Security Now! #621 - 07-25-17

Crypto Tension

This week on Security Now!

We start off this week with a fabulous picture of the week and for the first time in this podcast's 12-year history, our first quote of the week. Then we'll be discussing the chilling effects of arresting ethical hackers, the upcoming neutrality debate congressional hearing, something troubling I encountered at McAfee.com, an entirely new IoT nightmare you couldn't have seen coming and just won't believe, the long-awaited Adobe Flash end-of-life schedule, welcome performance news for Firefox users, the FCC allocates new sensor spectrum for self driving cars, three bits of follow-up errata, a bit of miscellany, and then: "Crypto Tension" -- A careful look at the presently ongoing controversy surrounding the deliberate provisioning of passive eavesdropping decryption being seriously considered for inclusion in the forthcoming TLS v1.3 standard. ***But first, a word from one of our several loving podcast sponsors...***

Quote of the Week

Todd Westby, the CEO of Wisconsin-based tech company Three Square Market, was explaining to a reporter from ABC News about how 50 of the company's 80 employees had agreed to be "chipped" -- implanted with an RFID tag the size of a grain-of-rice.

"There's really nothing to hack in it because it is encrypted just like credit cards are. ... The chances of hacking into it are almost nonexistent because it's not connected to the internet," he said. "The only way for somebody to get connectivity to it is to basically chop off your hand."

He also said that his wife, young adult children and others would be getting the microchip next week.

(What could POSSIBLY go wrong??)

Security News

Budapest's Hungarian public transportation authority (BKK) turned-in an 18 year old ethical hacker after he notified them of a trivial exploit of their online ticketing website.
https://www.bleepingcomputer.com/news/security/45-000-facebook-users-leave-one-star-ratings-after-hackers-unjust-arrest/

Get a load of this hack: The 18 year old (who has asked that his name not be made public) was poking at a newly available mobile online ticketing system. He discovered that after bringing up the BKK's website, he could simply press F12 to open the browser's built-in developer tool, modify the page's form submission code to alter a ticket's price... and because there was absolutely NO client or server-side pricing validation in place, the BKK system accepted the visitor-provided ticket price and issued a valid ticket at the reduced price.

As a demo, the young man says he purchased a ticket, normally priced at an equivalent of about $35 for just 20 cents. It was a simple proof of concept and he never used the ticket in any way. And, of course, when he initially made the trivial change to the web page he had no idea whether it would even work -- and, as we know, it's disgusting security design that it did work.

After responsibly notifying the transportation authority of his finding so that they could address this glaring deficiency in their system, and never even using the valid ticket, shortly afterward he was awoken in the middle of the night and arrested by police.

BKK management then boasted in a press conference about "catching the hacker" and declaring their systems "secure." BleepingComputer reports that since then, other security flaws in BKK's system have surfaced on Twitter. (Yeah, no kidding... I would hate to be the BKK right now if their publicly-facing system's design is SO slip shod. Can you even imagine what else must be wrong there?

And get this: The BKK has a $1 million dollar annual contract with a local company, T-Systems, for the maintenance of its IT systems.

And, since then additional rookie mistakes have come to light:

- The system stores passwords in clear text and eMails them in the clear if you ask for a password reminder.

- After logging in, visitors were also able to get the data of other users (apparently through manipulating the page's URL. Reports claim that it's possible to access other user's profiles which includes the user's full name, address and an ID number -- either a national id, driving license or passport number.

- If the URL (shop.bkk.hu) was entered into the browser nothing would happen because the site never implemented an HTTP to HTTPS redirect to bounce browsers' default assumption of HTTP over to HTTPS. So someone hearing of the URL and entering it would be out of luck.

- The tickets do not display properly in the iPhone's Safari browser.

- Someone found out that the admin password was "adminadmin" and logged in using that.

There was no tracking of ticket usage or cancellation after use. So the tickets were 100% copyable and there were reports of a few guys making a video showing the reuse of the same ticket though "ticket control" 10 out of 10 times without being caught or raising any alarm.

BKK representatives talked about how the system was under continuous attacks, of which none were successful, that there was no need to stop the system, and that everyone's data is safe. (Nothing to see here, these are not the droids you are looking for, move along.)

