Security Now! #990 - 09-03-24

Is Telegram an Encrypted App?

This week on Security Now!

Telegram's founder, owner and CEO arrested in France. What does that mean? One year after Microsoft began offering free cloud security event logging. How's that going? To no one's surprise, CrowdStrike is losing customers – But how many? Microsoft to meet with CrowdStrike and other vendors to discuss new solutions. Yelp is not happy with Google. Did/does Google put their thumb on the scale? Where do you go to purchase yourself some DDoS? How about sending a Telegram? Chrome exploits are becoming more rare and difficult to find so Google has upped the ante. Believe it or not, Cox Media Group is still promoting their incredibly privacy invading "Active Listening" capability. How about secretly having foreigners doing all of your work for you. What could possibly go wrong? And Johns Hopkins Cryptographer Matthew Green has become increasingly annoyed by Telegram's claims of being an encrypted messaging platform. So he finally asks the question: Is Telegram an Encrypted App?

When the Universe is suggesting that you should take the stairs... listen.

Security News

Telegram puts End-to-End Privacy in the Crosshairs

I gave this week's lead story the title "Telegram puts End-to-End Privacy in the Crosshairs" because I think that's probably what's ultimately being tested here. At the time of last week's podcast, the news was that Pavel Durov, founder and CEO of the Telegram instant messaging system, had been detained in France after we flew into and landed in French territory on a private jet. Next, we learned that his status had changed from "detained" to formally arrested. Followed by last Wednesday's release on 5 million Euros bail and being banned from leaving France since he is now facing charges over his responsibility for the many illegal and in some cases abhorrent things Telegram's users have been found doing in light of there being no content mediation within Telegram of any kind. Pavel is being held responsible for that.

The reason this is intensely interesting is that it brings us back to the big and still unanswered question of how the world is ultimately going to deal with end-to-end encrypted messaging and whether governments are going to allow their citizens to hold truly private electronic conversations without **any** form of content moderating oversight?

In the present case of Telegram, the charges which French authorities have levied against Pavel include being complicit in running an online platform that allows sharing of CSAM (which is, as we know, the abbreviation for Child Sexual Abuse Material), drug trafficking, fraud and money laundering, as well as not cooperating with authorities when required to do so by law.

The French news outlet Le Monde reported that France's police office that tackles violent crimes against children, issued the warrant for Durov's arrest. In a LinkedIn post that was later deleted, that office's Secretary General said that "at the heart of this case is the lack of moderation and cooperation of the platform (which has nearly 1 billion users), particularly in the fight against pedo-criminality."

The EU arm of Politico reported that the specific incident that was cited in the arrest warrant was Telegram's refusal to identify a specific user after being served a judicial request. Politico wrote, after viewing a document relating to the warrant: *"The warrants [for Pavel Durov and his brother Nikolai] were issued after an undercover investigation into Telegram led by the cybercrime branch of the Paris prosecutor's office, during which a suspect discussed luring underaged girls into sending "self-produced child pornography," and then threatening to release it on social media."*

According to the document, the suspect also told the undercover investigators that he had raped a young child. Telegram did not respond to the French authorities' request to identify the suspect. As we've often observed, Telegram is the most combative of all the major social media platforms in their attitude and approach to content moderation and lawful assistance requests. And it paints this as a clear benefit in its own FAQ, which explains that its distributed architecture is used to confound court orders. Telegram unabashedly boasts:

> *"Telegram uses a distributed infrastructure. Cloud chat data is stored in multiple data centers around the globe that are controlled by different legal entities spread across different jurisdictions. The relevant decryption keys are split into parts and are never kept in the same*

place as the data they protect. As a result, several court orders from different jurisdictions are required to force us to give up any data. Thanks to this structure, we can ensure that no single government or block of like-minded countries can intrude on people's privacy and freedom of expression. Telegram can be forced to give up data only if an issue is grave and universal enough to pass the scrutiny of several different legal systems around the world. To this day, we have disclosed 0 bytes of user data to third parties, including governments."

Telegram's terms of service do state that illegal pornographic content is not allowed on its publicly viewable areas. Its FAQ says it will only take action on illegal content in these areas, which comprise sticker sets, channels and bots. However, Telegram assures its users that:

All Telegram chats and group chats are private amongst their participants. We do not process any requests related to them.

In other words, within any private groups, which may include up to 200,000 people, anything goes without any supervision and with an explicit guarantee of technically imposed privacy. So it should be no surprise that many investigations have found child abuse material for sale on Telegram.

However, this is an example of an instance where the devil is in the details and where encryption may not mean what its user's imagine. Today's podcast topic addresses the interesting question of whether or not and to what degree Telegram is actually an encrypted app. Our long-time listeners may recall that I have never been impressed with Telegram's encryption because it's a perfect example of what we all know should not be done: Telegram uses a homegrown cipher that a couple of guys just made up. And they did this well after the world had learned how to do encryption correctly. The fact that they have offered a large cash prize to anyone who could break it doesn't change the fact that it's not based upon any sound, formally designed and tested cryptographic system. So we're going to look far more closely at Telegram at the end of today's podcast, since Johns Hopkins Cryptographer, Matthew Green just posted an intriguing piece titled: *"Is Telegram really an encrypted messaging app?"*

But be that as it may, Telegram offers one important feature that makes it unique among all of the private messaging systems: Whereas Telegram can comfortably provide privacy for 200,000 members of a large group, Apple's iMessage groups are limited to 32 participants, Signal groups are limited to 1,000, and WhatsApp's variant of Signal limits group size to 1024.
It turns out that implementing true end-to-end encryption across large groups with many participants is not trivial. But what much of the media misses is that, as we'll see, Telegram doesn't actually do that.

