 and Max OSX v10.11.4, both which were available in March 2016.

- High Level Protocol Analysis
    - Key server and registration
      iMessage key management uses a centralized directory server (IDS) which is operated by Apple. This server represents a single point of compromise for the iMessage system. Apple, and any attacker capable of compromising the server, can use this server to perform a man-in-the-middle attack and obtain complete decryption of iMessages. The current generation of iMessage clients do not provide any means for users to compare or verify the authenticity of keys received from the server. Of more concern, Apple’s “new device registration”mechanism does not include a robust mechanism for notifying users when new devices are registered on their account. This mechanism is triggered by an Apple push message, which in turn triggers a query to an Appleoperated server. Our analysis shows that these protections are fragile; in Appendix A we implement attacks against both the key server and the new device registration process.

    - Lack of forward secrecy
      iMessage does not provide any forward secrecy mechanism for transmitted messages. This is due to the fact that iMessage encryption keys are long-lived and are not replaced automatically through any form of automated process. This exposes users to the risk that a stolen device may be used to decrypt captured past traffic.

      Moreover, the use of long term keys for encryption can increase the impact of other vulnerabilities in the system. For example, in §5, we demonstrate an active attack on iMessage encryption that exposes current iMessage users to decryption of past traffic. The risk of such attacks would be greatly mitigated if iMessage clients periodically generated fresh encryption keys. See §7 for proposed mitigations.

    - Replay and reflection attacks
      The iMessage encryption protocol does not incorporate any mechanism to prevent replay or reflection of captured ciphertexts, leading to the possibility that an attacker can falsify conversation transcripts as illustrated in Figure 2. A more serious concern is the possibility that an attacker, upon physically capturing a device, may replay previously captured traffic to the device and thus obtain the plaintext.

    - Lack of certificate pinning on older iOS versions
      iMessage clients interact with many Apple servers. As of December 2015, Apple has activated certificate pinning on both APNs (Apple Push Notification) and ESS/IDS connections in iOS 9 and OS X 10.11. This eliminates a serious attack in which an MITM attacker who controls the Sender's local network connection and possesses an Apple certificate can intercept calls to the ESS/IDS key server and substitute chosen encryption keys for any Recipient (see Appendix A for further details). We note that devices running iOS 8 (and earlier) or versions of OS X released prior to December 2015 may still be vulnerable to such attacks. For example, at the time of our initial disclosure in November 2015 to Apple, pinning was not present in OS X 10.11.

    - Non-standard encryption
      iMessage encryption does not conform to best cryptographic practices and generally seems ad hoc. The protocol (see Figure 1) insecurely composes a collection of secure primitives, including RSA, AES and ECDSA. Most critically, iMessage does not use a proper authenticated symmetric encryption algorithm and instead relies on a digital signature to prevent tampering. Unfortunately it is well known that in the multi-user setting this approach may not be sound [21]. In the following sections, we show that an on-path attacker can replace the signature on a given message with that of another party. This vulnerability gives rise to a practical chosen ciphertext attack that recovers the full contents of some messages.

- Attack Overview
  There are two stages of the attack. The first exploits a weakness in the design of the iMessage encryption composition: namely, that iMessage does not properly authenticate the symmetrically encrypted portion of the message payload. In a properly-designed composition, this section of the ciphertext would be authenticated using a MAC in generic composition or via an AEAD mode of operation. Apple, instead, relies on an ECDSA signature to guarantee the authenticity of this ciphertext. In practice, a signature is insufficient to prevent an attacker from mauling the ciphertext since an on-path attacker can simply replace the existing signature with a new signature using a signing key from an account controlled by the attacker. In practice, the actual attack is slightly more complex; the first phase includes additional operations to defeat a countermeasure in the decryption mechanism, which we discuss below.

- Takeaways… The clear and present danger of Closed Protocol Security Design.

The Elegance of Deflate (via Leo & several others)

- http://www.codersnotes.com/notes/elegance-of-deflate/
- Our podcast #205 (July 16th, 2009) was titled "Lempel & Ziv".
- LZ is a WONDERFUL inspired invention and we had great fun covering and explaining it in detail.
- SMG loves compression. The LRS project was an offshoot of a compression idea.

SpinRite

Following up on running SpinRite in several sessions...

Router Switches

- http://www.deyisupport.com/cfs-file.ashx/__key/telligent-evolution-components-attachments/00-25-01-00-00-20-73-71/QCA8337N_5F00_Data_5F00_Sheet_5F00_MKG_2D00_17793_5F00_v1.0.pdf#page46
- http://wiki.mikrotik.com/wiki/Manual:Switch_Chip_Features
- http://wiki.mikrotik.com/wiki/Manual:TOC
- https://wikidevi.com/wiki/MediaTek_MT7621
- https://wiki.openwrt.org/toh/ubiquiti/ubiquiti_edgerouter_x_er-x_ka
- http://dd-wrt.com/wiki/index.php/Iptables#Deny_access_to_a_specific_Subnet
- http://dd-wrt.com/wiki/index.php/Switched_Ports#Separate_LAN_ports_.28into_another_subnet.29
- http://dd-wrt.com/wiki/index.php/Default_internal_device_network
- http://dd-wrt.com/wiki/index.php/Firewall

Micro Kernels

- The evolution of operating systems:
- Before the concept of a "supervisor" programs ran on the "brare metal"
    - Each "job" took over the entire machine.
    - While that program was running, the machine was completely tied up.
    - But these machines were hideously expensive.
    - There was HUGE pressure on the operators to keep the "jobs" running.
        - "Have you run my job yet?"
        - When a job would "ABEND" (abnormal end) a dump would be printed out for the job's programmers and the next job would be started.
    - A computer room might have ten tape drives and two types of printers. But if a job was running that only used three of the tape drives... all of the other (very expensive) hardware sat idle.

- What then evolved was "time sharing" where ONE big expensive machine was simultaneously SHARED among many users. No user had the instantaneous experience of having the machine all to themselves. You would often hear from a remote terminal operator: "Why is the system running so slowly today?" And not all users had equal priority on the machine.
- But it was a WIN for the machine's OWNERS, because this kept the expensive machine running continually.
- When I was working at Stanford's AI Lab and begged Les Earnest, the administrative liaison for access to their PDP-10 and 16 machines, he agreed, but said I could only use it in the evenings and weekends since it was too loaded down during the day.
- When I was at Berkeley, the computer science department had a CDC6600 and a 7600. We would submit card decks and come back hours later to find our deck and its printout in a "cubby hole."

- What is a Microkernel?
    - "No battle plan survives contact with the enemy."
        - Some things cannot be done by apps:
            - Program Loader
            - Scheduler (process & thread)
        - System resources are global and shared:
            - Memory management
    - ---
    - OS API services?
    - Device drivers?
    - File system abstractions?
    - Higher level functions? e.g. graphics

Puzzler for next week:

- @SGgrc I've often heard you talk about cryptography and prime numbers, but why can't non-prime numbers be used??