We KNOW that criminal cyberhacking is a very real thing today. Bad guys won't disclose the breaches they find, they'll exploit them to the limit... doing real damage over time to their victims.

Compare that to a teenager who verifies a problem and quietly, privately and responsibly reports it to the affected company, thus allowing them to fix the problem with no fanfare.

This IS a difficult problem because there IS a wide grey area between white and black hats. But the initially overbroad laws which are still on the books that label anything some authority in power doesn't like, as "cyberhacking" with arrests and worse NEED to be fixed. It's SO CLEAR that enterprises and governments are only damaging themselves by attacking those who are attempting to help them prevent attacks.

Whether or not ISPs likes it, Title II is still in effect, and Verizon was recently caught deliberately violating it.
Verizon claims that they were testing performance optimizations for video content on their network (whatever that means). Actually I think we know what that means -- slow down video. And the effect was clear: Netflix, YouTube and other video streaming services were being throttled. This was not disclosed by Verizon until after it was discovered and became public.

Users achieving nearly 30 Mbps on an LTE connection were measuring a reduced Netflix data rate of 10 Mbps.

Using a VPN during this time unthrottled the connection, since Verizon was then unable to peer into the traffic, did not know that it was coming from Netflix or YouTube, and could not throttle it based upon the data's source.

As we covered last week, one of the major ISPs said that they wanted their 2015 Title II classification as common carriers repealed, and that when that was done they would honor net neutrality voluntarily. Yet here we appear to have Verizon breaking net neutrality while still under Title II which makes that unlawful. What possible hope is there if ISPs are released from the legal obligation to treat all traffic equally?

ISPs don't want user to use VPNs, but if content- and source-based traffic shaping is applied to consumer data streams an increased use of VPNs is foreseeable.

Net neutrality faceoff: Congress summons ISPs and websites to hearing
https://arstechnica.com/tech-policy/2017/07/facebook-alphabet-amazon-and-netflix-called-to-testify-on-net-neutrality/

Jon Brodkin reporting for ArsTechnica:
The biggest websites and the biggest Internet service providers are being summoned to Congress to testify about net neutrality.

The chair of the House Energy and Commerce Committee, US Rep. Greg Walden (R-Ore.), said he is scheduling a full committee hearing titled, "Ground rules for the Internet ecosystem," for Thursday, September 7.

During an FCC oversight hearing this (Tuesday) morning, Walden Quote: "Today I'm sending formal invitations to the top executives of the leading technology companies including Facebook, Alphabet, Amazon, and Netflix, as well as broadband providers including Comcast, AT&T, Verizon, and Charter Communications, inviting each of them to come and testify before our full Energy and Commerce Committee.

The question, of course, is whether this is just political theatre and whether the lobbyists have already won and testimony is only being taken for face-saving purposes.

And, as we said here last week, for better or for worse, what we need here is clear and clean law rather than the vicissitudes of Presidential appointee mandates.

Greg Walden agrees. He wants Congress to step in and said both ISPs and websites should weigh in first. He said: "It's time for Congress to legislate the rules of the Internet, and stop the ping-pong game of regulations and litigation. Given the importance of this public policy debate and the work we need to do as a committee, it is essential that we hear directly from the country's top Internet and edge provider leaders who frequently speak out publicly about rules of the Internet. It's time they came before us and directly shared their positions and answered our questions. With more than a month's advance notice, I'm sure they can arrange their schedules to accommodate our invitations."

(Thank goodness it falls on a Thursday and not a Tuesday... because this will be a day of Congressional testimony I'm not going to want to miss!)

My browser complained when visiting the McAfee website

A quick update on McAfee Corporation:
TechCrunch had a nice piece of reporting at the start of this past April:
If you were on the internet in a certain era, you remember McAfee. It was the defensive line between you and the rest of the internet, reminding you with incessant popups that you were not hacked, not quite yet, but only if you renewed your subscription right away. Then Intel bought the firewall company in 2010 for an eye-popping $7.68 billion and billed it as Intel Security, and the name McAfee became more closely associated with the company's founder, a man who retired to Belize only to be accused of his neighbor's murder. (Johnny Depp will reportedly play John McAfee in an upcoming film.)