Telegram's unique value proposition is to provide large groups with unmoderated communication and some degree of privacy. Telegram describes itself as a 'cloud-based messenger' that provides 'seamless sync' across devices. But to do that, **it** needs to have access to the content of those messages. And we know that, because Telegram themselves **can** access the content of conversations. So it certainly could invest in moderation if it chose too. It chooses not to. NBC News reported that child safety groups in the US, UK and Canada all get short shrift from Telegram when reporting CSAM.

This is in contrast to an app like Signal, for example, which also espouses privacy-first values. Signal has built its app so that its technology reflects those values as much as possible while still enforcing privacy. So, although Signal collects no content from its users and only minimal metadata about how they use the service, Signal is able to and will respond to law enforcement requests, but only to the extent of providing account creation dates and the date an account last accessed Signal. This means that while Signal is not, in practice, a great deal more helpful than Telegram, at least Signal is not openly combative and can honestly say that it has wholeheartedly cooperated with court orders to the limit of its ability and technology.

So what of Pavel Durov? This may just be a shot across the bow. And it might wind up being good for Telegram's business model to see their founder and CEO being detained and tried for his refusal to comply. Since Telegram currently has only 41 million users in the European Union, this falls short of the 45-million user threshold that would subject it to the EU's Digital Services Act. With Telegram not categorized as a very large online platform, it's not subject to the EU's stricter transparency and content moderation rules.

However, the Financial Times recently reported that the EU is now investigating Telegram for misrepresenting the total number of EU users in order to fall below that 45 million threshold. [I'm shocked, I tell you! Shocked!] Last February's claim of only have 41 million users within the EU will be carefully examined.

The one thing I want to observe before we move on is that both of today's major mobile platforms – iOS and Android – manage their client apps with an iron grip. They do this to enforce both security and control over these client apps. For example, the reason SpinRite boots directly over its user's hardware and brings along its own minimal OS is because it **cannot** obtain the direct hardware access it requires from within any operating system environment.

Nothing like that exists for our mobile operating systems. None of the various messaging platforms are able to obtain anything approaching direct access to the platform's underlying hardware. So we should always be mindful of the fact that the OS runs the camera, runs the screen and runs the virtual keyboard. And that access to those resources is granted to the applications that are running. That's why we're able to seamlessly switch among applications without any application being able to prevent that. Apps are powerless clients of their mobile platform OS. So a messaging app such as Signal, WhatsApp, Telegram or iMessage may be as clever as it wishes with how the content that it communicates is encrypted and decrypted. But **everything** that is eventually communicated to and from its users passes through the control of the OS, and that OS is always able to "see" everything that's happening in the clear without any form of obfuscation or encryption.

There's a game of "hot potato" going on here, with no one wanting to take responsibility for the content that's passing through their control before it's encrypted or after it's decrypted. But the truth is that the vendor of the underlying platform – Apple or the supplier of an Android OS – is in the unique position to monitor what's going on before it's turned over to **any** messaging app, and to similarly inspect what its client apps decrypt before it's presented to its user. We know how adamantly the platform vendors want to stay as far away as possible from taking any responsibility for what their users and their client apps do. And I know that we all want to retain the total privacy that we're currently being assured we're receiving. But Pavel Durov's arrest and

indictment by French authorities shows us that we should probably regard the privacy we're enjoying today... as fleeting... since no government wants to be completely blind to the conduct of its citizenry.

Free security logging is good for everyone.

To set the stage for some news, recall that five months ago, last April, the U.S. Cyber Safety Review Board released a scathing report which squarely placed the blame on Microsoft for the nation state-linked intrusion into Microsoft Exchange Online which led to the theft of about 60,000 U.S. State Department emails the previous summer. The CSRB report stated that the breach <quote> "was preventable and should never have occurred." The report elaborated that a series of operational and strategic decisions by Microsoft pointed to a corporate culture that deprioritized investments in enterprise security and rigorous risk management, despite the central role the company plays in the larger technology ecosystem. The CSRB urged Microsoft to publicly share its plans to make fundamental, security focused reforms across the company and its suite of products. The board also recommended that all cloud services providers and government partners enact security-focused changes.

Among the criticism that was heaped upon Microsoft last year what that it was charging for zero-cost features such as security logging that would have gone a long way to help detect the early states of the various intrusions its users and customers had been experiencing. The tech savvy senator, Ron Wyden said at the time:

"Unfortunately, as Microsoft's $15 billion-plus cybersecurity business grows, Microsoft's incentives are not to deliver secure operating systems and cloud software to its customers, but to deliver insecure products and upsell them on cybersecurity add-ons. It shouldn't have taken multiple disastrous hacks of federal systems for Microsoft to make essential security features standard for government customers, but better late than never."

We're talking about this today because now, one year later, evidence is emerging of the effect of something as simple as free security logging. Last Tuesday, the publication CybersecurityDive posted a report titled: *"CISA officials credit Microsoft security log expansion for improved threat visibility"* They wrote:

> *Greater access to Microsoft event logs is paying off for U.S. government agencies and critical infrastructure providers, which have gained greater visibility into their network environments, the Cybersecurity and Infrastructure Security Agency said Saturday.*
>
> *Microsoft expanded free access to security logs in 2023 after a state-linked threat actor stole thousands of emails from the State Department after gaining access to Microsoft Exchange Online. Jeff Greene, CISA's executive assistant director for cybersecurity, confirmed via email: "Yes, Microsoft has expanded access to the logging elements that were used by the State Department to detect the 2023 compromise to a vastly larger set of customers, including all federal agencies and numerous critical infrastructure organizations. These new logs are being used by organizations today to detect threats." Greene added: "CISA will continue to work with Microsoft and other companies to ensure that their products are secure by design and that Microsoft lives up to the commitments it has publicly announced around improving the security of its products following the 2023 compromise."*

The win for the U.S. government comes as CISA, along with the FBI, National Security Agency and a group of foreign cybersecurity authorities led by Australia, released a best practices guide for event logging last week.

