780th MILITARY INTELLIGENCE BRIGADE (CYBER)

THE BYTE

Vol. 11, Issue 2

AVENGERCON VII:

Crowdsourcing Conflict

780th MI BDE
"STRENGTH AND HONOR"

COL Benjamin Sangster
Commander
CSM Jesse Potter
Command Sergeant Major

780th MILITARY INTELLIGENCE BRIGADE (CYBER), Fort George G. Meade, Maryland, publishes The BYTE as an official command information publication, authorized under the provisions of AR 360-1, to serve its Soldiers, Civilians, and Families.

Opinions expressed herein do not necessarily represent those of 780th MI BDE or the Department of the Army.

All photographs published in The BYTE were taken by 780th MI BDE Soldiers, Army Civilians, or their Family members, unless otherwise stated.

Send articles, story ideas, photographs, and inquiries to the 780th MI Brigade (Cyber) Public Affairs Officer (PAO) and The BYTE Editor, Steven Stover at steven.p.stover.civ@army.mil, or mail to 310 Chamberlin Avenue, Fort George G. Meade, MD 20755, or call (301) 833-6430.

On the Cover

COLUMBIA, Md. – AvengerCon VII, hosted every fall by Maryland Innovation and Security Institute to benefit the hackers of the U.S. Cyber Command community and the 780th Military Intelligence Brigade (Cyber), returned for a hybrid in-person and virtual event on November 30th and December 1st. Training sessions on November 30 included "Intro to Reverse Engineering" instructed by Jeremy Hawthorne, lead instructor for the Boston Cybernetics Institute (BCI) where he develops training for the U.S. military.

THIS EDITION OF THE BYTE MAGAZINE focuses on AvengerCon VII and features a few of the presentations.

AvengerCon, now in its seventh year, is hosted every fall by Maryland Innovation and Security Institute to benefit the hackers of the U.S. Cyber Command community and the U.S. Army 780th Military Intelligence Brigade. The event is open to all service members and employees of U.S. Cyber Command and Department of Defense personnel supporting cyberspace missions. AvengerCon features presentations, hacker villages, training workshops, and much more.

A special thanks to AvengerCon VII keynote speakers, Mudge and Sarah Zatko, as well as panelists, Colin Ahern, Katie Moussouris, and TJ O'Connor, and all the speakers, village hosts, and workshop instructors for a fantastic conference! Additionally, a special thanks to Mark Pomerleau, DefenseScoop, for publishing an article *AvengerCon military hacker conference to examine crowdsourcing conflict*.

Also in this edition of The BYTE, we pay our respects to CPT Hunter Phillips, 11th Cyber Battalion, who passed away from natural causes. A memorial service for CPT Phillips was held on February 15 at Fort Gordon, GA. The passing of a teammate is tragic, especially for those who knew him. We encourage each of you to support one another and to seek comfort from friends and Family in this time of mourning.

Praetorians! Strength and Honor

v/r,
Steve Stover
Public Affairs Officer
780th MI Brigade (Cyber)
Editor, The BYTE

AvengerCon VII – Back and Better Than Ever

By CPT Jacob Heybey, Cyber, AvengerCon VII Lead, 780th Military Intelligence Brigade (Cyber)

AFTER TWO YEARS OF COVID-19 RESTRICTIONS, AvengerCon is back and better than ever! While the community continued to host a fantastic event – virtually – over the past couple years, there was a palpable sense of relief when AvengerCon VII[1] was given the go-ahead to conduct a full in-person event with no limitations. Old standbys like the AvengerCon lock picking village returned in full force, and other events like an RF village, an Arduino-based escape room, and several social events took advantage of AvengerCon's return to in-person activities.

Though we were thrilled to return to unrestricted in-person activities, we kept some practices from the so-called "pandemic years". When possible, we made presentations, certain villages, and some workshops available to remote attendees through streaming services. This year, AvengerCon welcomed attendees based out of Georgia, Texas, Colorado, and Germany. We also improved our ability to publish presentations approved for public release – AvengerCon VII's presentations are available at https://www.dvidshub.net/tags/video/avengercon-vii. Finally, AvengerCon VII saw some first steps towards creating satellite sites; volunteers at Fort Gordon, Georgia, organized a small watch party for remote attendees to socialize in-person.

One of AvengerCon's biggest draws every year is our keynote speaker, and it was a fantastic experience to see our co-speakers this year. Famed security expert Peiter "Mudge" Zatko and noted mathematician Sarah Zatko attempted to bust some common cybersecurity myths through examining data. Combining a mix of statistical analysis, game theory, and a refusal to accept (seemingly) common sense takes without supporting data, they came to some surprising conclusions about what best practices in cybersecurity should be.

We owe the Zatkos a huge thank you for taking the time to speak at AvengerCon and sharing their years of expertise with our attendees.

For this year's theme, AvengerCon VII volunteers drew inspiration from current events in Ukraine. Shortly after Russia's invasion, Ukraine created a volunteer "IT Army" and encouraged online supporters worldwide to participate in a pro-Ukraine campaign using any means at their disposal – including information operations, open source intelligence (OSINT) analysis, and offensive cyber operations. Labeling this concept "crowdsourcing conflict", we asked our speakers and attendees to consider the ramifications of this phenomenon. While Ukraine seemingly had some public successes with this approach, there are risks and limitations: how can volunteers be vetted and coordinated? Should volunteers be considered combatants? Can we accurately assess the effectiveness of these volunteer-driven activities?

We dove deeper into the subject with our panel discussion: Colin Ahern (N.Y. State Chief of Cyber), Katie Moussouris (Luta Security), and TJ O'Connor (Florida Tech) tackled these questions from a variety of perspectives. Their discussion covered what defenders in such a situation are worried about (Colin Ahern commented on the democratization of cyber threats), the organizational characteristics needed to leverage the wider community (mature and well-tested security processes, according to Katie Moussouris), and whether the US should ever consider similar tactics (a resounding "no" due to potential unintended effects, per TJ O'Connor).

Each panelist brought a variety of experiences and background to the discussion, and we were fortunate to be able to gather Mr. Ahern, Ms. Moussouris, and Prof. O'Connor for our panel. Thank you again to our panelists for being generous with both their time and knowledge.

As ever, AvengerCon is only as good as our volunteers and community make it. I don't have enough space to list everyone who deserves recognition, but I can try to name most of them.

First, thank you to the Joint (Army, Navy, and Space Force!) military volunteers who invested their personal time into the making of AvengerCon this year:

- MAJ Neil Milchak (communications and advertising lead)
- Lt. (USN) Zechariah Clark, 1LT Calvin Kim (workshops)
- CPT Brian Maguire, SSG Hayden Brown (villages)
- Mr. Niccolo Hartley, 1LT Steve Cevallos, SSG Aisha Umar, SSG Nicholas Camp, CPT (USSF) Mat Boston (infrastructure)
- SSG Jiseng So (A/V support and video editing)
- CDT (Virginia Tech) Grant Smith, CPT Olivia Brundage, SFC Craig Seiler (presentations MCs)
- SFC Robert Jaramillo (event staff coordinator)
- WO1 Andrew Fricke (website maintenance)
- MAJ Stephen Rogacki (panel moderator)
- 1LT Kimi Walker, 1LT Amanda Roper, 1LT Adrian Naaktgeboren (satellite site)
- CW2 Jacob Worley (CTF event)

Additionally, students from Florida Tech volunteered their time to write challenges for the AvengerCon CTF event, so thanks to them as well.

Our speakers, village hosts, and workshop instructors produce the "meat" of AvengerCon. We welcomed speakers, hosts, and instructors from within the U.S. Cyber Command (USCYBERCOM) community, sister units, and industry partners. (Our most far-flung speaker this year delivered his talk from New Zealand.) Your collective preparation and materials continue to make AvengerCon a world-class conference. And a special thanks to those who contributed articles to this

1

edition of the Byte; make sure to read on to see more from our AvengerCon VII speakers.

Support from the Maryland Institute for Security Institute (MISI) through its partnership with the USCYBERCOM J9 has been invaluable in allowing AvengerCon to grow both physically and virtually. Karissa Kelly-Seckman, Mary Councill, Everett Flanders, Joseph Ramirez, Marcus DiMucci, and the rest of the MISI Dreamport team were essential to AvengerCon VII.

Besides our main volunteers, the staff of the 780th Military Intelligence Brigade (Cyber) also supported AvengerCon. In particular, the Brigade Public Affairs Officer (PAO), Steve Stover, did excellent work publicizing and documenting the event.

Last, but certainly not least, thank you to all AvengerCon attendees! Your participation and enjoyment of the conference make our efforts as volunteers worth it. If you liked AvengerCon, please consider volunteering next year – we are always looking for new volunteers to help us run the conference. You can reach out to me at jacob.o.heybey.mil@army.mil or send a message to our contact form at https://avengercon.com/contact.

I first experienced AvengerCon as a newcomer to the 780th MI Brigade back in 2017. That version of AvengerCon looked very different from this year's event. It was entirely contained in one room in McGill Training Center on Fort Meade, the vast majority of attendees were from the brigade itself, and there were no workshops, book signings, or CTF events (though AvengerCon's venerable Crypto Challenge was a popular event at the time). Despite its comparatively humble setting, this early AvengerCon was eye-opening for me as a young officer, broadening my horizons and providing me a way to engage with technical subjects in an informal environment. I volunteered to help plan AvengerCon in the hopes of giving other members of the community the same experience, and I hope AvengerCon continues to fill that niche for years to come!

References:

[1] *https://avengercon.com/theme/*

Post-Quantum Cryptography and U.S. Government Activities

By CSM Samuel Crislip, Command Sergeant Major, 782d Military Intelligence Battalion (Cyber)

IF WE IMAGINE A WORLD WHERE ENCRYPTION NO LONGER EXISTS. What would that look like from a financial perspective, for your personal accounting, for global markets, for businesses that thrive on intellectual property, or for any business with which you regularly conduct transactions? What would that mean for your personal information, your social media, or your intimate records? What would this mean from a national security perspective, where every military vehicle schematic, training doctrine, battle plan, personnel roster, unit structure, or even social roster is accessible to anyone?

This is the thesis of a quantum computing breakthrough that nation states and large technology companies are actively pursuing. Quantum computers are systems which utilize light particles to process or compute data at a level of complexity and speed that traditional computers will not be able to achieve. With the power a quantum computer breakthrough promises, we should see leaps in the fields of data and material sciences, drug and medical research, complex system optimization, artificial intelligence, and cryptography.

Current Encryption

In the context of cryptography, current standards rely on both symmetric and asymmetric encryption, depending on the requirement for encryption. Data at rest generally relies on symmetric encryption, which utilizes the same key to encrypt and decrypt data. While this possibly seems prone to brute-force attacks, when utilizing an AES encryption, a standard for protecting government classified information, it is highly resistant to current processing capabilities. With AES 256, the encryption key is a 256 bit number that undergoes 14 rounds of processing that includes mixing the number, transposing it, and/or substitution of bits (Bernstein & Cobb, n.d.). These processing steps provide the necessary complexity to avoid brute-force attacks at the cost of requiring more processing power. This is why it is primarily used for data at rest.

Data in transit uses less processing complexity with asymmetric encryption, which utilizes a public and private key for encryption. RSA encryption is one of the primary standards at this time, implementing 2048 bit prime numbers that require factoring for its public and private keys. With prime factoring at this size, current computing standards are incapable of breaking the decryption keys. Another popular standard is Diffie-Hellman which applies discrete logs or elliptical curves, even more complex mathematical processes to attack, to create its encryption standards (Lake, 2021).

We depend on each of these standards along with others to keep our information, data, and secrets secure from those who would use it against us or for nefarious reasons. The assumption is that today's systems simply do not have the processing capacity to defeat the complex mathematical algorithms on which these standards focus. Unfortunately, this creates two distinct concerns regarding what the future holds within the realm of what is possible with quantum processing and the risk our current data faces if it is captured today to be unlocked in that near future.

Download Now, Decrypt Later

Among the amount of information you want to protect, most people would likely want to protect that information for a lifetime or longer, especially if it may be considered financially or personally damaging. The government sets declassification standards for the information it wants to protect to around 10 years, but they may protect more sensitive data for up to 25 years (Basic Laws and Authorities | National Archives, n.d.). However, a quantum computing breakthrough may happen without significant forewarning, making any of those protection guidelines and personal security ideas void. Moreover, our adversaries are actively collecting anything they want today with the intention of waiting for that crucial step. Consider any zero-day attack, until that attack took place, no one knew it was possible. If we apply this logic across any information we seek to protect, we may find ourselves in a much more nervous state about tomorrow's data processing achievements. Therefore, it is essential we start preparing today for tomorrow's attack. If we use Mosca's Theorem, we can note that if we combine the time we need to protect some information (A) with the time necessary to update security standards (B) and find that the time (A+B) exceeds the time projected to reach a post-quantum world (A+B > PQ), then we are already in a risk environment (Relyea, 2022). So it is relevant to understand where we are now with regards to securing data with new encryption standards.