But things didn't work out with Intel (or Belize, for that matter) and so the unit formerly known as Intel Security will be McAfee once again. Today, Intel is officially inking a deal that will spin McAfee out, with the asset management firm TPG taking a 51 percent stake in the company at a $4.2 billion valuation. Intel will retain a 49 percent stake.

McAfee currently secures two-thirds of the world's 2,000 largest companies and grew its revenue 11 percent in the first half of 2016.

So… Yesterday I go to a page at McAfee.com -- McAfee.com, the security firm -- and I am greeted by an across-the-page fixed-position floating bar text message stating that: "Your browser is blocking some features of this website. Please follow the instructions at http://support.heateor.com/browser-blocking-social-features/ to unblock these.

HEATEOR:
"Sassy Social Share, Super Socializer WordPress"
Headline: Why is My Browser Blocking Social Features of the Webpage?
March 17, 2017

Your browser might be blocking Social Features of the webpage you are facing issues with, related to loading social content. [**Oh no!**] If you are using Mozilla Firefox browser [Yes] and it has Tracking Protection feature enabled [Of course], you may have issues in getting content loaded from Social Media websites, such as – Facebook, Twitter etc. These features include Social Share Counts, Social Avatars, Social Comments and Social Login. [Yes!... all those social things we so desperately need from our McAfee Enterprise Security provider.]

To get the social content unblocked, you need to disable Tracking Protection of Firefox by following the steps mentioned below:

- Open a new tab and type about:config in the Firefox Location bar. Press Enter.
- The about:config "This might void your warranty!" warning page may appear.
- Click the "I accept the risk" button to continue to the about:config page.
- Search for trackingprotection
- Double-click privacy.trackingprotection.enabled to set its value to FALSE.

This is the site that the McAfee Enterprise Security Company website's pop-up interruptive text banner diverted to me. Perhaps John McAfee is still around?

Here's a headline that just takes your breath away:
Roomba Maker Preparing to Sell Maps of Your Home to Advertisers
https://www.bleepingcomputer.com/news/technology/roomba-maker-preparing-to-sell-maps-of-your-home-to-advertisers/

Yesterday, iRobot's CEO, Colin Angle announced plans to sell maps of users' homes to advertisers.

In 2015, iRobot started selling Roomba models capable of mapping homes, so the vacuums would know where they should go, and stop bumping into furniture or other things. Until now, these maps have been kept and used only internally on the device to aid its navigation and understanding of its environment. But iRobot realized there was a monetization possibility there and now plans to upload the maps of its customer's homes to its servers, from where they will be sold to online advertisers like Amazon, Apple, or Google.

BleepingComputer reports that the primary buyers aren't regular ad companies, but makers of smart home voice assistants, like Amazon (Alexa), Apple (Siri), and Google (Home). The idea is that these companies could buy this data and combine it with the telemetry they already obtain from their devices and build more sophisticated user profiles that they, in turn, can then sell down the road to classic advertising companies, or offer advertising inside their products.

iRobot's CEO told Reuters in an interview: "There's an entire ecosystem of things and services that the smart home can deliver once you have a rich map of the home that the user has allowed to be shared." Colin Angle did tell Reuters that no user data would be sold without permission.

BleepingComputer's reporting said: In spite of the bold privacy-intrusive business strategy, Angle is well aware that some users won't like the company's direction. This is why Angle said they won't share any data unless users agree beforehand.

However... We all know about "The Tyranny of Default" and that iRobot's desire will be to obtain as much "permission" as possible. So it would be wise, if you have a Roomba smart vacuum roaming around your home and are concerned about the privacy of your environment, to keep an eye on any iRobot privacy or ToS updates in the near future.

Adobe's Flash End-Of-Life is finally scheduled.
2020 cannot come soon enough!

Google: So long, and thanks for all the Flash
- https://blog.chromium.org/2017/07/so-long-and-thanks-for-all-flash.html

MaryJo for ZDNet
- http://www.zdnet.com/article/microsoft-commits-to-eliminating-flash-support-in-windows-by-2020/

I, of course, refuse to run FLASH. It's one thing to require it for video playback, though it's been possible to play video with pure HTML on all browser for so long that there's really no excuse for requiring it. GRC's videos have been Flash-Free for years.