The new guide is part of an effort to combat sophisticated threat activity from state-linked threat groups, such as Volt Typhoon. The group uses living-off-the-land techniques to disguise its threat activities using normal security tools that won't trigger alerts when moving around computer networks. Security researchers at Reliaquest have been tracking a ransomware actor known as Medusa, which has also used living-off-the-land techniques in multiple attacks.

Alex Capraro, cyber intelligence analyst at Reliaquest, said via email: "By implementing the best practices for event logging and threat detection outlined in this guide, organizations can enhance their ability to identify and mitigate malicious activities, thereby protecting their networks, devices, and data from compromise."

So that's all good news. Microsoft made free security event logging available a year ago and it's turning out to be extremely beneficial.

CrowdStrike hemorrhaging customers

So far, CrowdStrike reports that it expects to lose around $60 million in net new annual recurring revenue and subscription revenue in the aftermath of its technical outage. I don't have a good sense for what that represents as a percentage of total revenue, but it doesn't sound like much. Nevertheless, CrowsStrike is endeavoring to retain customers by offering various discounts. Their CEO George Kurtz denied rumors that the company was losing customers to rivals but of course that will happen to some degree, even though I'm sure I'd be staying unless I was otherwise unhappy. The changes they've made seem solid. Both SentinelOne and Palo Alto Networks have claimed that they've been fielding calls from CrowdStrike customers over the past weeks. And, again, I wouldn't doubt that for a moment. But it doesn't seem like many are leaving.

Microsoft to meet privately with EDR (Endpoint Detection & Response) vendors

Microsoft's announcement reads:

On Sept. 10, 2024, [which is a week from today] *Microsoft will host a Windows Endpoint Security Ecosystem Summit at our Redmond, Washington, headquarters. Microsoft, CrowdStrike and key partners who deliver endpoint security technologies will come together for discussions about improving resiliency and protecting mutual customers' critical infrastructure. Our objective is to discuss concrete steps we will all take to improve security and resiliency for our joint customers.*

The CrowdStrike outage in July 2024 presents important lessons for us to apply as an ecosystem. Our discussions will focus on improving security and safe deployment practices, designing systems for resiliency and working together as a thriving community of partners to best serve customers now, and in the future.

In addition to ecosystem partners, Microsoft will invite government representatives to ensure the highest level of transparency to the community's collaboration to deliver more secure and reliable technology for all. It is expected that the Windows Endpoint Security Ecosystem Summit will lead to next steps in both short- and long-term actions and initiatives to pursue, with improved security and resilience as our collective goal. We will share further updates on these conversations following the event.

I would imagine that the government representatives are invited as a means of showing that something is being done to keep anything like this from happening again. In other reporting about this I saw that Microsoft plans to discuss new ways of building EDR products so that they can still get their job done while relying more on user mode and less on proprietary kernel drivers.

Yelp's Unhappy with Google.

Yelp has filed an antitrust lawsuit against Google, alleging the company has a monopoly over the search market, which it has abused to promote its own review business. It's certainly true that controlling search is an incredibly powerful place to be. We view the Internet through what our chosen search engine reveals. My wonderful luddite realtor friend thought that Google WAS the Internet. Since "search" is our portal, any bias in search results will have a profound impact. There's a reason why SEO – Search Engine Optimization – has been a booming business.

Telegram as the hotbed for DDoSass – DDoS as a Service

Everyone seems to be piling on Telegram this week. It's not as if they probably don't more attention. And in today's Internet threat landscape, that's what's going to happen on any large unmoderated social network platform which everyone assumes is somehow secure. In this instance the security firm FalconFeeds has taken a deep look into the flourishing business of DDoS for hire, or more specifically DDoSaaS as it's formally called: DDoS as a Service.

In a posting last Thursday titled *"DDoS-as-a-Service: The Dominating Phenomenon on Telegram"*, Jacob Abraham wrote:

In today's digital landscape, Distributed Denial of Service (DDoS) attacks have become one of the most powerful tools in a cybercriminal's arsenal. These attacks, often facilitated by DDoS-as-a-Service (DDoSaaS) platforms, DDoS-for-Hire services, and botnet-for-hire networks, can disrupt online services, extort businesses, and even advance political agendas. At FalconFeeds.io, our latest research reveals a staggering 3,529 DDoS incidents occurred in Europe during the first half of 2024, making up 60% of the total cyberattacks we analyzed.

The rise of DDoS-as-a-Service (DDoSaaS) on platforms like Telegram is a significant contributor to this alarming trend. Telegram has emerged as a hotbed for cybercriminals looking to offer DDoS-as-a-Service (DDoSaaS). On various Telegram channels and groups, vendors openly advertise a range of DDoS attack services at different price points, making it alarmingly easy for even those with minimal technical expertise to hire a DDoS attack. Telegram's encryption and anonymity features create an ideal environment for these illegal activities to flourish unchecked.

Our research has identified over 140 Telegram channels and groups actively offering these services, with 80% of them being currently active and trading these services primarily through cryptocurrencies. This trend underscores the growing accessibility and anonymity of DDoS attacks, posing a significant threat to businesses and individuals alike.

So called "Basic Attacks" are available for as little as $10 per month. And the power and cost scales upward with more "Sophisticated Attacks" being prolonged and high-intensity costing as much as thousands of dollars. Price lists are often displayed on Telegram channels, with discounts available for repeat customers or bulk orders. This availability and accessibility has turned DDoS into a commodity, available to anyone willing to pay.

And again, where are these services to be found? On Telegram.