Post-Quantum Cryptography: Government Effort

In 2017, the National Institute of Standards and Technology (NIST) utilized the 2002 Federal Information Security Management Act to push a call for proposals for post-quantum cryptography standardization, focusing on quantum safe or resistant cryptographic algorithms, known as post-quantum cryptography (PQC). NIST based this request on the concern that quantum computers were proving capable of quantum entanglement, fault tolerance, and error correction while also realizing that integration of new cryptographic standards would be a complex obstacle. Therefore, NIST published a multi-round submission process for PQC algorithms, focusing on security, cost, performance, maturity, and implementation to demonstrate proof of concept for the

various cryptographic ideas (National Institute of Standards and Technology, 2017). As NIST evaluated submissions, it found submissions were either code-based, multivariate, lattice-based, hash-based, or isogeny-based cryptography and narrowed the field down from over fifty in Round 1 down to four variations by Round 4. On 5 July 2022, NIST announced its selection of one public-key encryption and key-establishment algorithm and three digital signature algorithms for standardization of PQC, paving the way forward for implementation of PQC (Selected Algorithms 2022 - Post-Quantum Cryptography, 2022).

Throughout the push to find viable post-quantum encryption options, NIST worked with the Department of Homeland Security (DHS) to create a "roadmap" for a whole of government integration of next generation encryption. The two organizations focused on key elements to support customer efforts in preparing for a future of quantum computing, focusing on standards development, inventories of current systems and requirements, internal standardization requirements, prioritization, and planning (Department of Homeland Security, 2021). While not all inclusive, it provides a necessary framework to create a whole of government understanding of where we are and where we need to go to prepare for a change in encryption and cryptography.

Government PQC Implementation

The government is already demonstrating its drive to incorporate post-quantum cryptography in its partnership with QuSecure as it implements QuProtect, an end-to-end cryptography solution that uses software to provide quantum resilience and security. The government has placed QuProtect on some legacy systems for the Air Force, Space Force, and North American Aerospace Defense Command (NORAD), with promising results. They have not seen an increase in either bandwidth or latency while fully employing the system in place of previous encryption methods. They have the system operating over open internet channels, applying the cryptographic standards the NIST approved during its PQC standardization competition. QuProtect has not announced any security breaches of its technology, providing an early indication of its capabilities in the realm of security (HPCwire, 2022).

Over time it is likely we will see the government scale up its efforts to implement PQC across its variety of networks, especially as efforts grow to defeat current encryption standards. China recently bragged about breaking RSA encryption utilizing a quantum machine. They claim their 10-qubit system was able to defeat a 48-bit RSA encryption scheme (Martin, 2023). While evidence would suggest that this achievement does not mean RSA encryption or other standards are in immediate danger given the scheme China broke is not a relevant security protocol, it still demonstrates a movement towards a future of insecurity.

Why Does It Matter?

Skepticism exists of the reality or likelihood of anyone ever achieving a usable quantum computer. This would suggest that any concern of a post-quantum world would never materialize and we, collectively as a global society and effort, would have wasted decades on this non-viable technology. In turn, we are wasting valuable time developing PQC standards when researchers could be focused on goals that may better benefit society. However, where is the harm or the fault in creating a better security standard for today's complex and interwoven cyber society? We find ourselves increasingly reliant on computers, networks, and emerging technologies. Every new device we build and incorporate into daily life introduces an even greater number of potential vulnerabilities for breaching privacy and security. As threat vectors increase, we have an obligation to ensure a robust protection system to protect from security breaches and privacy leaks. Further, it is paramount that we are able to secure critical infrastructure that society relies on to live in our complex environments. Therefore, we should be doing all we can to continue to improve security standards, especially cryptographic schemes, regardless of hypothetical emerging threats. We would be far worse off if that threat came to fruition and we had done nothing to prepare. to new jobs in the next few years but as military leaders come and go we do still truly represent continuity for the mission and for the culture. If 780th is not a place where you enjoy coming to work what are you doing to make it better for yourself, and for those who come after us?

Are we helping to create the change we crave?

References:

Basic Laws and Authorities | National Archives. (n.d.). National Archives |. Retrieved October 31, 2022, from https://www.archives.gov/about/laws/appendix/12958.html

Bernstein, C., & Cobb, M. (n.d.). What is the Advanced Encryption Standard (AES)? Definition from SearchSecurity. TechTarget. Retrieved October 31, 2022, from https://www.techtarget.com/searchsecurity/definition/Advanced-Encryption-Standard

Department of Homeland Security. (2021, October 1). Preparing for Post-Quantum Cryptography Infographic. Retrieved November 18, 2022, from https://www.dhs.gov/sites/default/files/publications/post-quantum_cryptography_infographic_october_2021_508.pdf

HPCwire. (2022, July 12). US Gov and QuSecure Orchestrate Post-Quantum Encryption Communication Over a Government Network. HPCwire. Retrieved November 21, 2022, from https://www.hpcwire.com/off-the-wire/us-gov-and-qusecure-orchestrate-post-quantum-encryption-communication/

Lake, J. (2021, March 23). What is the Diffie–Hellman key exchange and how does it work? Comparitech. Retrieved October 31, 2022, from https://www.comparitech.com/blog/information-security/diffie-hellman-key-exchange/

Martin, A. (2023, January 4). Chinese researchers claim to have broken RSA with a quantum computer. Experts aren't so sure. The Record by Recorded Future. Retrieved January 23, 2023, from https://therecord.media/chinese-researchers-claim-to-have-broken-rsa-with-a-quantum-computer-experts-arent-so-sure/

National Institute of Standards and Technology. (2017, January 3). 1 Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process. https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf

Relyea, R. (2022, June 15). Post-quantum cryptography, an introduction. Red Hat. Retrieved October 31, 2022, from https://www.redhat.com/en/blog/post-quantum-cryptography-introduction

Selected Algorithms 2022 - Post-Quantum Cryptography. (2022, November 1). Post-Quantum Cryptography | CSRC. Retrieved November 9, 2022, from https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022

Attacking the Brain: Adversarial Artificial Intelligence

By MAJ George Sieretzki, U.S. Cyber Command ARE

THE RECENT, DRAMATIC, AND HIGHLY PUBLICIZED ADVANCES IN ARTIFICIAL INTELLIGENCE (AI) have thrust the astonishing power of AI into the mainstream consciousness. Tech giants such as Google, Microsoft, and IBM are all investing heavily in machine learning, and the results threaten to revolutionize the world we live in. Current research is producing applications that will disrupt traditional business models across a variety of industries, including healthcare, education, robotics, transportation, finance, customer service, and media content creation, - to name just a few.

As impressive as commercial AI capabilities are becoming, the impact on military operations promises to be just as dramatic. Organizations, such as the Defense Advanced Research Project Agency (DARPA) have been on the forefront of this research for decades, and there is no doubt that the near future will see widespread integration of AI in autonomous weapons systems, surveillance and reconnaissance, cybersecurity, predictive maintenance, battlefield decisions making, and one particularly interesting field of study: Adversarial AI.

Adversarial AI – The Dark Side

Adversarial AI is the study of how AI systems can be intentionally fooled or manipulated. This has become a rapidly growing area of research, especially when applied to security-critical applications where the consequences of misclassification or manipulation can be severe, and in military applications, even fatal.

"Adversarial Examples" Attacks

Adversarial examples are inputs to AI systems that have been deliberately altered to cause the model to make incorrect predictions. These types of attacks can be leveraged to fool facial recognition systems, speech recognition systems, text recognition, and even deep reinforcement learning agents.

For example, adding small, carefully crafted changes to an image or text, may cause a highly accurate classification model to misclassify the data. An "adversarial text attack" might be conducted by a change to the number "7" that is imperceptible to humans, in a system that is trained to recognize hand-written digits, and resulting in the model predicting a "9" instead. Another technique is an "adversarial image attack", which could be implemented by a slight modification of an image of a stop sign that results in a self-driving vehicle misinterpreting and ignoring the sign altogether.

"Poisoning" Attacks

Poisoning attacks involve inserting malicious data into the training data used to build an AI model, causing the model to make incorrect predictions on specific inputs in the future. This can be achieved by manipulating the labels of the training data, adding or removing data points, or by changing the distribution of the data.

A personal example stemming from a vendor presentation from as far back as 2001, saw a network penetration test team developing techniques for evading an AI driven intrusion detection system (IDS) by slowly altering its understanding of what "normal" network traffic looked like. By injecting small, incremental increases in anomalous packets over time during the learning phase of the AI, the AI would eventually be taught to classify voluminous network scans and blatant network-based attacks as a normal occurrence, and thus fail to alert.

"Model Inversion" Attacks

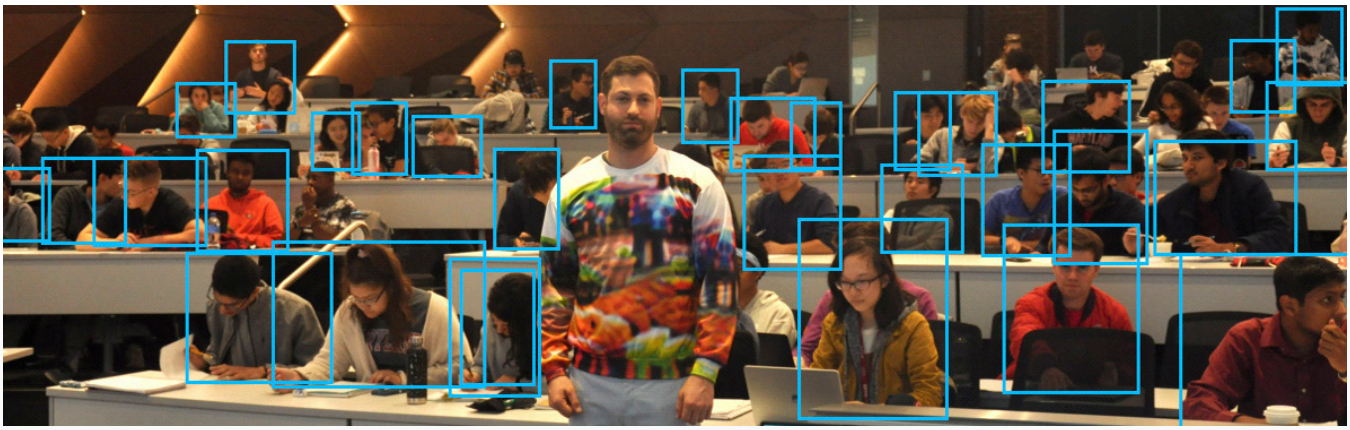
Model inversion attacks involve using an AI model to reverse engineer the private information or data used to train the model. For instance, an attacker could use a model inversion attack to infer personal characteristics of individuals who were included in the training data of a facial recognition system.

At the 2020 USENIX Security Symposium, researchers demonstrated an attack on GPT-2, a language model trained on scrapes of the public Internet. The experiments showed that it was possible to extract hundreds of verbatim text sequences from the model's training data, including personally identifiable information (names, phone numbers, and email addresses), IRC conversations, and source code.

"Evasion" Attacks

Evasion attacks involve an adversary trying to evade detection by a machine learning classifier. For example, by changing the color or shape of a weapon in an image, an attacker may be able to evade a classifier trained to detect weapons. These attacks can also be used to evade spam filters, intrusion detection systems, and other security systems that rely on machine learning.

In a recent example, the University of Maryland Computer Science department demonstrated how a sweater, which had been imprinted with images, could be used as an "invisibility cloak" to shield the wearer from an AI powered "classifier" trained to identify and/or track humans. While the AI was highly effective in successfully identifying all people in a video, the AI was unable to recognize the wearer of the sweater as a person.