But I'm also confounded by non-video sites that have "FLASH helpers" of one sort of another where my browser complains that a page is apparently attempting to run flash. I just say no and never explore it further... but the MOST likely scenario is that an embedded advertisement is the culprit and accepting a third party advertiser's FLASH script is the LAST THING you want to do!! As we have too often covered here though the past 12 years of this podcast, "FLASH ads" have been the primary "malvertising" infection vector.

But Flash's sun is finally setting. This morning, Google's Chromium team blog headine was: "So long, and thanks for all the Flash" (paying homage to the 4th book in Douglas Adam's wonderful Hitchhiker's Guide to the Galaxy trilogy.)

Quote: "This morning, Adobe announced their plans to end support for Flash in late 2020. For Flash developers this will mean transitioning to HTML, as Chrome will increasingly require explicit permission from users to run Flash content until support is removed completely at the end of 2020. HTML is faster, safer, and more power efficient than Flash and works across desktop and mobile. Three years ago, over 80% of Chrome daily desktop users visited sites with Flash. Today only 17% of users visit sites with Flash and we're continuing to see a downward trend as sites move to HTML. We strongly encourage sites that still rely on Flash to make the move to HTML as there will be an increasing number of restrictions on Flash leading up to the end of support. MaryJo writes: Adobe finally has drawn a line in the sand, noting that Flash will no longer be

supported after 2020. Microsoft officials said they'd do their part to wind down Flash support in the company's Internet and Edge browsers, so that Flash support will be entirely removed from Windows by the end of 2020, as well.

Flash in Edge already is only click-to-run, as of the Windows 10 Creators Update. Today, Microsoft posted its timeline and plan for getting rid of Flash over the next three years.

From Microsoft's post:

- Through the end of 2017 and into 2018, Microsoft Edge will continue to ask users for permission to run Flash on most sites the first time the site is visited, and will remember the user's preference on subsequent visits. Internet Explorer will continue to allow Flash with no special permissions required during this time.

- In mid to late 2018, we will update Microsoft Edge to require permission for Flash to be run each session. Internet Explorer will continue to allow Flash for all sites in 2018.

- In mid to late 2019, we will disable Flash by default in both Microsoft Edge and Internet Explorer. Users will be able to re-enable Flash in both browsers. When re-enabled, Microsoft Edge will continue to require approval for Flash on a site-by-site basis.

- By the end of 2020, we will remove the ability to run Adobe Flash in Microsoft Edge and Internet Explorer across all supported versions of Microsoft Windows. Users will no longer have any ability to enable or run Flash.

Google, Mozilla and Apple also are committing to dropping Flash support by 2020 in their respective browsers.

Firefox is still in the running for users who enjoy organizing with tabs:
Although June's Netmarketshare shows Firefox commanding only a 12% share of the browser universe -- whereas Chrome has nearly 60% of the market at 59.5%.

Mozilla's "Quantum Flow" project is bearing fruit in the next release number 55 of Firefox. When loaded down with a massive test-case of 1,691 open tabs, the current version of Firefox (54) required over four minutes to start and consumed 2GB of system memory. By comparison, FireFox 55 with Quantum Flow started the same daunting test-set of tabs in just 15 seconds and consumed less than half a gig of RAM.

http://www.techradar.com/news/firefoxs-blazing-speed-with-huge-numbers-of-tabs-leaves-chrome-in-the-dust

Next-gen car technology just got another big upgrade
https://www.washingtonpost.com/news/innovations/wp/2017/07/13/next-gen-car-technology-just-got-another-big-upgrade/

The FCC has just approved a sizeable new chunk of radar spectrum for use by vehicular environment sensing radar. This will enable the use of reduced cost and increased precision sensors in our next generation autos.

As we know, many consumer vehicles already use radar for collision avoidance, automatic lane-keeping and other purposes. But right now (the Washington Post writes), vehicle radar is divided into a couple of different chunks of the radio spectrum. Last Thursday, the Federal Communications Commission voted to consolidate these chunks — and added more, to allocate additional bandwidth to vehicle radar.