Chrome grows more difficult to exploit

Last Wednesday, Google announced that it would be increasing, in some cases by as much as a factor of 5, the reward bounties it would be offering for the responsible discovery and reporting of Chrome exploits due to the increased difficulty of exploiting Chrome. So that's all good news for the world's #1 web browser. Google said:

Time flies: Believe it or not, Chrome Browser turns 16 this year and, following closely behind, the Chrome VRP (Chrome's Vulnerability Rewards Program) is turning 14! As Chrome has matured over these years, finding the most impactful and exploitable bugs has become more challenging. At the same time, new features are frequently introduced into Chrome that may result in new issues which we also want to encourage being reported.

Therefore, it is time to evolve the Chrome VRP rewards and amounts to provide an improved structure and clearer expectations for security researchers reporting bugs to us and to incentivize high-quality reporting and deeper research of Chrome vulnerabilities, exploring them to their full impact and exploitability potential.

In this blog post, we'll explain how we have moved away from a single table of reward amounts for non-mitigated bugs, and separated out memory corruption issues from other classes of vulnerabilities. This will allow us to better incentivize more impactful research in each area, and also reward for higher quality and more impactful reporting.

We have remodeled our reward structure for memory corruption vulnerabilities into the following categories:

- *High-quality report with demonstration of RCE: Report clearly demonstrates remote code execution, such as through a functional exploit.*
- *High-quality report demonstrating controlled write: Report clearly demonstrates attacker controlled write of arbitrary locations in memory.*
- *High-quality report of memory corruption: Report of demonstrated memory corruption in Chrome that consists of all the characteristics of a high-quality report.*
- *Baseline: A report consisting of a stack trace and PoC displaying evidence that memory corruption is triggerable and reachable in Chrome.*

*While the reward amounts for baseline reports of memory corruption will remain consistent, we have increased reward amounts in the other categories with the goal of incentivizing deeper research into the full consequences of a given issue. The highest potential reward amount for a single issue is now **$250,000** for demonstrated RCE in a non-sandboxed process. If the RCE in a non-sandboxed process can be achieved without a renderer compromise, it is eligible for an even higher reward, to include the renderer RCE reward.*

https://bughunters.google.com/blog/5302044291629056/chrome-vrp-reward-updates-to-incentivize-deeper-research

I'm not going to go into any finer detail here, so I've dropped a link to Google's BugHunter's posting into the show notes for anyone who may want more. It's a good move and good news that since Chrome is becoming more difficult to exploit the payouts are increasing commensurately. This may also be the first time I have ever – anywhere – seen a software publisher actually say: *"At the same time, new features are frequently introduced into Chrome that may result in new issues which we also want to encourage being reported."* Anyone who has been following this podcast for more than a few months will think "yeah, of course", but who's actually ever heard any publisher say that? So, props to Google.

Cox Media Group's "Active Listening" has apparently not ended

When I encountered this next bit of news I thought I was experiencing Deja Vu. The summary was titled *"CMG's Active Listening"* and read: *"After media companies and device vendors spent a decade telling customers that microphones baked into their devices are not secretly recording audio, a leaked pitch deck from the Cox Media Group (CMG) is advertising a new service that can show ads to users based on what they've said near microphones. Google kicked CMG from its advertising program after 404 Media acquired the slide deck and then asked Google to comment."* Since this was ringing some bells, I went to GRC's Security Now! page and entered "Cox Media Group" into the search. The first link and summary that appeared was from our podcast #953, the last podcast of the year, dated December 21st, and that podcast was titled "Active Listening". After the news what CMG was reportedly doing and bragging about on their webpage which had the URL "active-listening-an-overview" they took the page down, but not before the Internet Archive's spiders found and archived the page. And that was GRC's shortcut of the week which is still pointing to the page in question. So it's http://grc.sc/953 and it's still every bit as unnerving as it was nine months ago.

The page starts out with: *"Imagine a world where you can read minds. One where you know the second someone in your area is concerned about mold in their closet, where you have access to a list of leads who are unhappy with their current contractor, or know who is struggling to pick the perfect fine dining restaurant to propose to their discerning future fiancé. This is a world where no pre-purchase murmurs go unanalyzed, and the whispers of consumers become a tool for you to target, retarget, and conquer your local market. It's not a far-off fantasy-it's Active Listening technology, and it enables you to unlock unmatched advertising efficiency today so you can boast a bigger bottom line tomorrow. Do we need a bigger vehicle? I feel like my lawyer is screwing me. It's time for us to get serious about buying a house—No matter what they're saying, now you can know and act."*

And lower down under "how we do it" they say:

> *Whether you're a scrappy startup or a Fortune 500, Active Listening makes the unreachable in-reach. CMG can customize your campaign to listen for any keywords/targets relevant to your business. Here is how we do it: We flesh out comprehensive buyer personas by uploading past client data into the platform. We identify top-performing keywords relative to the type of customer you are looking for. We set up tracking via pixels placed on your site so we can track your ROI in real-time. AI lets us know when and what to tune into. Our technology detects relevant conversations via smartphones, smart TVs, and other devices. As qualified consumers are detected, a 360 analysis via AI on past behaviors of each potential customer occurs. With the audience information gathered, an encrypted evergreen audience list is created. We use the list to target your advertising via many different platforms and tactics, including: Streaming TV/OTT, Streaming Audio, Display Ads, Paid Social Media, YouTube, Google/Bing Search (pay per click). Our technology provides a process that makes it possible to know exactly when someone is in the market for your services in real time, giving you a significant advantage over your competitors. Territories are available in 10 or 20-mile radiuses, but customizations can be made for regional, state, and national coverage.*

And then in their own FAQ, incredibly, they actually ask and answer:

> ***A:** Is Active Listening Legal?*
>
> ***Q:** We know what you're thinking. Is this even legal? The short answer is: yes. It is legal for phones and devices to listen to you. **When a new app download or update prompts consumers with a multi-page terms of use agreement somewhere in the fine print, Active Listening is often included.***
>
> ***A:** How Does Active Listening Technology work?*
>
> ***Q:** Our technology is on the cutting edge of voice data processing. We can identify buyers based on casual conversations in real time. It may seem like black magic, but it's not-it's AI. The growing ability to access microphone data on devices like smartphones and tablets enables our technology partner to aggregate and analyze voice data during pre-purchase conversations.*

What just happened to bring this back on our radar is that 404 Media, the same group that had previously reported on CMG's webpage which was quickly taken down, obtained the marketing pitch deck that is still, nine months later, being sent by CMG to prospective companies. 404 Media forward the deck to Google who then reportedly kicked CMG off its Partner Program in response.