University of Maryland "Invisibility Cloak" at https://www.cs.umd.edu/~tomg/projects/invisible/

Impact on Cyber Operations

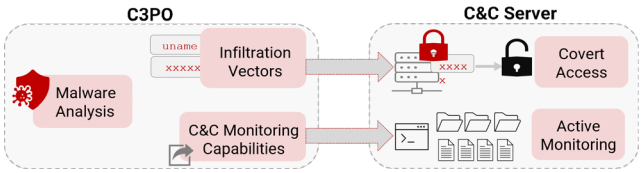
Adversarial AI represents a new, rapidly evolving attack surface for cyber operators to both defend and exploit. While offensive cyber operations have traditionally aimed to either compromise or deny the use of an opponent's critical data, future missions might be accomplished by causing subtle changes in the integrity of data to subvert the logic of decision-making AI. And on a future AI-driven battlefield, the results of such attacks could have far more destructive consequences than traditional data theft or denial of service. When AI matures to the point where it autonomously controls the kinetic application of military force, protecting the "brain" may become the highest priority for our future network defenders. ■

Exploiting Malware Communication Protocols for Command-and-Control Server Infiltration

By MAJ Jonathan Fuller, PhD, Army Cyber Institute, United States Military Academy

BOTNET DISRUPTIONS AND TAKEDOWNS ARE DRIVEN BY COMMAND AND CONTROL (C&C) server monitoring before any action is taken and after to gauge success. This means that disruption or takedown attempts are not only provably necessary but must be targeted and effective. Existing approaches can be categorized as passive or active monitoring. Passive monitoring (e.g., sensor node injection) is coarse-grained and may not give accurate insights into the botnet, i.e., the number and location of the victims and the extent of damages incurred. It also requires a full reverse engineering effort to maintain sensor nodes making this approach not widely used. Conversely, active monitoring may provide better insights into botnet operations. However, active monitoring techniques, including remote penetration testing and domain seizure, are noisy making them easily detectable. Seeking a better solution, we propose that standard protocols, which are increasingly used by botnets, can be leveraged for general and covert C&C server monitoring.

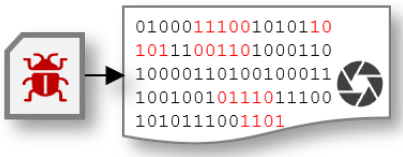
In previous botnet disruption and takedown attempts, authorities first monitored the C&C server to prove malware as the catalyst for incurred damages before legal permission was granted for counteraction. Yet, accurate monitoring goes beyond determining the legality of counteraction. For example, to protect the 2020 election, Microsoft took down 120 Trickbot C&C servers[1]. Accurately identifying C&C servers pre-takedown (profiling), then tracking successes post takedown (validation), required an in-depth understanding of the peers in the botnet, C&C server locations, and weaknesses to leverage for botnet disruption. Therefore, successful monitoring must result in accurate, legally admissible information gathered during profiling and remain covert to avoid discovery by C&C orchestrators, prompting defensive evasion or hardening. An ideal solution should provide authorities with a means to access the C&C server under the guise of normal bot operation.



As the end-host agents of a C&C orchestrator, bots are entrusted with C&C server access. In fact, attackers are entirely dependent on the information exfiltrated by bots to gain situational awareness in a victim's network. To enable command and control, bots use standard protocols for file transfer, data storage, and message-based communication. However, many standard protocols are bloated, meaning that they provide feature-rich and unfettered access to the server beyond the subset of features implemented by a given client. A similar trend is often observed in benign software where bloated client-side protocols lead to unauthorized server access[2]. This prompted our key insight: *bloated protocols combined with the trust* C&C *servers place in their bots expose a scalable opportunity for covert monitoring of* C&C *servers through protocol infiltration.*

To explore this insight, we turned our attention to how the authorities could recover C&C server access privileges from bloated bots (bots using bloated protocols) allowing them to spoof bot-to-C&C communication. To this end, we designed and implemented C3PO[3], an automated memory-image-based symbolic analysis measurement pipeline. C3PO, analyzes a malware memory image to identify (1) bloated protocols, (2) infiltration vectors (i.e., authentication information to spoof bot-to-C&C communication for covert access), and (3) C&C monitoring capabilities (i.e., capabilities in the end-host bot that reveal the C&C server's composition and content to guide active monitoring).

Malware Analysis via Memory Image Extraction. Leveraging Detours[4], C3PO extracts the memory image during malware execution by hooking network-related APIs. This is based on a key observation: Irrespective of the packing scheme, after unpacking, the malware must invoke APIs to interact with its C&C server. Therefore, C3PO extracts multiple memory images by hooking all network-related APIs using a trampoline which replaces instructions in the hooked API with a call to our custom code. This custom code writes a memory image to a file and returns to the trampoline to continue execution as normal. After extracting the memory images, C3PO conducts memory-image-based analysis to measure the prevalence of bloated protocol use.



Bloated Protocol Identification. Standard protocols are often used for: (1) file transfer, (2) data storage, and (3) message-based communication. However, their ubiquitous integration into benign software has prompted research into inherent vulnerabilities, which has led to unauthorized server access. Noticing a similar trend in malware, we select a common bloated protocol, the File

Transfer Protocol (FTP). Based on the protocol, we constructed a database of all protocol identifiers for C3PO to reference during protocol identification. Protocol identifiers include APIs (e.g., FTPPutFile) or keywords/tokens (e.g., mget). Based on the protocol implementations and the database as a reference, C3PO extends the angr[5] binary analysis framework to construct a control flow graph (CFG) of each malware memory image from the point the memory image was taken to all reachable code. C3PO then stitches the CFGs together at overlapping code, ensuring no duplication. Then, C3PO traverses the combined CFG to identify invocations of protocol use, i.e., their identifiers. After FTP is identified, C3PO continues the analysis to identify information that can be used to spoof bot-to-C&C communication toward covert access.

Infiltration Vector Retrieval. Infiltration vectors (IVs) are the credentials used by the bot to connect to the C&C server. To spoof bot-to-C&C communication, C3PO identifies IVs using program slicing and symbolic execution. Slicing is used to partition the combined CFG revealing only the areas relating to bloated protocol use. This reduces the computational complexity by limiting the available execution paths. C3PO explores along the slice by mutating branch predicates using symbolic data. As C3PO progresses through the malware, it will encounter concrete data including the credentials used to connect to the C&C server via the bloated protocol. For example, when a malware uses FTP, it first establishes a secure connection to its C&C server using InternetConnectA. The arguments of interest in this API are the server address, port, username, and password. Thus, C3PO dereferences data buffer pointers corresponding to the arguments of interests to capture the IVs. Using the IVs, C3PO can spoof bot-to-C&C communication and masquerade as a trusted bot.

C&C Monitoring Capabilities Identification. Bots execute capabilities on the infected systems, some of which can be leveraged to provide targeted and active monitoring. These C&C monitoring capabilities exfiltrate victim data and are valuable because they alert the authorities about the types and format of data stored on the C&C server. C3PO again leverages its slicing technique to identify execution paths that lead to data exfiltration APIs (e.g., send). Then, C3PO symbolically executes along data exfiltration slice to identify all additional APIs that precede the data exfiltration API. In this way, C3PO gathers a list of APIs-in-sequence that influence the contents of the exfiltrated data. These sequences of APIs are then compared against the capability models to identify the C&C monitoring capabilities. The capability models, listed in the table below, are derived by manually reverse engineering known malware and by using insights from security reports identified through MITRE ATT&CK.[6]

Category	C&C Monitoring Capabilities
Browser Password Stealing	(1) Mozilla Stealer Chrome Stealer Internet Explorer Stealer
Service Password Stealing	(2) WiFi Stealer Kerberos Stealer Windows System Stealer
Victim Profiling	(3) Registry-stored System Details Live System Operating State System OS Details Victim Locale Information
Spying, Live Monitoring	(4) Keylogger Screen Capture Audio Capture
File Exfiltration	(5) High-level Protocols Raw Socket Transfer

C3PO Deployed. We deployed C3PO against the Detplock malware, a remote access trojan that allows the bot orchestrator to execute commands on the infected machines. For an ethical approach, we follow the precedence established in previous works while exposing the weaknesses that make C&C servers vulnerable to infiltration. Besides, Burnstein[7] provides legal and ethical conduct for cybersecurity research, arguing that injecting traffic into C&C servers can be considered consent when using the communication channel the bot orchestrators provided to the enslaved systems. Similarly, we use the bot-to-C&C channel and the authentication details provided to us through the Detplock malware. Moreover, after verifying access permissions we (1) only retrieve the metadata (e.g., file quantity, etc.) of the service being investigated and (2) perform no write operations. We emphasize that we do not exploit, disrupt, or attempt takedown of the C&C server, avoiding any claim of tortious interference.

The table below summarizes C3PO's covert monitoring results C3PO extracted IVs such as the username, password, server address, and port number. This C&C server responds to FTP queries, which we used to only catalog file metadata, enumerating directories, keeping count of the number of directories and files as well as file extensions and file sizes. Overall, we identified approximately 640MB of data including over 2,500 files across 47 directories. Of the 31 file extensions found, the most common extensions were PNG (44%), HTML (34%), TXT (8%), and EXE (6%). C3PO also identified Victim Profiling, Live Monitoring, and File Exfiltration capabilities. From covert monitoring, C3PO discovered many PNG files, which was expected since its analysis showed that Detplock performed Live Monitoring by taking screenshots. Lastly, C3PO found malicious files on the C&C server's download directory, confirming that Detplock spreads other malicious payloads. Specifically, 7 Windows EXE and 2 BIN files contained suspicious metadata. Their SHA256 signatures revealed ASPack v2.12 packing and a search on VirusTotal revealed more than 2 anti-virus engine detections.

Protocol	FTP
Infiltration Vectors	**Username**: eg{***} **Password**: vrg{***} **Server**: {***}.co.kr **Port**: 21
C&C Monitoring Capabilities	Victim Profiling, Live Monitoring, and File Exfiltration
Covert Monitoring Outputs	(1) PNG files confirming the live monitoring capability (2) 9 malicious executables and binaries

Final Thoughts. Microsoft analyzed 61,000 Trickbot samples[8] to identify communication routines and locate C&C servers for take down. This staggering analysis was only performed because Trickbot was posed to interfere with US elections. Understandably so, it is evident that resources are employed only when the botnet is large enough to warrant a response. Should organizations or authorities wait to counteract botnets only when threatened with considerable negative effects? Are there other approaches to lower the cost of entry into botnet counteraction? Some existing works do provide the means to target botnets for disruption. For example, Malvuln[9] is a malware exploit repository that lists numerous malware hardcoded cleartext credentials. However, each exploit is manually derived, a prohibitively tedious task. Moreover, existing botnet counteraction methods likewise rely on intensive and manual malware investigations per malware family. Thus, we propose that our techniques, as implemented in C3PO, provides authorities a scalable means to infiltrate and monitor C&C servers, a fundamental enabler of botnet disruption and takedown.

The views expressed herein are those of the author and do not reflect the position of the United States Military Academy, the Department of the Army, or the Department of Defense.

The full details of C3PO can be found in the paper "C3PO: Large-Scale Study of Covert Monitoring of C&C *Servers via Over-Permissioned Protocol Infiltration"[10]*

References:

[1]*https://blogs.microsoft.com/on-the-issues/2020/10/20/trickbot-ransomware-disruption-update/*

[2]*http://www.infosecwriters.com/Papers/AMandal_Thick_Client_Application_Security.pdf*

[3]Covert Monitoring of C&C Server via Protocol InfiltratiOn

[4]*https://www.microsoft.com/e n-us/research/project/detours/*

[5]*https://angr.io/*

[6]*https://attack.mitre.org/*

[7]Burstein, Aaron J."Conducting Cybersecurity Research Legally and Ethically."LEET 8 (2008): 1-8.

[8]*https://blogs.microsoft.com/on-the-issues/2020/10/12/trickbot-ransomware-cyberthreat-us-elections/*

[9]*https://www.malvuln.com/*

[10]Jonathan Fuller, Ranjita Pai Kasturi, Amit K. Sikder, Haichuan Xu, Berat Arik, Vivek Verma, Ehsan Asdar, and Brendan Saltaformaggio, "C3PO: Large-Scale Study Of Covert Monitoring of C&C Servers via Over-Permissioned Protocol Infiltration, "In Proceedings of the 28th ACM Conference on Computer and Communications Security (CCS 21), Seoul, South Korea, 2021.

get_reveng() – Exploring an Approach to Reverse Engineering

By MAJ Austyn Krutsinger, USARPAC G3 CEMA

IMAGINE YOURSELF AS A YOUNG, curious 6-year-old child. As you see the world, nearly everything is a new experience. Your inquisitive mind constantly seeks to find truth and understanding. Imagine you see a small little quadcopter hovering overhead, zipping left and right like a hummingbird in flight. Your mind would start racing, trying to figure out why the drone decided to move. How does it know where to go? Does it get tired? Does something tell it where to go? How does the drone hear the instructions? The questions could go on.