FCC commissioner, Clyburn said: “While we enthusiastically harness new technology that will ultimately propel us to a driverless future, we must maintain our focus on safety — and radar applications play an important role.”

And in an extremely wonderful pun, the Washington Post's Brian Fung wrote: Thursday’s decision by the FCC lets vehicle radar take advantage of the spectrum ranging from 76 GHz to 81 GHz — reflecting an addition of four extra gigahertz — and ends support for the technology in the 24 GHz range.

I'll note that the increase in radar frequency is very significant. From an engineering standpoint, the move from 24 Ghz to a 76-81 Ghz band significantly increases the resolving power of the radar. And the higher frequency means smaller and more efficient devices. We're all familiar with audio speakers, tweeters and woofers. We know that to produce low audio frequencies in free air requires a large diameter speaker cone -- commonly known as a woofer. But that higher frequencies can be efficiently generated with smaller speakers. This analogy holds at microwave frequencies were this factor of three jump will allow for many more array sensors within the same area. This is a bit win both for radar assist and future full autonomous driving.

Humble Book Bundle -- 6 days remaining at 11am Pacific this morning

- http://bit.ly/sn-621

- The "Humble Bundle Downloader:
    - https://github.com/diogogmt/humblebundle-downloader

Errata

An Amazon Echo Can't Call the Police—But Maybe It Should
https://www.wired.com/story/alexa-call-police-privacy/

Wired Magazine followed-up on what turned out to be a widely reported but erroneous story about an unnamed home audio device (originally believed to be a Google Home unit than an Amazon Echo) autonomously responding to a very loud and fraught domestic dispute by phoning the police.

Except... that never happened.

WIRED: Despite what you may have heard, an Amazon Echo did not call the police earlier this week, when it heard a husband threatening his wife with a gun in New Mexico. On Monday, news reports took Bernalillo County authorities' version of those events credulously, heralding the home assistant as a hero. The alleged act also raised an important question: Do you really want to live in a world where Alexa listens to your conversations, and calls the cops if she thinks things are getting out of hand?

The good news is that you don’t live in that world. Amazon’s Alexa can’t, and did not, call 911. Google Home can’t do it either. No voice-assistant device on the market can. That doesn't invalidate the core question though, especially as Amazon Echo, Google Home, and their offshoots increasingly gain abilities and become more integral to everyday life. How intrusive do you want to let these devices be? Should they be able to call the police? Maybe not even just when specifically prompted, but because they may have heard, for instance, a gun shot?

The Bernalillo County incident almost certainly had nothing to do with Alexa. But it presents an opportunity to think about issues and abilities that will become real sooner than you might think.

The Bernalillo County Sheriff’s Department reported, specifically, that when a man drew a gun on his wife in a home where an Amazon Echo was placed, he said to her, “Did you call the sheriffs?” and the Echo misinterpreted that as a command to call the sheriffs, who then showed up at the front door. The authorities later clarified that someone in the house could be heard in the 911 recording yelling, “Alexa, call 911.”

This could not have happened as described. Amazon’s Echo requires a "wake word" to activate; the default is “Alexa,” but you can also customize it to “Echo,” “Amazon,” or “Computer.” And while they can make calls, an Alexa-powered device can only call another Alexa-powered device. Not only that, but it can only call other Alexa devices that have enabled calling, and have been added to your contact list. Most importantly, these exchanges don't take place over the public switched telephone network, the worldwide network that allows wireless or land phones to actually make calls.

In other words, the sheriffs would have needed an Alexa device of their own for that to ever work, one that the couple in the domestic dispute had in their contact list. Later, the police said that the Alexa was used in combination with some kind of home phone or cellular phone system. That at first sounds more plausible, but is actually also technologically impossible, as the Echo

does not support calls over Bluetooth.

Someone called the police that day. It just wasn't Alexa.

Is Google minting certs for random domains?

- Whoops! It wasn't my intention to suggest that!

iOS v10.3.3 is the one that fixed the BroadPwn bug.

- Not v10.3.2 as we thought. -- v10.3.2 fixed a DIFFERENT bug.