That was the right thing for Google to do, but how is it that a massive media group such as CMG is able to, with a straight face, say that consumers are permitting this, making it legal, because "somewhere in the fine print" this permission is being given. Wow.

Cascading Bloom Filter follow-up

Last week's serious propeller cap pure computer science nerdfest episode was every bit as much of a hit as I hoped it might be. It's fun thinking about new things, especially for this audience. But I wanted to take a moment to acknowledge the feedback I received from a number of our more technical listeners who correctly observed that the three layers of bloom filtering I described last week could not always be guaranteed to be sufficient. Those observations were correct. My goal was to clearly establish the concepts involved. To talk about Bloom filter applications where the filter's inherent tendency to produce false positives would and would not represent any actual trouble. And then, in cases where no false positives could be tolerated, to introduce the idea of successive layers of bloom filters, where the next layer of the cascade would capture and be trained on the set of false positives generated by the previous layer. So those who noted that the third layer might also produce some false positives were 100% correct. And a fully realized implementation of this system actually takes the form of a variable depth cascade where successively smaller layers continue to be added and trained until no misbehavior is observed when the entire corpus of unexpired certificates is fed through the entire cascade. Eventually, there will be nothing to train a next layer on, since not a single false positive will manage to make it all the way through the cascade. In retrospect, I suppose that I could have explained that last week, but as it was I felt that it was already a lot.

And also, just for the record, I used one megabit as the number of bits in the first Bloom filter level, which would be addressed by 20 bits taken from the candidate certificate's hash, purely for the sake of illustration since that made it much easier to describe and visualize. The actual size of the first filter and of each successive filter, as well as the number of Bloom layer bits that will be set by the addition of each certificate are all well understood and are determined by a bunch of fancy math. But that was also irrelevant to understanding the overall concept of probabilistic Bloom filtering, which was last week's goal.

So, definitely big props to our listeners who said "uh, Steve, you do realize that three layers might not always do the job, right?" And speaking of listeners and their feedback...

Closing the Loop

We were talking a couple of weeks ago about the security company who discovered that they had inadvertently hired an actively hostile employee based in North Korea who had gone to extreme measures to spoof their identity and set up a fake domestic operating location.

As it happens, one of our listeners was recently solicited in a similar fashion. He wrote:

> *Hi Steve,*
>
> *I was interested in the story from SN 985 about North Korean hackers posing as US workers and getting hired by American tech companies.*
>
> *I'm currently between jobs, and I got an email from someone claiming to be from Malaysia who found my profile on a job board. This person is proposing a "collaboration" wherein I get hired for a remote American tech job, then he impersonates me and does all the work. I send*

85% of the paycheck to him, pocketing the other 15% for myself. I don't think anyone's ever approached me to ask for my participation in something so blatantly illegal before. Though if I'm being honest I was momentarily tempted, since it would be easy money for me and he'd still be making more this way than he could working in his own country. Sounds like a win-win, apart from the whole fraud thing and serious criminal and reputational liability to me.

Anyway, I never responded to the message so I can only speculate, but I wonder if this is actually how the situation at KnowBe4 happened. I have no reason to believe the sender of the email used his real name, or that he's based in Malaysia. It might be more plausible that this message is part of the sort of large campaign that uses an "IT mule laptop farm" as described in the story. His Gmail address is generic and formulaic enough that I suspect there are many other identities being controlled by the same party. The message itself is so carefully wordsmithed that it doesn't strike me as a personal note from a fellow dev. I also received a follow-up message a week later, which felt more automated than not.

Regardless, I thought you might be interested to see it since the public reads about the aftermath of these stories but their onset usually happens behind closed doors. Forwarding the full message here in case you'd like to read it on-air.

Thanks, Parker

This is intriguing, indeed. So here's the solicitation email that Parker received:

Subject: Open to a collaboration?

Hi Parker, I hope you're doing well and don't mind me reaching out. I'm Lucas, a full-stack developer from Malaysia. I found your profile on https://app.usebraintrust.com/talent/ and wanted to propose a collaboration. I don't currently have any projects that need your help, but our collaboration could be either technical or non-technical.

For the non-technical aspect, I'd like your help with entrepreneurial factors for my development work. If we end up getting jobs together and working on them, it would be a technical collaboration.

To keep it short, I'm looking to get well-paid jobs with companies or clients in the US. While this is feasible from Malaysia, they tend to prefer hiring developers from similar time zones. Unfortunately, I'm in GMT+8, while the United States is in PT to ET. Especially for full-time jobs at companies, they typically don't hire developers outside of the US.

So, I believe the best way to get US jobs is to "impersonate" someone who resides in the US. It might sound risky, but it won't be risky as long as we keep this 100% confidential. Besides, I don't mean that I want your identity information.

Have you heard of Upwork.com or Toptal? They're the largest freelancing job markets in the world, where most individual clients in the US look for developers for their projects. There's no easy way to get well-paid jobs, and Upwork or Toptal has a lot of competitive freelancers. However, I'm very confident that I can get great jobs to make decent money.