Now bring yourself back to the present. You see the same drone hovering overhead, and your experiences tell you someone is probably flying it with a remote control. You may wonder where they bought the drone or how much it costs, but you might not think through the same questions as your younger self. How does the drone hear what the controller is saying? What language does the drone speak? Will the drone listen to me instead of the controller? As we get older, we sometimes lose our natural curiosities. The topic of reverse engineering radio signals may not help answer all of the universe's questions. Still, I hope to help lower the barrier to entry by developing an understanding of reverse engineering. My goal is to inspire and (re)invigorate your curiosity

Defining the Problem

Before diving into reverse engineering, we must set and understand our goal. Do we simply want to understand what frequency the signal is transmitted so we can replay it? Do we want to know what data the transmission contains? Are we okay with only looking at the RF energy, or do we want to look at the signal generation source in some firmware? While there are many paths we could go down, our goal is to *understand the data contained in the charge port remote transmission so we can synthesize our signal.*

Understanding the data contained in the transmission is quite a large and abstract goal. How do we solve this problem? We solve it the same way as any complex task: decomposing the problem into smaller, more atomic problems. When we take the time to understand our goal and break down the problem, we often find that each smaller sub-problem is much easier, sometimes even trivial, to solve yet still contributes to our overall goal. Looking at our goal of *understanding the charge port remote transmission data*, we know we'll need to capture the signal of interest. To capture the signal, we'll need some kind of hardware, such as a software-defined radio, and some software to interact with our software-defined radio. We'll need to know the charge port's center frequency and the signal's bandwidth. These, of course, are not all the sub-problems we need to solve to reach our ultimate goal. However, we can already start seeing the benefit and ease of solving these smaller problems. Figure 1 shows an example mind map to help decompose our goal into smaller parts. While this example may be trivial to some, the idea and process apply to solving any complex problem.

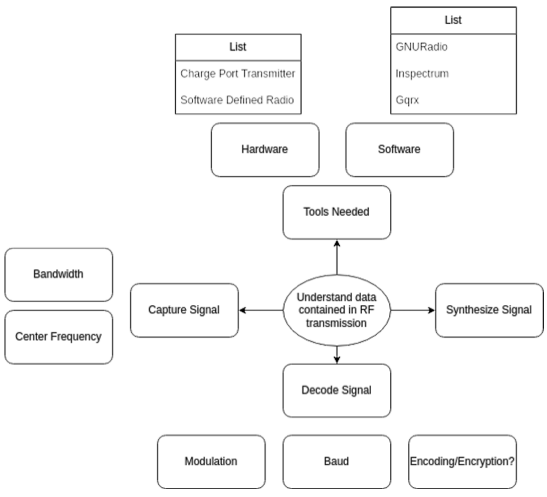


Figure 1 Example mind map of problem decomposition

Gather Tools

Now that we have our goal and an idea of what we need to accomplish to reach our goal, we can start gathering the necessary tools. We could use many hardware and software to solve this problem. Here I'll just use some of the more popular options. The main motivation is that you can find many tutorials online for many of these, and the software is free of cost. We need the transmitter[1] and a way to receive the signal from the transmitter. We use a software-defined radio (SDR), specifically the HackRF One[2]. We need software on the computer to interact with the SDR to view the spectrum and save the raw in-phase/quadrature[3] (IQ) representation of the signal. In this write-up, we're using Gqrx[4]. After we capture the signal of interest, we will use Inspectrum[5] to analyze the signal, then finally, GNU Radio[6] to synthesize our signal.

Again, there are many different, and probably better, ways to approach this problem and different software and hardware solutions. What I've outlined here was chosen because of their popularity and ubiquity.

Capturing the Signal

Start by opening up *Gqrx* and configuring the I/O device. You shouldn't need to change anything, but you can decrease the sample rate. It usually defaults to 8,000,000, which is fine, but *Gqrx* kept freezing during my captures. I finally got some captures when I decreased the sample rate to 4,000,000. I don't know if that is directly part of the problem, but it seemed to be a good-enough interim solution. Others[7] have experienced this type of problem too.

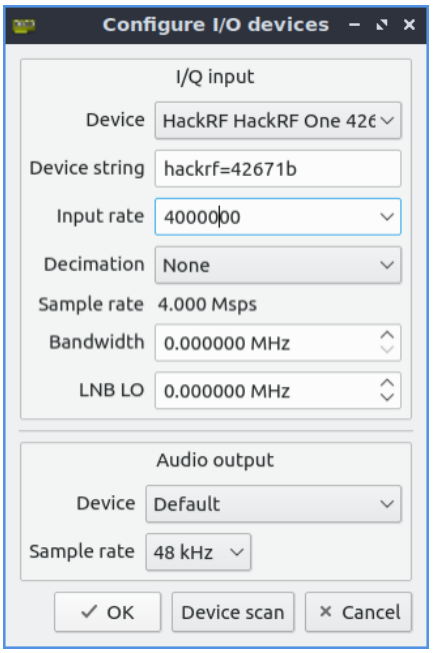


Figure 2 Configure HackRF One

Now that we've configured the I/O device, we need to set the correct frequency. Based on the FCC ID (2AEIM-1023049)[8] of the Tesla charge port remote, we know it operates at 315MHz. We don't want that to be our center frequency, though. We'll always get a "loud" signal at the hardware center frequency caused by DC voltage on the radio. To get past this, we'll technically listen to 500KHz (NOTE: 500KHz is arbitrary. Just pick something that offsets the signal a bit from the SDR's hardware center frequency) above or below our target 315MHz. A frequency slightly offset from 315MHz will move the "loud" signal away from the target, making it easier/possible to analyze.

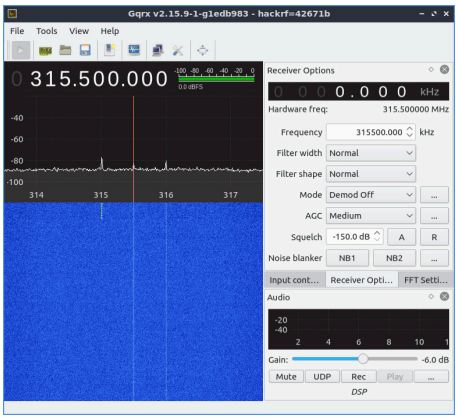


Figure 3 Seeing the Signal of Interest

Now, we should see something like Figure 3. Right around 315MHz, we can see a few yellow blips on the spectrogram; that's our signal. It's time to record it for further analysis. In *Gqrx*, we can use the *I/Q Recording Tool in Tools > I/Q recorder*. A recording is simple. Select the target directory to save the file, and press the *Rec* button to start and stop recording. You must ensure *Gqrx* actively listens for the recorder to capture data.

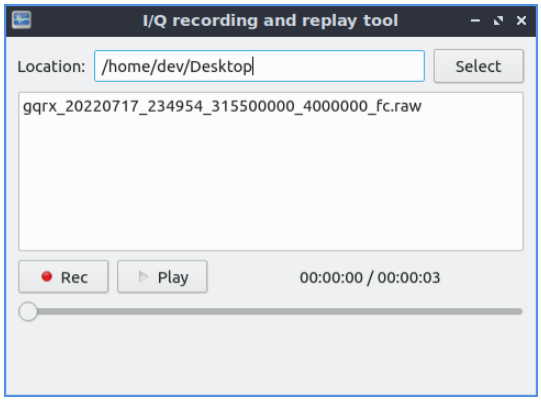


Figure 4 Saving a Raw Signal Capture

I'll take a quick moment here to remind us of a few small details to keep in mind. If you remember, we said the signal is saved as a complex number using 32-bit floating point numbers to represent each sample, so 16 bytes per sample. We also set *Gqrx* to capture 2 million samples per second. For every second of the signal captured, we need 32 million bytes of data. Thirty-two megabytes isn't too much of a burden by today's standards. However, increasing the capture time and sample rate could drastically increase the data storage requirements.

Analyze the Signal

We'll use *Inspectrum* to look at the signal we just captured. When you open the raw I/Q data captured earlier, you must scroll over until you find a stand-out signal. There may be a solid line across the spectrogram view, the DC voltage spike you get on some SDRs. If you didn't offset your hardware center frequency, your signal would show up above that solid line or below it depending on how you offset the signal.

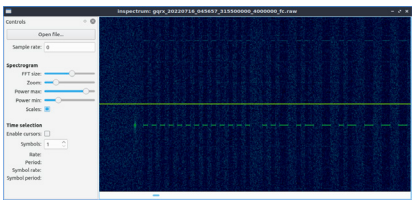


Figure 5 Open Captured Signal in Inspectrum

Once you find some darker-colored dashes in the plot, you have likely seen the signal of interest. Right-click on the spectrogram view and select *Add derived plot > Add* amplitude plot to begin analysis. Use your mouse to drag the shaded area on the spectrogram over the signal of interest. You may also need to change the height of the shaded area only to cover a small part of the signal. Lastly, you may need to adjust the *Power max* on the left *Controls* panel. As you change to these variables, the signal peaks rise over the dashed line on the amplitude plot, and the signal troughs stay below the dashed line. The image below shows how we want the amplitude plot to look.

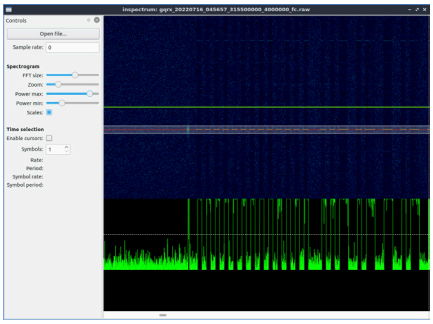


Figure 6 Add amplitude plot

Now that we have an amplitude plot with stronger signals above that dashed line and weak signals below the dashed line, we can add a *Threshold* plot which gives us a clean on/off signal. To add the Threshold plot, right-click on the amplitude plot and select *Add derived plot > Add threshold plot.*

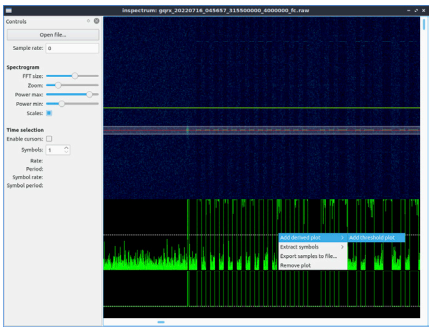


Figure 7 Add threshold plot from an amplitude plot

With the threshold plot, we can more easily discern the signal. At this point, we are making the transition from raw data, in the form of energy, to information, in the form of 0's and 1's. Check Enable cursors in the left Controls panel of *Inspectrum*. You'll see a big vertical semi-transparent bar appear, and we'll use this to align with the signal and count symbols.

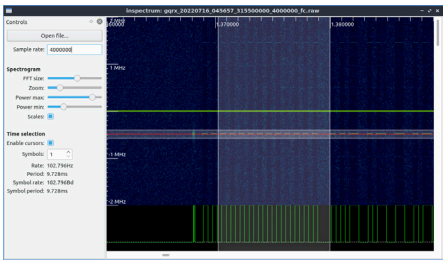


Figure 8 Add cursors to start understanding signal parameters

If you remember, back when capturing the signal, we specified the sample rate and how many samples per second the SDR would read and save. Now that we're looking into the characteristics of the signal, it's best to start representing everything as precisely as possible. On the left side of the *Controls* panel, change the *Sample rate* to match whatever you used to capture the signal. If using Gqrx, the sample rate could be to the default 8,000,000. For this write-up, the sample rate is 4,000,000 samples/second, but you could lower it down to 1Msps and still acquire enough of the signal to correctly understand the data.

We need to align the symbols to where we think the signal starts and increase the number of Symbols in the *Controls* panel. In this screen grab, just to get things lined up, I set the symbol count to 25.

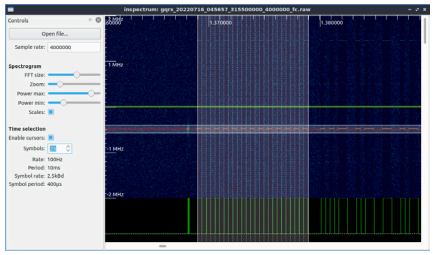


Figure 9 Aligning symbols with signal

With the symbol cursors displayed over the signal, we can add more symbols to cover up the entire signal. I like to add a few symbols at a time, ensure all symbols continue to line up on a rise/fall of the signal, and drag the right side of the overlay as needed to ensure everything lines up. After about 3,356 symbols, we'll cover the entire charger transmission.