Miscellany

- **The first trailer for season 2 of Netflix's Stranger Things is available.**

- **"Don't F**k with Paste" extension for FireFox & Chrome**

SpinRite

From: Steven Almas

My neighbor, who is a single mother, asked for my IT help. She had a external hard drive with 60,000 photos of her family and children, which was her only copy. The HD did not mount on any computer and she was very upset. I suggested that we purchase SpinRite and try that out. Since I work in IT, I took a dedicated machine and ran SpinRite on her external hard drive, and SpinRite was able to recover ALL but 13 of the 60,000 photos! She now has a comprehensive backup solution in place, and we are forever grateful to Steve Gibson and SpinRite!

Thank You from another Steve!

Closing The Loop

Ned Griffin (@Ned_Griffin) ... and several others:

- @SGgrc Steve, Did I really hear you say my Win 10 machine?
  Never thought I would hear you are using a flying turd OS ??

Several of our listeners sent me images of worrisome account login restrictions:

- Someone noted that the signup for a gov.uk website allowed a maximum of 12 characters and no symbols.
- Someone else sent a login page screen shot showing 8 to 15 characters.

I wanted to remind everyone that while these sorts of limitations pose a definite concern, because they give us cause for concern about the underlying security awareness of the site's design and technology, in and of themselves they are insufficient evidence either way.

For example, a site COULD allow a super-long password with any characters, but which it stores directly in the plaintext that the user submits. So if their database leaked it would be game over for all of the site's users, despite the unlimited password complexity offered up front.

And on the flip side, a site could restrict their input password to just 8 characters, but then pair that with a per-user large random nonce, and run that password through a monster memory-hard acceleration-resistant five-second hashing process to make every brute force guess -- whether online or offline -- impossibly slow and costly.

And, as we know, since hashing inherently turns variable-length content into a fixed-length hash, the whole idea of a site setting ANY fixed upper bound on password length is frightening because it suggests that that may be the size of their account database's plaintext password storage field. Or put another way... If a password is being processed by any PBKDF (password based key derivation function) then it's input length doesn't matter at all.

CRYPTO TENSION

The TLS v1.3 Explicit Wiretap Controversy

"Data Center use of Static Diffie-Hellman in TLS 1.3"
Internet draft document authored by Matthew Green.
https://tools.ietf.org/html/draft-green-tls-static-dh-in-tls13-00

Abstract:
Unlike earlier versions of TLS, current drafts of TLS 1.3 have instead adopted ephemeral-mode Diffie-Hellman and elliptic-curve Diffie-Hellman as the primary cryptographic key exchange mechanism used in TLS. This document describes an optional configuration for TLS servers that allows for the use of a static Diffie-Hellman secret for all TLS connections made to the server. Passive monitoring of TLS connections can be enabled by installing a corresponding copy of this ky in each monitoring device.

While ephemeral (EC) Diffie-Hellman is in nearly all ways an improvement over the TLS RSA handshake, it has a limitation in certain enterprise settings. Specifically, the use of ephemeral (PFS) ciphersuites is not compatible with enterprise network monitoring tools such as Intrusion Detection Systems (IDS) that must passively monitor intranet TLS connections made to endpoints under the enterprise's control. This includes TLS connections made from enterprise load balancers at the edge of the enterprise network to internal enterprise TLS servers. It does not include TLS connections traveling over the external Internet.

Such monitoring is ubiquitous and indispensable in some industries, and loss of this capability may slow adoption of TLS 1.3.

This document describes an optional configuration for TLS servers that allows for the use of a static Diffie-Hellman secret for all TLS connections made to the server. Passive monitoring of TLS connections can be enabled by installing a corresponding copy of this key in each monitoring device.

An advantage of this proposal is that it can be implemented using software modifications to the TLS server only, without the need to make changes to TLS client implementations.

4. Security considerations

> We now consider the security implications of the change described above:
>
> The shift from fully-ephemeral (EC) Diffie-Hellman to partially static Diffie-Hellman affects the security properties offered by the TLS 1.3 handshake by eliminating the Perfect Forward Secrecy (PFS) property provided by the server. If a server is compromised and the private key is stolen, then an attacker who observes any TLS handshake (even one that occurred prior to the compromise) will be able to recover traffic encryption keys and will be able to decrypt traffic.