Here's how it would work:

- *You open an Upwork or Toptal account and log in to it on your secondary laptop.*
- *I connect to your secondary laptop via the AnyDesk app, and I search for jobs.*
- *You receive money into your bank account once I finish jobs with clients.*
- *You take your commission and send me the remaining. This would be a non-technical collaboration, and I would suggest a split of 15% ~ 20% for you and 80% ~ 85% for me.*

For full-time jobs at US companies, which obviously makes us way more money than freelancing jobs, I would apply for jobs on LinkedIn and you would crack the interviews. However, I'd say this is the advanced step of the collaboration, which should be based on a strong foundation of trust between us.

Here's how it would work:

- *I apply for company jobs on LinkedIn using your LinkedIn account and get you scheduled with interviews.*
- *You crack the interviews and get job offers.*
- *I perform day-to-day work on those jobs while you attend the scrum meetings. (I can join the meetings if team members usually turn off cameras.)*
- *You get paid into your bank account bi-weekly or monthly, and you send me my portion after deducting your commission. This would be a mixture of technical and non-technical collaboration, and I would suggest a split of 20% ~ 25% for you and 75% ~ 80% for me.*

Please reply to this email if you want to schedule a call to discuss this further or if you have any other opinion for the collaboration.

Best, Lucas

My own "credibility filter" snaps on when I read a sentence like "*For full-time jobs at US companies, which obviously makes us way more money than freelancing jobs, I would apply for jobs on LinkedIn and you would crack the interviews.*" — That sentence, and the rest of the note for that matter, does not strike me as having been written by a non-native English speaker. Maybe it was AI generated. But in any case, Parker's sense of right and wrong kept him from responding. Since this problem of North Korean infiltrators worming their way into Western jobs is clearly very real, with a solicitation this slick and polished, it occurred to me that this might have been some sort of sting operation designed to catch westerners who would be willing to expose their employers to potential hostile exploitation.

Is Telegram an Encrypted App

This blog is reserved for more serious things, and ordinarily I wouldn't spend time on questions like the above. But much as I'd like to spend my time writing about exciting topics, sometimes the world requires a bit of what Brad Delong calls "Intellectual Garbage Pickup," namely: correcting wrong, or mostly-wrong ideas that spread unchecked across the Internet.

This post is inspired by the recent and concerning news that Telegram's CEO Pavel Durov has been arrested by French authorities for its failure to sufficiently moderate content. While I don't know the details, the use of criminal charges to coerce social media companies is a pretty worrying escalation, and I hope there's more to the story.

But this arrest is not what I want to talk about today. What I do want to talk about is one specific detail of the reporting. Specifically: the fact that nearly every news report about the arrest refers to Telegram as an "encrypted messaging app."

This phrasing drives me nuts because in a very limited technical sense it's *not wrong.* Yet in every sense that matters, it fundamentally misrepresents what Telegram is and how it works in practice. And this misrepresentation is bad for both journalists and particularly for Telegram's users, many of whom could be badly hurt as a result.

So... Does Telegram have encryption or doesn't it?

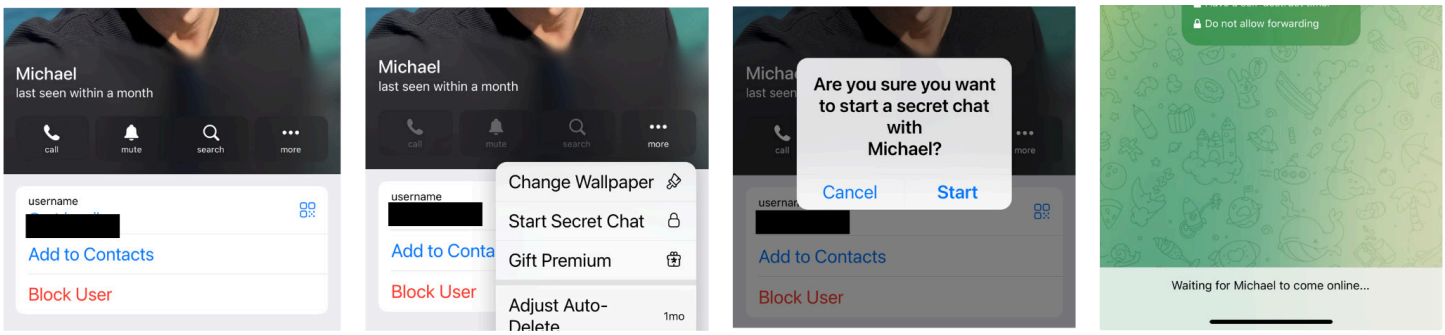
Many systems use encryption in some way or another. However, when we talk about encryption in the context of modern private messaging services, the word typically has a very specific meaning: it refers to the use of default end-to-end encryption to protect users' message content. When used in an industry-standard way, this feature ensures that every message will be encrypted using encryption keys that are only known to the communicating parties, and not to the service provider.

From your perspective as a user, an "encrypted messenger" ensures that each time you start a conversation, your messages will only be readable by the folks you intend to speak with. If the operator of a messaging service tries to view the content of your messages, all they'll see is useless encrypted junk. That same guarantee holds for anyone who might hack into the provider's servers, and also, for better or for worse, to law enforcement agencies that serve providers with a subpoena.

Telegram clearly fails to meet this stronger definition for a simple reason: it does not end-to-end encrypt conversations by default.

If you want to use end-to-end encryption in Telegram, you must manually activate an optional end-to-end encryption feature called "Secret Chats" for *every single private conversation you want to have*. The feature is explicitly not turned on for the vast majority of conversations, and is only available for one-on-one conversations, and ***never*** for group chats with more than two people in them. As a kind of a weird bonus, activating end-to-end encryption in Telegram is oddly difficult for non-expert users to actually do.