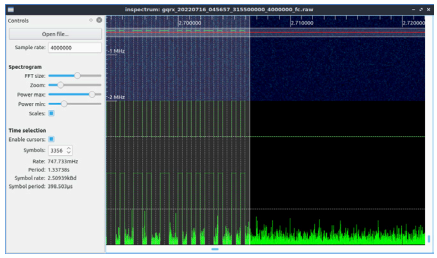


Figure 10 Extending symbols to the end of the captured signal; 3,356 symbols

When everything lines up, we can see that the charge port remote uses a signal with the following characteristics:

- Period: about 1.3s transmit time
- Symbol rate: about 2.5 kBd
- Symbol period. about 398 microseconds

The last step of this analysis is to try to make sense of all these on/off, high/low symbols. We now have complete information in the form of 0's and 1's, but we to turn the add another layer of abstraction to the information. This is the point where we may see a certain numbers of bits represent fields of a structured frame for example. *Inspectrum* helps us by automatically extracting the symbols giving us an output of 0s and 1s so we can do a higher-level signal analysis if needed. Right-click inside Inspectrum and select *Export samples to file... or Extract symbols > Copy* to clipboard to extract the symbols. The latter will give us a comma-separated list of 0s and 1s we can paste into notepad.

Looking at the bits in notepad, we can see what appears to be the start of a frame, some unique sequence of bits, then the actual data.

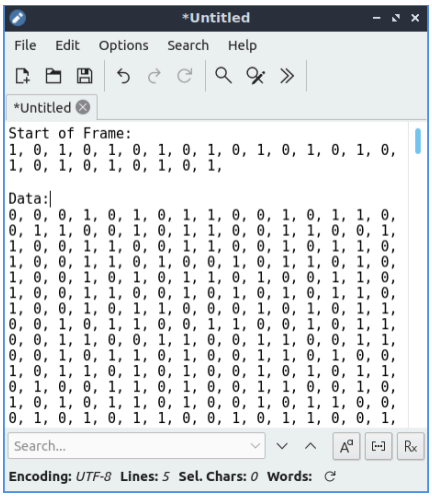


Figure 11 Copied bits from Inspectrum

When looking at how much data there was, we may wonder why the signal was so long and what kind of patterns it holds. We can search for text and see what appears to be a clear repeating start of the frame, and sure enough, it repeats ten times.

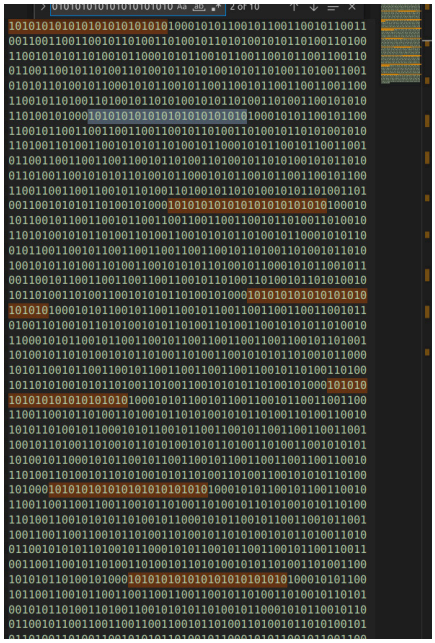


Figure 12 Using VSCode to visually see repeating text

If we search the entire chunk before the preamble, we see the bits that repeat are 336 bits (42 bytes) and repeat ten times for a total of 420 bytes. Here we can use context from Elon Musk's Tweets to help us know we're probably on the right path to understanding the signal.

Figure 13 Typical Tweet

The 42-byte charge port signal is:

0xAAAAAA8ACB32CCCCCB4D2D4AD34CAB4B1596659999969A5A95A69956962B2CCB33332D34B52B4D32AD28

The entire 420-byte signal is:

*0xAAAAAA8ACB32CCCCCB4D2D4AD34CAB4B1596659999969A5A95A69956962B2CCB33332D34B52B4D32AD28AAAAAA
8ACB32CCCCCB4D2D4AD34CAB4B1596659999969A5A95A69956962B2CCB33332D34B52B4D32AD28AAAAAA8ACB32CC
CCCB4D2D4AD34CAB4B1596659999969A5A95A69956962B2CCB33332D34B52B4D32AD28AAAAAA8ACB32CCCCCB4D2
D4AD34CAB4B1596659999969A5A95A69956962B2CCB33332D34B52B4D32AD28AAAAAA8ACB32CCCCCB4D2D4AD34CA
B4B1596659999969A5A95A69956962B2CCB33332D34B52B4D32AD28AAAAAA8ACB32CCCCCB4D2D4AD34CAB4B159665
9999969A5A95A69956962B2CCB33332D34B52B4D32AD28AAAAAA8ACB32CCCCCB4D2D4AD34CAB4B1596659999969A5
A95A69956962B2CCB33332D34B52B4D32AD28AAAAAA8ACB32CCCCCB4D2D4AD34CAB4B1596659999969A5A95A69956
962B2CCB33332D34B52B4D32AD28AAAAAA8ACB32CCCCCB4D2D4AD34CAB4B1596659999969A5A95A69956962B2CCB
33332D34B52B4D32AD28AAAAAA8ACB32CCCCCB4D2D4AD34CAB4B1596659999969A5A95A69956962B2CCB33332D34B
52B4D32AD28*

Synthesize the Signal

We already have the raw I/Q data captured from the remote, and we could simply replay that with the SDR to achieve the same effect as using the remote. Instead of merely replaying the raw signal, we will synthesize the signal from the earlier 0s and 1s we extracted in *Inspectrum*.

You can find a downloadable GNU Radio flowgraph on GitHub[9]. You should be able to open the flowgraph up in GNU Radio and run it with some modifications. The Soapy Sink (the output) block is configured for a HackRF One and would need changing if using a different SDR. Also, you may not have a Soapy sink as it only comes with GNU Radio after a specific version.

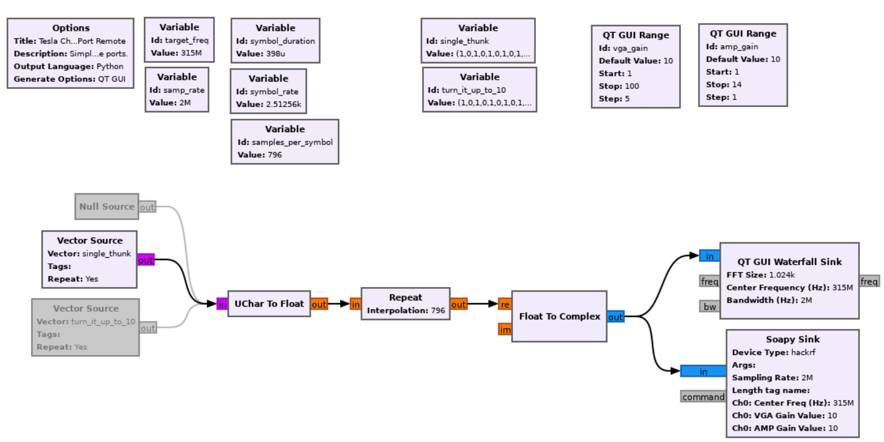


Figure 14 GNU Radio flowgraph for synthesizing charge port remote signal

Conclusion

Throughout the journey of this write-up, we (hopefully) invoked a sense of wonder and curiosity of our surrounding. We explored a way to solve open-ended and complex tasks by decomposing a problem into smaller, more manageable problems. Then we walked through the deliberate process of capturing, analyzing, and synthesizing an unknown signal of interest. There is a plethora of ways to solve this same problem outlined here, and I'm sure many ways more intuitive than what I've described. The point I hope to leave you with is that the barrier to entry for reverse engineering may not be as big as you think. With a little curiosity and a little time, I hope you can start, or continue, on your journey to really get reverse engineering.

References:

[1]*https://shop.tesla.com/category/charging*

[2]*https://greatscottgadgets.com/hackrf/one/*

[3]IQ data is a way to represent a wave. For this reading, you can think of the IQ data as a complex number consisting of two 32-bit floating point numbers; one float for the I and the second float for the Q.

[4]*https://gqrx.dk/*

[5]*https://github.com/miek/inspectrum*

[6]*https://www.gnuradio.org/*

[7]*https://bbs.archlinux.org/viewtopic.php?id=265800*

[8]*https://fccid.io/2AEIM-1023049*

[9]*https://github.com/akrutsinger/tesla-charge-port-signal/blob/main/src/tesla_charge_port_remote_ook_synthesizer.grc*

Let's Play the Quantum Coin Game

By CPT Hamilton Bonds, 2d IO Battalion, 1st IO Command (L)

DO YOU WANT TO GET STARTED DEVELOPING QUANTUM CIRCUITS IN PYTHON? IBM's Quantum Labs makes creating and visualizing quantum circuits straightforward, and you can do more than what is written in this article if you take the time to go through the Qiskit tutorials. To be clear, there are other quantum simulators, but they *simulate*. If you want the real quantum experience, please use IBM's Qiskit software development kit (SDK) and call one of their quantum computer backends. Load up your virtual machine (VM) while you read if you want to play the Quantum Coin Game.

Two problems in the 2021 DEFCON Qualification Capture-the-Flag (CTF) Competition were based on this game, and many more CTFs, including the Google CTF, have included problems on quantum encryption and quantum computing. The case for quantum computing is solving mathematical problems that would have otherwise been impossible. However, an unfortunate side-effect is that quantum computing threatens modern encryption to such an extent that the National Institute of Standards and Technology (NIST) runs an initiative to develop quantum-resistant algorithms. Current efforts in quantum-resistant cryptography (QRC) will inevitably fail as quantum computing progresses. Therefore, if the industry of cybersecurity does not immediately embrace quantum cryptography, classical encryption implementations will swiftly become obsolete and render vulnerable the systems they formerly protected.

Quantum Welcomes All. Whether you are comfortable with working with so-called black box systems you do not completely understand or you are yearning to see the applications of what you already know about quantum information systems, then with a little time and motivation, you can learn the intricacies of quantum circuits. For this article, we will be exploring a simplified demonstration of quantum cryptography principles called the "Quantum Coin Game", during which we will lose to a quantum computer roughly 97% of the time. Remember, the point is not whether you win or lose, but if you have fun!

Concept of the Game. The game is played in as many iterations as you would like, but the quantum computer always gets the first and last turn. If the coin lands on *heads*, then the quantum computer wins, and if it lands on *tails*, you win. Neither you nor the quantum computer can be aware of the other's actions, which is a cryptographic concept called "strong coin flipping." Each turn occurs as follows:

1. The quantum computer "places" the coin on *heads*, and then "flips" the coin
2. You, the human, can then choose to flip the coin again or do nothing
3. The quantum computer "plays" and then the final results are shared

Installing Qiskit on Python. No matter your skill level, it is useful to build quantum circuits with Qiskit inside of a Jupyter Notebook because of Jupyter's ability to display visualizations, and Qiskit has a lot of graphical modules. With or without Jupyter, anyone can access IBM's quantum simulation circuits, and with an IBM account, anyone can utilize IBM's real-life quantum computers - both for free! If you're less interested in the *how* and more interested in the so what, feel free to skip to the section **Practical Applications in Cybersecurity.**

When your VM loads, load a terminal and run the following command according to your specific requirements:

If NOT using Jupyter Notebook:

```
pip install qiskit
```

If using Jupyter Notebook:

```
conda create -n name_of_my_env python=3
conda activate name_of_my_env
pip install 'qiskit[visualization]'
```

For the rest of the article, we will only be presenting code that runs in Jupyter Notebook.

Running the Game. The game consists of 4 components - (1) a rudimentary human-machine interface (HMI), (2) the turn decision subroutine, (3) the classical system, and (4) the quantum system. The code below annotates the location of all 4 components.

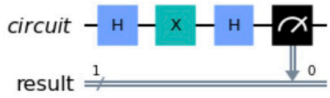
```
from qiskit import QuantumCircuit, Aer, IBMQ, QuantumRegister,
ClassicalRegister, execute
from qiskit.tools.jupyter import *
from qiskit.visualization import *
import qiskit.tools.jupyter
import ipywidgets as widgets

# Component 1 - HMI
button_p = widgets.Button(
    description='Play'
    )
gate_p = widgets.Dropdown(
    options=[('Identity', 'i'), ('Bit Flip', 'x')],
    description='Choice: ',
    disabled=False,
    )
out_p = widgets.Output()
def on_button_clicked(b):
    with out_p:

        # Component 4 - the quantum system
        circuit_p = QuantumRegister(1, 'circuit')
        # Component 3 - the classical system
        measure_p = ClassicalRegister(1, 'result')

        qc_p = QuantumCircuit(circuit_p, measure_p)

        # Turn 1
        qc_p.h(circuit_p[0])

        # Turn 2
        if gate_p.value == 'i':
            qc_p.i(circuit_p[0])
        if gate_p.value == 'x':
            qc_p.x(circuit_p[0])

        # Turn 3
        qc_p.h(circuit_p[0])

        # Measure
        qc_p.measure(circuit_p, measure_p)

        # Visualize the circuit
        display(qc_p.draw(output="mpl"))

        # This is a quantum simulator, and it's pretty good
```

This code runs the game, so give it a try by running your notebook. You will be presented with a choice:

Figure 1: Component 1 - the HMI presents the human with a choice

If you choose “Identity”, your turn is “do nothing” (top of Figure 2), but if you want to select “Bit Flip”, you will flip the coin.


```
You Lose to Quantum. Quantum Computer Wins
```




```
You Lose to Quantum. Quantum Computer Wins
```

Figure 2: A visualization of the results of Identity (top) and Bit Flip (bottom)

The quantum computer’s first and last turns before measurement are represented by the blue box with an “H”, which symbolizes the application of a Hadamard operator. Your turn is either the “I” or “X” box between the Hadamard operators. The last black box with a meter represents the final measurement.