<snip>
Thus the modification described in Section 4 represents a deliberate weakening of some security properties. Implementers who choose to include this capability should carefully consider the risks to their infrastructure of using a handshake without PFS. Static secret keys should be rotated regularly.

Jul 22, 2017 • Stephen Checkoway
https://www.cs.uic.edu/~s/musings/tls13-enterprises/
Assistant Professor in the Department of Computer Science, University of Illinois at Chicago

<PARAPHRASING FOR BREVITY>
As the TLS 1.3 standardization process (hopefully) comes to a close, there has been some drama on the TLS WG mailing list and at the recent IETF 99 meeting in Prague regarding the use of TLS 1.3 in enterprise networks. This is a surprisingly contentious and important topic that I suspect many people who don't follow protocol development closely may have missed.

Transport Layer Security (TLS) is, without exaggeration, the most important security protocol in use on the Internet today. It is the successor protocol to the older SSL protocol and is used to cryptographically protect a wide variety of Internet communication including online banking, (a significant fraction of) email traffic, more than half of all web browsing, and an ever-increasing amount of normal Internet activity.

TLS is standardized by the Internet Engineering Task Force (IETF) which is organized into a set of working groups (WGs). Each working group has a charter which describes its mission. The TLS WG is currently charged with designing the fourth iteration of the TLS protocol, TLS 1.3. This multi-year process takes place primarily on the TLS mailing list as well as in regular, in-person meetings. The 99th IETF meeting just concluded.

Sounds pretty dry, what's the drama about?

Much of the work is pretty dry and technical. One of the WG's goals for TLS 1.3 is to produce a more secure protocol than prior versions which have had a series of subtle problems. To that end, the WG has removed a number of cryptographic options that reduced the security. This includes removing options like ciphersuites (sets of cryptographic algorithms that work together to secure the traffic) that do not provide forward secrecy.

To quote Wikipedia, "A public-key system has the property of forward secrecy if it generates one random secret key per session to complete a key agreement, without using a deterministic algorithm. This means that the compromise of one message cannot compromise others as well, and there is no one secret value whose acquisition would compromise multiple messages." Forward secrecy also generally requires the session key to be destroyed once the session ends to prevent an adversary from decrypting traffic afterward.

Forward secrecy is a very desirable property in a cryptosystem. As I recall, when removing the non-forward-secret ciphersuites was proposed on the mailing list, there was broad consensus.

At some point, late into the TLS 1.3 design process, some enterprise network operators began to realize that this would reduce their ability to inspect traffic in order to troubleshoot problems within their networks and started asking the TLS WG to restore some of the removed ciphersuites or provide some other mechanism to support their internal network requirements. (The most recently proposed mechanism uses what's called static Diffie–Hellman and works by reusing encryption keys. Interestingly, a form of this is used today as a minor optimization and isn't technically forbidden by TLS 1.3.)

Initially, the WG refused to consider any proposal which would hurt or remove forward secrecy. Recently, as the TLS 1.3 standardization effort has begun to draw to a close, the enterprise network operators have become more vocal. On the mailing list and at the in-person meetings, three viewpoints have emerged. The debate between those with conflicting points of view has been vigorous and, in terms of the sheer number of words written, quite lengthy.

What, exactly, do the enterprise folks want?

In a nutshell, these network operators want the ability to decrypt the traffic that is inside their own networks. Let's call this the enterprise viewpoint. Now keep in mind, any network traffic that is inside their network was either (a) generated from inside their network in which case the enterprise's own computers created the plaintext in the first place; or (b) the traffic was sent from the Internet to one of the enterprise's computers. In either case, they already have the ability to do whatever they want with the plaintext, including storing all of it and examining it at will.

If they already have access to the plaintext, why do they need changes to TLS 1.3 to enable them to get plaintext?

This question is key to the whole debate. The enterprise viewpoint holds that operators need to be able to decrypt traffic from packet captures from various vantage points within the network. For example, they would like to decrypt traffic before and after a load balancer, web server, or

database server in order to pinpoint which part of the network infrastructure is causing problems. On the mailing list and in person, they have been adamant that decryption from package capture (rather than, say, endpoint logging) is the only way they can perform this sort of network debugging at the scale they need given the fragility of what appear to be mind-bogglingly complex network architectures.