For one thing, the button that activates Telegram's encryption feature is not visible from the main conversation pane, or from the home screen. To find it in the iOS app, I had to click at least four times — once to access the user's profile, once to make a hidden menu pop up showing me the options, and a final time to "confirm" that I wanted to use encryption. And even after this I was not able to actually have an encrypted conversation, since Secret Chats only works if your conversation partner happens to be online when you do this.



Starting a "secret chat" with my friend Michael on the latest Telegram iOS app. From an ordinary chat screen this option isn't directly visible. Getting it activated requires four clicks: ***(1)*** *to get to Michael's profile (left image),* ***(2)*** *on the "…" button to display a hidden set of options (center image),* ***(3)*** *on "Start Secret Chat", and* ***(4)*** *on the "Are you sure…" confirmation dialog. After that I'm still unable to send Michael any messages, because Telegram's Secret Chats can only be turned on if the other user is also online.*

Overall this is quite different from the experience of starting a new encrypted chat in an industry- standard modern messaging application, which simply requires you to open a new chat window.

I need to interrupt for a moment to clarify and explain something that's probably not clear: There is a world of difference between a messaging app providing true end-to-end encryption, and merely having encrypted communications. Matthew doesn't bother to draw attention to this distinction because he lives in the world of encryption where the phrase "end-to-end encryption" has a very specific meaning. But it's easy to miss this important distinction.

The reason iMessage imposes a 32-member limit on group messaging, and Signal and WhatsApp both impose 1K limits, is that these services, which Matthew describes as "industry-standard modern messaging applications" are all actually encrypting every party's message, end-to-end, individually to every other party. Telegram is incapable of doing this – ever. It has no ability to do this under any circumstances. So, while it's true that Telegram's individual connections are always encrypted, it's only when two – and only two – parties are simultaneously online and Telegram's users opt to enable end-to-end encryption for that dialog, that any truly unobservable conversation takes place over Telegram. ALL larger group chats are being decrypted by Telegram's servers for re-encryption and sending to other Telegram users. Obviously, this is a crucial distinction. Returning to Matthew's explanation, he says...

While it might seem like I'm being picky, the difference in adoption between default end-to-end encryption and this experience is likely very significant. The practical impact is that the vast majority of one-on-one Telegram conversations — and literally every single group chat — are visible on Telegram's servers, which can see and record the content of all messages sent

between users. That may or may not be a problem for every Telegram user, but it's certainly not something we'd advertise as particularly well encrypted. (If you're interested in the details, as well as a little bit of further criticism of Telegram's actual encryption protocols, I'll get into what we know about that further below.) So does default encryption really matter? Maybe yes, maybe no! There are two different ways to think about this.

One is that Telegram's lack of default encryption is just fine for many people. The reality is that many users don't choose Telegram for encrypted private messaging at all. For plenty of people, Telegram is used more like a social media network than a private messenger. Getting more specific, Telegram has two popular features that makes it ideal for this use-case. One of those is the ability to create and subscribe to "channels", each of which works like a broadcast network where one person (or a small number of people) can push content out to millions of readers. When you're broadcasting messages to thousands of strangers in public, maintaining the secrecy of your chat content isn't as important.

Telegram also supports large public group chats that can include thousands of users. These groups can be made open for the general public to join, or they can set up as invite-only. While I've never personally wanted to share a group chat with thousands of people, I'm told that many people enjoy this feature. In the large and public instantiation, it also doesn't really matter that Telegram group chats are unencrypted — after all, who cares about confidentiality if you're talking in the public square?

But Telegram is not limited to just those features, and many users who join for them will also do other things. Imagine you're in a "public square" having a large group conversation. In that setting there may be no expectation of strong privacy, and so end-to-end encryption doesn't really matter to you. But let's say that you and five friends step out of the square to have a side conversation. Does that conversation deserve strong privacy? It doesn't really matter what you want, because Telegram won't provide it, at least not with encryption that protects you from sharing your content with Telegram servers.

Similarly, imagine you use Telegram for its social media-like features, meaning that you mainly consume content rather than producing it. But one day your friend, who also uses Telegram for similar reasons, notices you're on the platform and decides she wants to send you a private message. Are you concerned about privacy now? And are you each going to manually turn on the "Secret Chat" feature — even though it requires four explicit clicks through hidden menus, and even though it will prevent you from communicating immediately if one of you is offline?

My strong suspicion is that many people who join Telegram for its social media features also end up using it to communicate privately. And I think Telegram knows this, and tends to advertise itself as a "secure messenger" and talk about the platform's encryption features precisely because they know it makes people feel more comfortable. But in practice, I also suspect that very few of those users are actually using Telegram's encryption. Many of those users may not even realize they have to turn encryption on manually, and think they're already using it.

And this brings me to my next point: Telegram knows its encryption is difficult to turn on, and they continue to promote their product as a secure messenger. Telegram's encryption has been subject to heavy criticism since at least 2016 (and possibly earlier) for many of the reasons I

outlined in this post. In fact, many of these criticisms were made by experts including myself, in years-old conversations with Pavel Durov on Twitter.

I'll interject here to note that back in the morning of March 29th, 2015, after Matthew first sat down to take a serious long look at Telegram's encryption protocol & system his Tweet linked to Telegram's page https://core.telegram.org/mtproto/auth_key titled "Creating an Authorization Key" and about what he found there he Tweeted: "Like seriously. What the F is even going on here?" He continues...