Quantum Encryption Explained. Losing 97% of the time is tough to handle, but let’s understand why we lost and how it applies to quantum encryption. For the sake of brevity, we will not expound upon every subtopic of quantum encryption.

Starting out, the quantum computer initializes the coin on heads, meaning the qubit is set to state $|0\rangle$, represented as

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

with *tails* as

$$|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

The Identity Matrix $\boldsymbol{I}$ is represented as

$$\boldsymbol{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The Bit Flip $\boldsymbol{X}$ operator:

$$\boldsymbol{X} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

The Hadamard $\boldsymbol{H}$ operator:

$$\boldsymbol{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

We can see how each operation works in this game from the perspective of linear algebra:

In the Initialize step, the quantum computer instantiates the qubit as a “0”, $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

In Turn 1, the quantum computer puts the qubit into uniform superposition (keep reading for more information on superposition),

$$\mathrm{H}|0\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1*1+1*0 \\ 1*1-1*0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = |+\rangle$$

which consequently results in an equal probability of measuring either a 0 or a 1, hence why we can refer to the system as we refer to a coin. Here is how we calculate the probability:

$$p(0) = |\alpha|^2,\ p(1) = |\beta|^2$$

where

$$|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

and

$$|\alpha|^2 + |\beta|^2 = 1$$

Given that

$$|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$$

we can calculate the probability of getting either a 0 or a 1 as

$$p(0) = \left| \frac{1}{\sqrt{2}} \right|^2 = \frac{1}{2},\ p(1) = \left| \frac{1}{\sqrt{2}} \right|^2 = \frac{1}{2}$$

In Turn 2, the human has a choice to do nothing to (take the identity of) the qubit,

$$I|+\rangle = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1*1+0*1 \\ 0*1+1*1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = |+\rangle$$

or the human can choose to “flip the coin” using the bit flip operator

$$X|+\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 0*1+1*1 \\ 1*1+0*1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = |+\rangle$$

Whatever the human’s choice, the qubit is still returned to the same state. A classical choice is irrelevant to a superposed state because either result is equally valid.

In Turn 3, the quantum computer returns the coin back to its base state,

$$H|+\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \left(\frac{1}{\sqrt{2}} \right)^2 \begin{bmatrix} 1*1+1*1 \\ 1*1-1*1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 2 \\ 0 \end{bmatrix} = |0\rangle$$

Finally, we Measure the result of our operation and get

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Quantum wins! Now, this may seem like cheating at first, but consider what the quantum computer does in its first turn. The Hadamard gate transforms the initialized qubit from essentially a binary 0 to where it represents two states: both a 0 and a 1 simultaneously. This is what is referred to as superposition, a behavior which is fundamental to quantum mechanics, quantum information systems, and quantum encryption.

Demonstrate the power of quantum superposition and comment out two lines in your code to look like this:

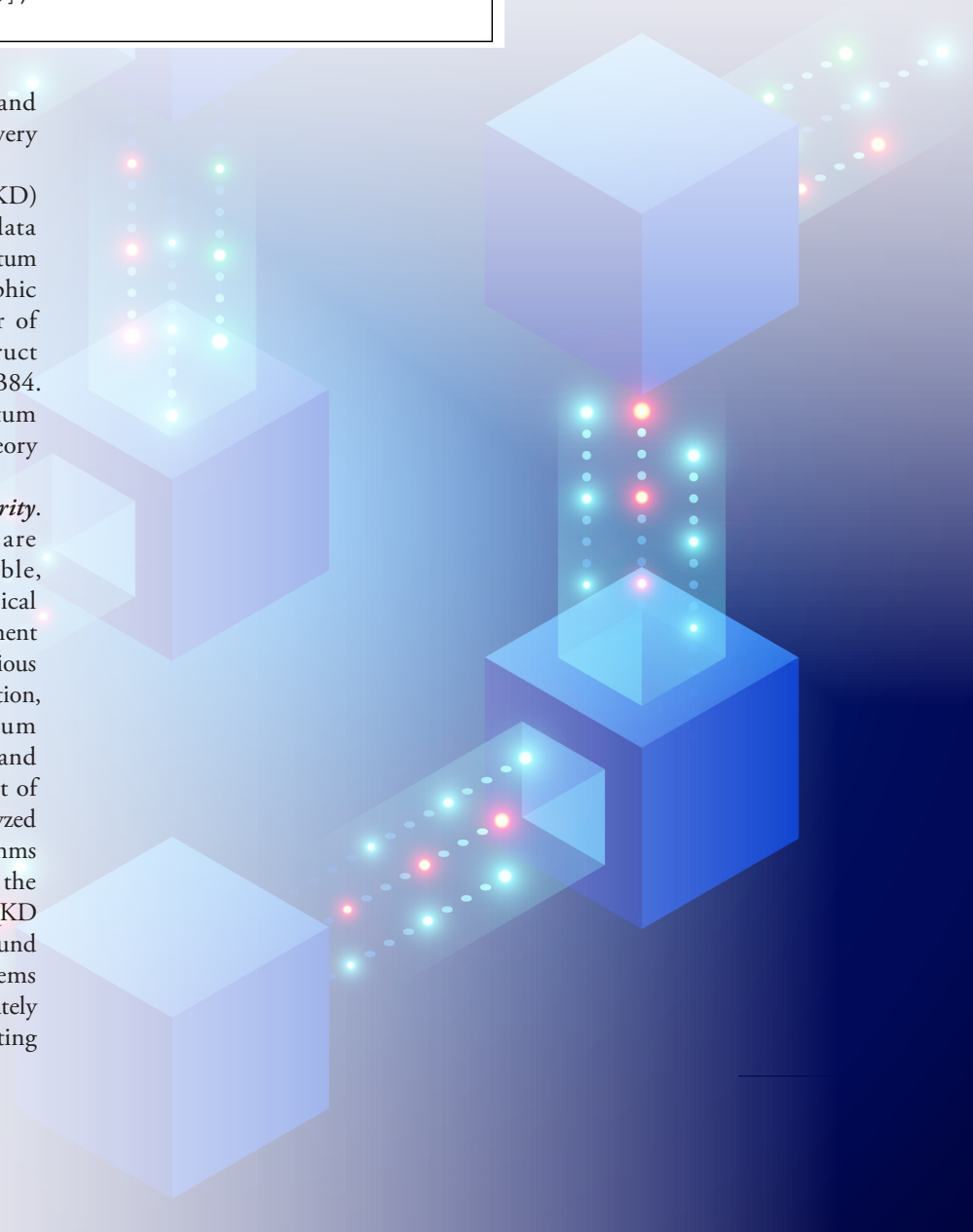
```
...
        # Turn 1
        #qc_p.h(circuit_p[0])
...
        # Turn 3
        #qc_p.h(circuit_p[0])
...
```

Now run the simulation again and select "Bit Flip". You should win every time now!

Quantum key distribution (QKD) involves protocols that enable data encryption by quantum means. Quantum coin flipping is a nascent cryptographic primitive that harnesses the power of quantum superposition to construct more complex protocols, such as BB84. Qiskit has a walkthrough for quantum key distribution, so dive into the theory and the code to understand more.

Practical Applications in Cybersecurity. Quantum encryption devices are commercially available, affordable, effective, and compatible with classical internet systems. Such devices implement algorithms that are theoretically impervious to tampering and unauthorized decryption, harnessing properties of quantum mechanics to detect eavesdroppers and renegotiate private keys. The advent of "quantum-resistant cryptography" catalyzed the development of encryption algorithms that are intended to resist or negate the power of quantum computing, but QKD is demonstrably a more technically sound solution. In order to ensure all systems transmitting sensitive data are adequately prepared for future encryption-defeating capabilities and eavesdropping threats, organizations must include quantum encryption devices in their networks. Additionally, implementing QKD implies that cyberspace defenders must train to understand and operate quantum encryption devices. This article should serve as the first step in the journey to understand quantum encryption.

References. All concepts presented in this article can be found in various sections of the Qiskit documentation at https://qiskit.org/documentation/ and in your nearest linear algebra textbook. ■



Protect Yourself From Gamer Input

By Capt. Robert Guiler, USAF 341 COS

VIDEO GAMES HAVE BEEN A PERVASIVE PART OF OUR CULTURE FOR DECADES. Their software runs on everything from extravagant desktops to small, custom, handheld devices like Tamagotchis. The unique input options for video games can make it challenging to identify vulnerabilities in their software. While many modern P.C. games and their servers can be targeted with traditional exploits, many retro or simpler games can only be accessed using their intended gaming mechanics. Whether it is a slot machine, Super Nintendo, or treadmill, the software can still be vulnerable if you know where to look.

There is a uniquely motivated group of penetration testers, called speedrunners, that specifically target video games. Their aim is to identify glitches, bugs, and vulnerabilities in order to beat games as quickly as possible. This group has managed to find multiple local code execution vulnerabilities using nothing more than joysticks, button presses, and skill. In one case, *Super Mario World*, a runner named SethBling wrote a malicious payload by having Mario jump and toss koopa shells to posture memory into a specifically crafted exploit. In another, *Ocarina of Time*, runners exploit the main character's position, loading triggers, and joystick in order to gain full access to the Nintendo64's memory. Using these methods, speedrunners have done everything from writing custom graphics to corrupting the system to exploit other games.

Protect Yourself From Gamer Input at AvengerCon VI in 2021 focused on three specific cases where glitches could lead directly to full memory access. These were the two games mentioned above, as well as the original Pokemon games for the Gameboy Advanced. In each case, small glitches allow the attacker to establish variables so that a method improperly grants partial access to memory. Typically, this allows a small payload to run that has been created using sprite positioning, player actions, and button inputs. To perform more complex attacks, hackers use the small payload to allow player input from less restricted inputs. For *Ocarina of Time* that means using the small payload to remove restrictions on, and then execute, the name variable of the player character so that they can type out the payload. For *Pokemon Yellow*, this means pointing the smaller payload to allow the menu or player controls to write directly to memory.

Protect Yourself From Gamer Input II at AvengerCon VII in 2022 focused on ways you can discover these types of vulnerabilities. Most modern games have ways of being patched, but they are not immune to major glitches. While it is rare, major glitches like the *Pokemon MissingNo or World of Warcraft Corrupted Blood* can happen. In most cases, multiple smaller bugs are chained together to build a larger exploit. One may establish a variable outside of expected parameters, another skips the checks a method performs to validate, and a final bug prevents the game from crashing. Keep in mind, most of the examples do all of this using jumps, rolls, grenades, intentional deaths, and other typical user input rather than overflowing a buffer.

While most of these exploits are only useful to do fun things in older games, there are numerous examples where there have been real world consequences. Duplicating or glitching items or gold in massively multiplayer online games can be worth millions of dollars, as shown by numerous DEFCON talks linked in PYFGI2. Similar glitches can be used to "troll" or "grief" other players by ruining their games, rankings, or in-game items. In some cases, glitches ruined in-game economies so much that full server resets were required in order to get them back into balance. If you are interested in learning more about video game glitches, check out speedrun.com or /speedrun on reddit. ■

Hacking DevOps

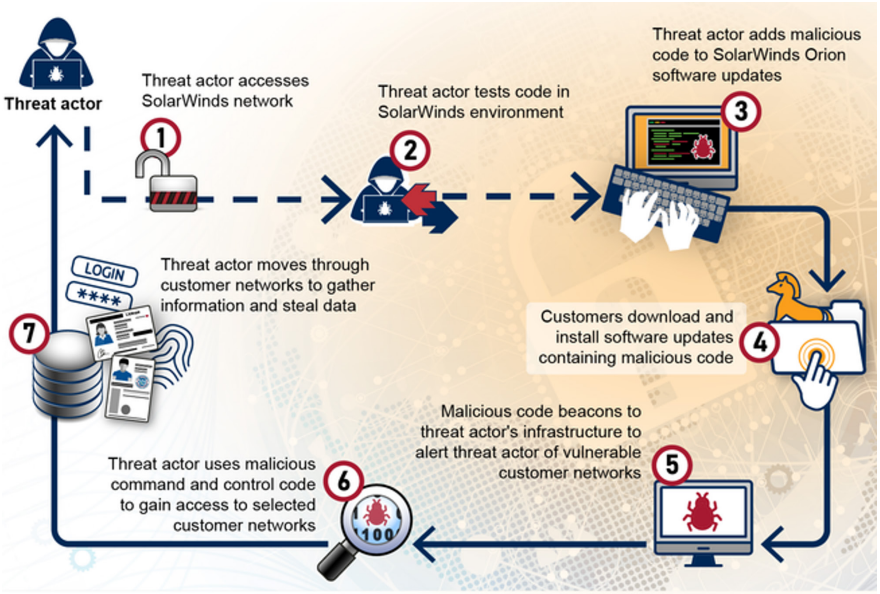
By Phillip Marlow, Lead Systems Engineer at MITRE

MALICIOUS SOFTWARE UPDATES, like the one embedded in SolarWinds Orion just a couple of years ago, can have major impact to organizations. The cleanup from the SolarWinds episode cost millions of dollars, and it has been far from the only incident of its kind. Other incidents include the malicious update of software packages in open-source repositories which have impacted millions of downstream users. These incidents are taking advantage of weaknesses in the software supply chain and the DevOps practices which enable it.