It seems pretty reasonable to support this usecase. What's the problem with accommodating their request? After all, this will only be for use inside their own networks.

On the one hand, this is reasonable and is completely supported today using TLS 1.2. (Indeed, one of the suggestions has been for network operators to continue using TLS 1.2 inside their networks if they need this capability.) On the other hand, there's no technical way to confine proposals to enable decryption to a particular network or data center.

[[Except that, technically, there kinda is: Users who wanted to prevent TLS 1.3 static key decryption eavesdropping could disable TLS 1.3 support in their browsers, which would force a downgrade to TLS 1.2... which is still completely secure.]]

There are two major concerns raised by those opposed to breaking or degrading forward secrecy. Let's call this the forward-secret viewpoint. One concern raised by those with the forward-secret viewpoint is that proposals such as the static Diffie–Hellman approach mentioned above will enable wiretapping which would violate the IETF's Policy on Wiretapping. Although that may be true (and this is hotly contested), some other technical mechanisms have been proposed that would make such wiretapping externally visible.

The second concern is both more subtle and, I think, more compelling. TLS (and SSL before it) has a history of supporting weak cryptography and this support has come back to bite us several times. The best of example of this is the export ciphersuites. These used cryptographically weak algorithms but were, at one point in time, the only ciphersuites that could be legally exported from the US. Two decades after the use of export ciphersuites should have ended, researchers showed how to abuse support for these deprecated algorithms in modern TLS libraries to man-in-the-middle TLS connections.

The forward-secret viewpoint holds that the TLS WG should not standardize any weaker form of TLS and if this makes some network operators' jobs harder, then so be it.

That's two viewpoints—enterprise and forward-secret—what's the third?

Let's call the third viewpoint, the pragmatic viewpoint. This viewpoint holds that whether or not enterprise network operators really need the decryption capability, some of them really want it. And since they really want it, they're going to do something to get it. It's strictly better for the mechanism to be designed in public, following normal IETF procedures, than to be cobbled together by people whose focus is on operations and not, necessarily, on security. It's worth noting that at least one of the authors of the static Diffie–Hellman proposal mentioned above firmly holds the pragmatic viewpoint.

Which viewpoint is correct?

Before I say which viewpoint I think makes the strongest case, I want to point out that I'm sympathetic to all three viewpoints. The network operator's job is not an easy one (or so I assume; it's definitely outside my particular area of expertise). If they say they need plaintext in order to do their job, I don't think I'm in a position to contradict them.

The pragmatic viewpoint is quite compelling. All else being equal, I'd much rather have the IETF design a standard mechanism to support the network operators' needs than have a hodgepodge of home-grown, difficult to use, noninteroperable, and potentially insecure solutions.

[[And if I forgot to mention this when talking about Matt Green's role in this, I believe this is his position. Given his historical positioning on issues like these, he must simply be saying: "If this is what we're going to have, let's at least analyze it fully and carefully and make sure we're NOT opening Pandora's Box with unintended consequences.]]

But, as they say, all else is rarely equal. The Internet is for End Users, not for network operators. The protocols we design today will, for better or for worse, be in use for decades. End-users have been paying the price for our mistakes and past compromises on security. As protocol and implementation deficiencies necessitate new network hardware and software, the network operators have paid their own price.

To rebut the enterprise and pragmatic viewpoints, I need not take a security-maximalist view. The sense of urgency from the operators and the pragmatists is, I believe, unwarranted. Yes, switching to TLS 1.3 will prevent operators from doing precisely what they're doing today; however, there is currently no need to switch. TLS 1.2 supports their usecase and TLS 1.2, when used correctly, is secure as far as we know. Of course the network operators won't receive the benefits of mandatory forward secrecy, but that is precisely what they are asking to give up in TLS 1.3.

Designing secure protocols is hard. To date, our best efforts have not been as successful as we would like. In my view, the only option we have is to design the most secure protocols we can to achieve our stated objectives. We may still get it wrong, of course. My hope is that in 20 years, we won't, once again, be dealing with security issues we know about today. Instead, I hope we'll be dealing with a whole new set of security issues.