Although the interaction with Durov could sometimes be harsh, I still mostly assumed good faith from Telegram back in those days. I believed that Telegram was busy growing their network and that, in time, they would improve the quality and usability of the platform's end-to-end encryption: for example, by activating it as a default, providing support for group chats, and making it possible to start encrypted chats with offline users. I assumed that while Telegram might be a follower rather than a leader, it would eventually reach feature parity with the encryption protocols offered by Signal and WhatsApp. Of course, a second possibility was that Telegram would abandon encryption entirely — and just focus on being a social media platform.

What's actually happened is a lot more confusing to me. Instead of improving the usability of Telegram's end-to-end encryption, the owners of Telegram have more or less kept their encryption UX unchanged since 2016. While there have been a few upgrades to the underlying encryption algorithms used by the platform, the user-facing experience of Secret Chats in 2024 is almost identical to the one you'd have seen eight years ago. This, despite the fact that the number of Telegram users has grown by 7-9x during the same time period.

At the same time, Telegram's CEO and sole owner, Pavel Durov has continued to aggressively market Telegram as a "secure messenger." Most recently he issued a scathing criticism of Signal and WhatsApp on his personal Telegram channel, implying that those systems were backdoored by the US government, and only Telegram's independent encryption protocols were really trustworthy.

While this might be a reasonable nerd-argument if it was taking place between two platforms that both supported default end-to-end encryption, Telegram really has no legs to stand on in this particular discussion. Indeed, it no longer feels amusing to see the Telegram organization urge people away from default-encrypted messengers, while refusing to implement essential features that would widely encrypt their own users' messages. In fact, it's starting to feel a bit malicious.

So what about the boring encryption details? Since this is a cryptography blog I'd be remiss if I didn't spend at least a little bit of time on the boring encryption protocols. I'd also be missing a good opportunity to let my mouth gape open in amazement, which is pretty much what happens every time I look at the internals of Telegram's encryption. I'm going to handle this in one paragraph to reduce the pain, and you can feel free to skip past it if you're not interested.

I'm going to interrupt Matthew again to note that he has laced his description, which follows, with asterisks and later he explains that...

"Every place I put an "" in the paragraph is a point where expert cryptographers would, in the context of something like a professional security audit, raise their hands and ask a lot of questions."* Okay. So Matthew writes:

According to what I think is the latest encryption spec, Telegram's Secret Chats feature is based on a custom protocol called MTProto 2.0. This system uses 2048-bit* finite-field Diffie-Hellman key agreement, with group parameters (I think) chosen by the server.* (Since the Diffie-Hellman protocol is only executed interactively, this is why Secret Chats cannot be set up when one user is offline.*) MITM protection is handled by the end-users, who must compare key fingerprints. There are some weird random nonces provided by the server, which I don't fully understand the purpose of* — and that in the past used to actively make the key exchange totally insecure against a malicious server (but this has long since been fixed.*) The resulting keys are then used to power the most amazing, non-standard authenticated encryption mode ever invented, something called "Infinite Garble Extension" (IGE) based on AES and with SHA2 handling authentication.*

I'm not going to go further than this. Suffice it to say that Telegram's encryption is... unusual.

If you ask me to guess whether the protocol and implementation of Telegram Secret Chats is secure, I would say quite possibly. To be honest though, it doesn't matter how secure something is if people aren't actually using it.

So, is there anything else to know?

Yes, unfortunately. Even though end-to-end encryption is one of the best tools we've developed to prevent data compromise, it is hardly the end of the story. One of the biggest privacy problems in messaging is the availability of loads of meta-data — essentially data about who uses the service, who they talk to, and when they do that talking.

This data is not typically protected by end-to-end encryption. Even in applications that are broadcast-only, such as Telegram's channels, there is plenty of useful metadata available about who is listening to a broadcast. That information alone is valuable to people, as evidenced by the enormous amounts of money that traditional broadcasters spend to collect it. Right now all of that information likely exists on Telegram's servers, where it is available to anyone who wants to collect it.

I am not specifically calling out Telegram for this, since the same problem exists with virtually every other social media network and private messenger. But it should be mentioned, just to avoid leaving you with the conclusion that encryption is all we need.

Okay. So there are many useful bits among what Matthew wrote. One is that while Telegram's crypto is bizarre, on its face it's not obviously insecure. But neither has it ever been shown to be secure. Mostly, it's just bizarre. Or as Matthew put it... What the "F"??

The most important thing for Telegram's users to appreciate is that what Matthew referred to as today's industry-standard encrypted messaging apps provide always-on end-to-end encryption by default, while extending that true end-to-end encryption no matter how many individuals are participating in chat groups.

Remember that last time we talked about iMessage we saw that not only had Apple implemented true multi-party end-to-end messaging, but that iMessage was also offering true forward secrecy by periodically and continuously rolling its conversation keys. iMessage and Signal offer technology that Telegram has never had and, as Matthew noted, shows no sign of obtaining or even wanting. And, indeed, Telegram's popularity may not really be about true security. It's more about subscribing to its channels with a weaker assumption that "Things are secure" only because Telegram also has an unearned reputation for being a secure messaging system.

The reputation is unearned because Telegram is completely unable to make true end-to-end encryption available by default. And while it can be enabled in real-time for chats between exactly two online parties, true end-to-end encryption is **never** available for use by any group. It's clear that over the years Matthew has become increasingly annoyed that Telegram has been misrepresenting the security it offers and that just like Telegram's users, the press has similarly adopted the assumption that Telegram is a secure solution that's comparable, or perhaps even superior, to its alternatives. We know that's not the case.

Matthew began by posing the question: "Is Telegram an Encrypted App?" – The most generous answer would be that while it can be forced to provide state-of-the-art end-to-end encryption between two online parties, it certainly is not "as encrypted" as the general public, its users and the press have come to assume. More than anything else its ability to broadcast to very large groups has turned it into a social media sharing platform with an air of undeserved security and privacy.