DevOps is an incredibly popular way of rapidly delivering and deploying software. Its growth within the Department of Defense is eclipsed only by its adoption in industry. But it is not just the United States and its allies which are taking advantage of this technique; other nations and even criminal enterprises are taking advantage of DevOps to increase their operational resilience. Given its broad use within the Department and by our adversaries, it is critical that both our defensive and offensive cyber operators understand the challenges and opportunities DevOps presents to cybersecurity.

Offensive operators and cyber assessors can take advantage of the need for DevOps tools to have broad permissions within an environment to increase their access once they have achieved a foothold. Many opportunities exist to move laterally within an environment or to expand into other test or production environments once development tooling has been compromised.

Imagine this scenario: using a simple attack, perhaps via phishing, to get access to a developer's laptop. From there you are able to use a publicly known bug in GitLab, a popular Continuous Integration/Continuous Delivery (CI/CD) system, to subvert the software build process at your target. From there you begin to exfiltrate details about their test and production environments which allow you to craft a stealthy malicious update which no longer needs to conform to the target's security testing practices – because you'll just report compliance regardless of the test results. Finally, you use the target's own infrastructure-as-code solution to ensure that you retain uninterrupted access, even through software updates and the use of new servers.



Source: GAO analysis of documentation from publicly released private industry and federal agency reports; images: kras99/stock.adobe.com, anna_leni/stock.adobe.com. | GAO-22-104746

Passwords, test results, build artifacts, and more are within the reach for someone who has compromised a DevOps pipeline. This gives these operators not only greater access, but also better persistence mechanisms to maintain that access. All this means that DevOps tools are a greater target than previous development resources have been.

Defenders need a broad understanding of the software development process and technical environment in order to effectively protect against these kinds of attacks. They can use the many security features that are provided within modern software development tools, provided they know how to use them properly. Many of the same tools that were exploited in the scenario above can also be configured to increase security, limit the blast radius of any compromises, and increase the opportunities to detect both inadvertent and malicious security events before they become major incidents.

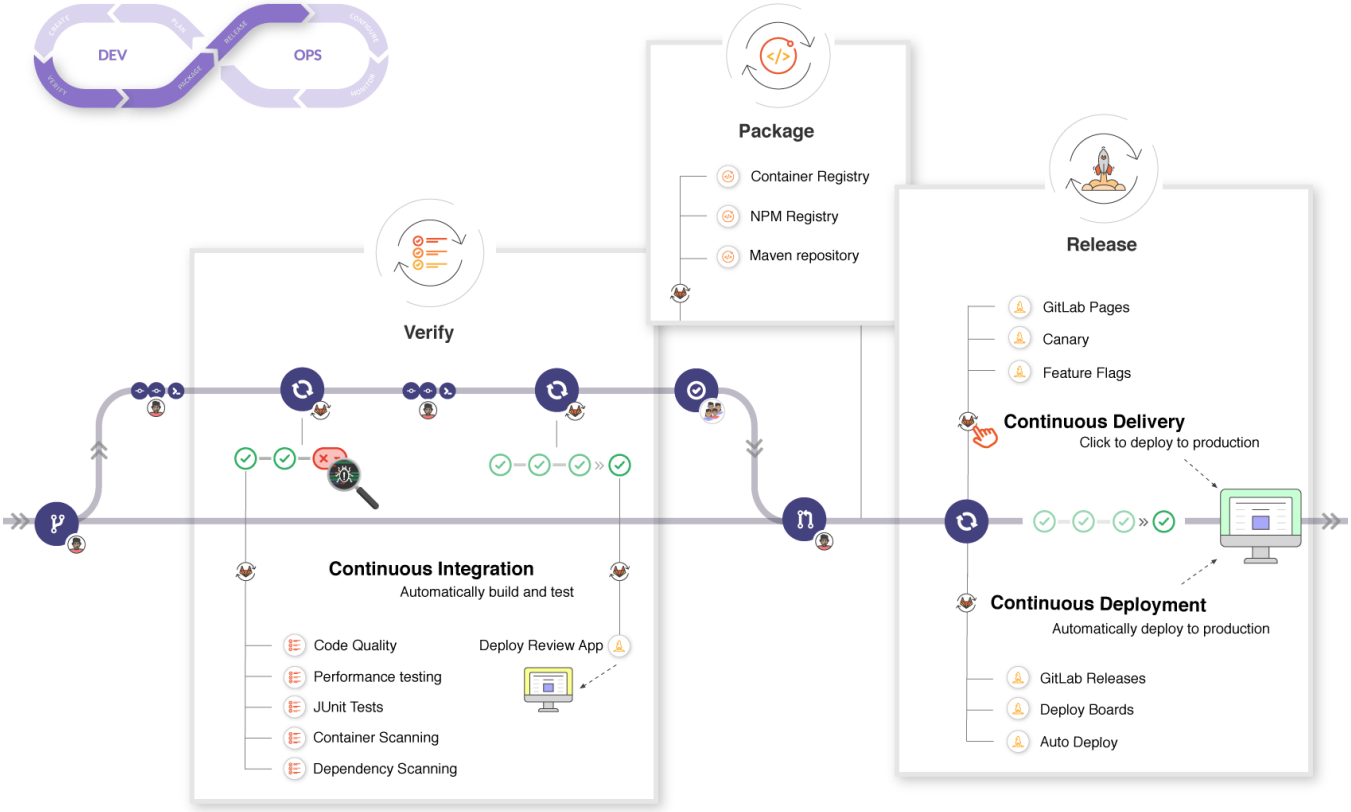
Instead of the scenario above, imagine this instead: despite a compromise of the GitLab, attackers were only able to exfiltrate limited information about the environment because the CI/CD pipeline was limited in scope. Attackers lacked access to make changes to test plans or results, making it much more difficult remain undetected or to alter the software to be malicious. They could not expand into other tools because passwords were properly protected within the system by an encrypted vault. Furthermore, the attacker's attempts to maintain their access were quickly detected and automatically removed by properly secured infrastructure-as-code solutions, which also patched GitLab more quickly than

a traditional patch cycle could enable, meaning that attackers were unable to repeat their attack.

By recognizing the need for security throughout the development process, this time defenders were able to use the same tools that were of concern before to severely limit the impact attackers were able to have. But getting to this point is complex and difficult work, requiring deep and continuing coordination between cybersecurity, development, operations, and the missions they support. Learning how to do this requires deliberate attention and effort and will have enormous impacts to security whether it is done well or poorly.

Approved for Public Release; Distribution Unlimited. Public Release Case Number 23-0678

The author's affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions, or viewpoints expressed by the author.



The problem with Chekhov's Gun

By Mike Reid, PMP, Director of Project Development, Xorre

WE ARE GOING TO LOOK AT A LOGIC PROBLEM that impacts offensive security and how we can use Artificial Intelligence (AI) and Machine Learning (ML) to solve it. We are going to look at the implications of "Chekhov's Gun," and how its lessons can be applied to digital security.

What is Chekhov's Gun?

According to ChatGPT: Chekhov's Gun can be summarized as a storytelling principle that states that any element in a story that is introduced but not used later on, is a wasted element. It is named after Russian playwright Anton Chekhov, who wrote that "if in the first act you have hung a pistol on the wall, then in the following one it should be fired. Otherwise don't put it there."

Put another way, any element introduced is intended to be used, otherwise why expend the effort of introducing it? It's now used as a logic problem, where "Chekhov's _____" deals with the logical implications of an item's existence.

The problem with Intent

When applied to security and encryption, an adversary can make assumptions about one's intentions and outcomes based on their actions, which can be more indicative than anything physically found.

Consider the scenario of an embezzler being searched by the police. To protect themselves, they purchase a top-of-the-line hidden safe and move all incriminating files into it. When the police search the embezzler's place and find nothing, the safe has done its job.

However, during the search a receipt is found for the purchase of the safe. The discovery of a record of the safe's purchase and installation raises suspicions and opens the embezzler up to risk in multiple ways.

Firstly, the police may assume that the embezzler's intent was to hide the existence of the safe and its contents, as they would not have purchased an expensive hidden safe if they had nothing to conceal as a reinforced fireproof safe would be larger and cheaper.

Secondly, they may assume that the safe contains valuable and important items. You logically wouldn't buy a safe and then not put these items into it.

The problem with assumptions

These assumptions can lead to coercion, the implication of guilt, and severe punishment or execution in some cases. Police have a specific data point (the location of the safe) they are able to probe for, and then immediately verify. They may not be able to force you to answer, but they know what question to ask.

It can also create the implication of guilt if you refuse to reveal the location. Most people, privacy issues aside, would likely cooperate given the possible alternatives. Some adversaries may assume guilt based on refusal to comply.

A specific digital example of this problem is the use of TOR, an incredibly useful tool for secure communication. However, the use of TOR alone is seen as a guilty act in many parts of the world, and multiple governments may assume one is a drug dealer and execute them based on this assumption alone.

Fortunately, just as there are solutions to Chekhov's Gun in storytelling, such as the use of Red Herrings, we can apply these same approaches to digital security and hidden items. These are story elements that are included to be deliberately misleading.

Solving "Chekhov's Hidden Container" problem

We are going to carry on with the scenario above, but with Red Herrings created via machine learning. In the case of the embezzler, this could include creating multiple fake data sets and clues to make it impossible for the police to make assumptions about any individual activities.

We are going to look at this from the perspective of "General Stickyfingers," who is a military commander as well as embezzler. To solve the problem of intent, we are going to build our environment around these hidden compartments.

We get furniture exclusively from "Hidden Containers 'R Us" and stack them everywhere – and even put some inside of other hiding places. We will remove the ability to draw any inferences regarding the existence of the hidden compartments, and your adversary will never be completely sure they found them all.

However, even with these precautions, we are still vulnerable to a search. We can make it much more difficult, but we are still vulnerable to brute force efforts.

Apply Machine Learning

To mitigate this risk, we could use AI and ML to generate fake financial documents, which could be unique to individually generated financial crimes or implicate different individuals for the same crime. This would make it extremely difficult for the police to identify which documents are genuine without outside references.

In addition to these measures, we could also use our position as a commander to further muddy the waters. We could mandate that all furniture used in their buildings also include hidden compartments, and encourage the occupants to use some of these hiding places without providing any context on which ones are relevant.

By implementing these strategies, we could protect ourselves from searches and interrogations, and make it difficult for our opponents to make assumptions about their intent. It would also be harder to know which hiding places are relevant, decoys or are known by whom, providing an additional level of obfuscation to their secrets.

The final decision is if we want the environment for every individual to be identified or completely unique. If we make each environment identical, we

completely remove the ability to draw any conclusions from the existence of any containers. If we make each one unique, we increase the complexity of the investigation and make it much more difficult to draw comparisons.

Building an Obfuscation Engine

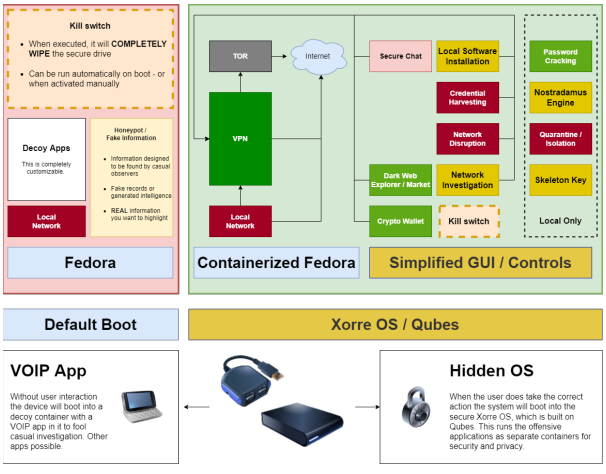
Now we will take these same principles and apply them to digital security. Our goal will be to remove all assumptions based on intent, and apply AI and ML to protect our information.

Protecting Civilian allies

The first and most important use of this technology is to protect our civilian allies. In the current global conflicts, we see an incredible willingness from these populations to support the cyber conflict.

However, these individuals often lack the technical skills and knowledge to effectively defend themselves online, and there are significant personal risks for those who are caught participating in cyber activities. This puts an effective limit to how effectively these volunteers can participate.

To address these challenges, it is important to develop tools and protections that can be easily used by civilians. By utilizing the concept of "hidden compartments" within these tools, we can provide a layer of security and deception that can confuse and mislead adversaries.



One example of this approach is the use of containers within an open-source Linux operating system. These tools can be configured to run automatically, without the need for user input.

By providing allies with the knowledge on how to access the specific containers, they can utilize the tools while keeping them hidden within a sea of fakes. Additionally, fake information troves can be intentionally left in plain sight to further mislead adversaries.

The key to this approach is the choice of how each tool is constructed. If each tool uses an identical structure, with the same configuration of hidden compartments, it becomes much more difficult for adversaries to discover the relevant tools.

Furthermore, by flooding a population with these tools, it becomes impossible to determine which small subset of the population is actively using them. This makes it much more difficult for adversaries to target specific individuals or groups.

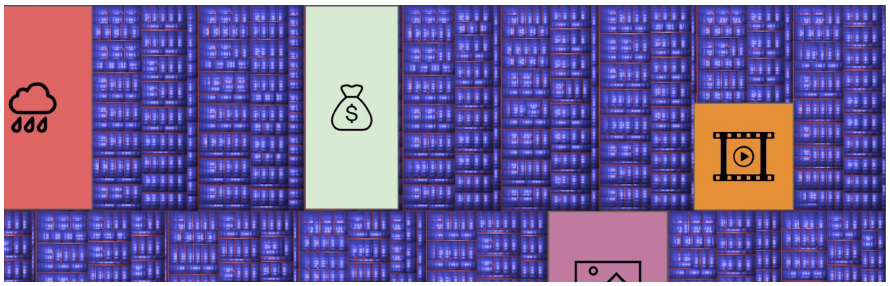
The best part of this approach is that you don't have to deploy the tools to be successful. The fact they exist and could be compromising vital systems might be enough to force a reaction. Getting an opponent to distrust their systems could be as effective as destroying them.

Obfuscation and Encryption Engine

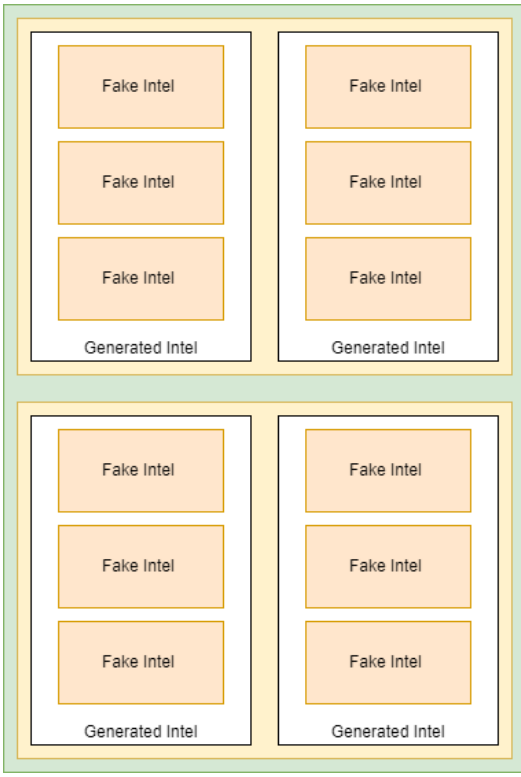
In order to ensure the security of sensitive files, it is important to implement a robust encryption and obfuscation system. In a large-scale cyber conflict, thousands of individuals may have access to the data, making it crucial to have a system that can withstand intense scrutiny.

One solution is to use nested containers with multiple layers of encryption. Files are initially encrypted and placed within a container, which is then itself encrypted. This process is repeated to the desired level of depth. This method is highly secure, and the use of multiple encryption keys can provide an added layer of protection.

The image below shows a visualization of one container (so 1/20th of the total size), and shows the legitimate data caches surrounding the generated files:



However, this method also raises concerns about intent, as it can make one's intentions obvious in certain situations. To address this, we can use machine learning to create fake caches of data that appear relevant but are not.

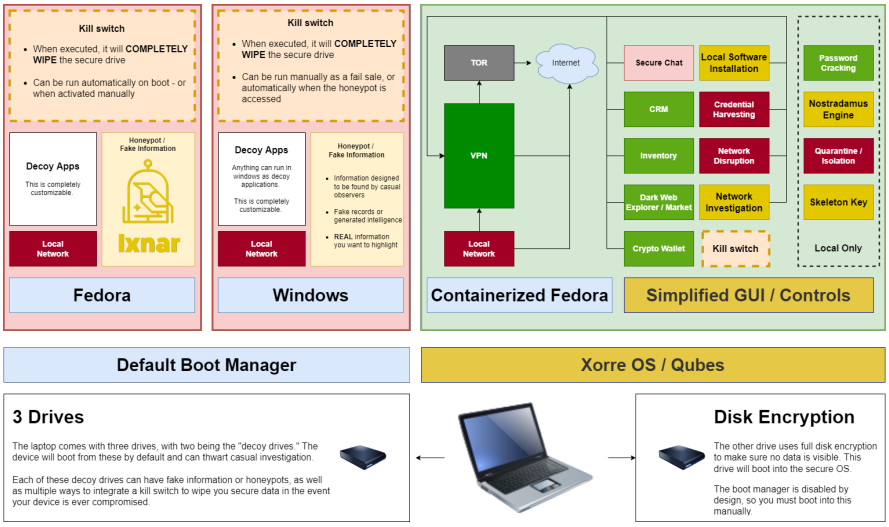


These fake caches can be created in a virtually limitless supply, and can be used to poison the well with false information, making it impossible for adversaries to make any assumptions about the true content of the data. Additionally, these fake caches can be used as honeypots to pass fake intelligence to your adversary.

Secure Hardware / Laptops

The final common use case for security and encryption is to secure a laptop or other hardware device. This scenario, while simple in concept, involves applying the principles discussed earlier on a localized level. To build on our initial scenarios, this would be the most effective way to utilize the hidden compartments once they are constructed.

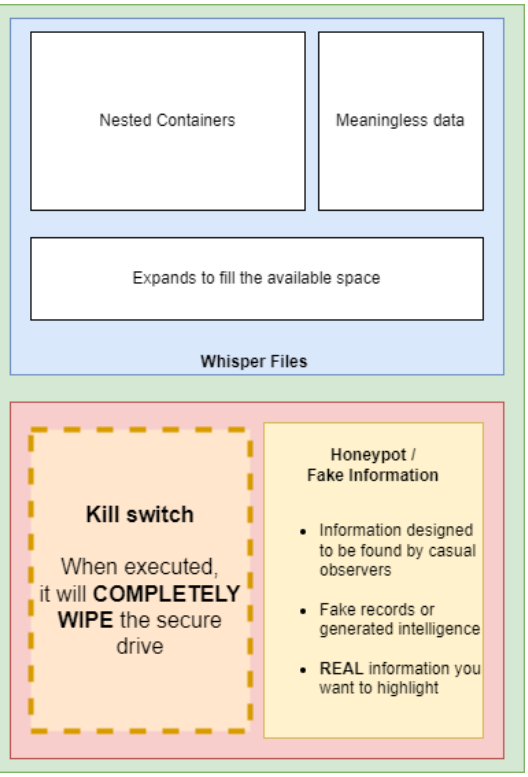
To secure a laptop, we begin by using multiple hard drives and a "decoy boot" that runs automatically and is designed to deter casual investigations. This decoy boot is a fully functioning operating system that is separate from the main, secure operating system.



The main operating system and data are stored on multiple other drives, which are protected using the secure containers previously discussed.

In addition to these containers, tripwires are added within the system to detect and respond to unauthorized access. These tripwires include fake containers and honeypots that are designed to catch the attention of an adversary. When accessed, they trigger a complete wiping of the data on the device, providing a failsafe mechanism to protect against unintended investigations.

Additionally, there is a manual destruction mechanism that the operator can initiate covertly to avoid discovery.



TLDR: *Your intent can tell more than your data*

The main takeaway should be that your adversaries can use your intent to make dangerous assumptions that might be worse than actual exposure. This is not a new problem, but we have new technology to deal with it. By using AI and ML, you can hide your intent on a scale that was previously impossible. Think about what your adversary could determine about you and use these techniques to make it work against them. ■

Connecting and Supporting Military Cyber Professionals

By Christine Billingsley, Chief Operating Officer, Military Cyber Professionals Association, christine@milcyber.org

IT WAS GREAT TO BE BACK AT AVENGERCON to share awareness about the opportunities the Military Cyber Professionals Association (MCPA) offers the Praetorians and the broader community of military cyber professionals across the joint force. For those unaware of the MCPA, please allow me to share a few highlights below. If you want to learn more, check out our website: *public.milcyber.org*.

The MCPA has the mission of developing American military cyber professionals and investing in our nation's future through science, technology, engineering, and mathematics (STEM) education. It's a national 501(c)(3) educational nonprofit and public charity, and serves the functions of a regimental association for the American military cyber community. We do this through a variety of operationally relevant programs like events (such as our national convention *HammerCon or the National Service Panel at DEF CON)*, publications (like CYBER magazine or *Military Cyber Affairs* journal), and recognition medal *(the Order of Thor)*.

While there is no shortage of organizations that play a role in this community, we are the only ones established by military cyber professionals specifically for military cyber professionals. We are not here to sell you things or take your money. Just the opposite, actually, as we waive membership fees for US military and government personnel, and honorably discharged veterans. Attendance at our amazing national convention (see HammerCon.org for details) is in the vicinity of Fort Meade, awards CPE/CEUs, and is completely free for US military personnel (even food) so you know we are here for you!

Since the establishment of the MCPA a decade ago, many members of the 780th have taken advantage of what the MCPA has to offer and we encourage you to lean into it. The Order of Thor is a flexible tool for leaders at all echelons to recognize excellence and special contributions, so leverage it by nominating deserving personnel through the process on our website. We look forward to seeing a legion of Praetorians at next year's AvengerCon and, even sooner, at HammerCon 2.0 on Thursday 18 May at Capitol Technology University in Laurel, MD. Thank you for your service to our nation!

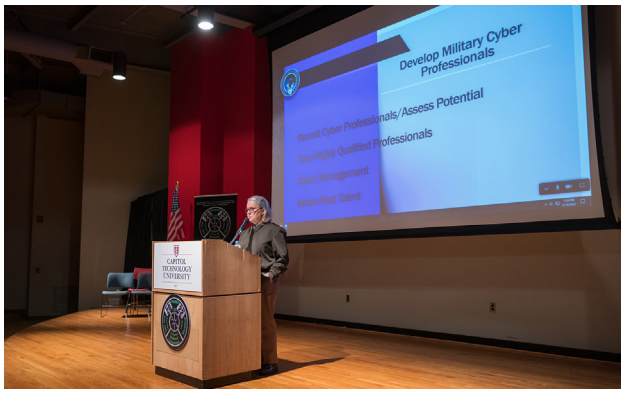


Photo of CSM Sheryl Lyon (SEL of USCYBERCOM and NSA), keynote at HammerCon 2022.

Photo of the AvengerCon VII team recognized with the Bronze Order of Thor medal by MG (Retired) George Franz (Inaugural Commander of the CNMF and member of the prestigious MCPA Board of Advisors), Colonel Brad Rhodes (AvengerCon speaker and MCPA Colorado Chapter President), and CW5 (Retired) Craig Jones (MCPA Fort Meade Chapter President).

Tool Developer Qualification Course

FORT GORDON, Ga. – Soldiers, Family and Friends celebrated the accomplishments of Class 2023 after they successfully completed the graduation requirements for the 780th Military Intelligence (MI) Brigade (Cyber) Tool Developer Qualification Course (TDQC) in a ceremony hosted by the 11th Cyber Battalion at the Friendship Chapel, February 10.

Command Sergeant Major (CSM) Ronald Krause, Cyber & Electronic Warfare Corps CSM, was the guest speaker.

The ceremony recognizes the achievements of nine enlisted Soldiers from the 782nd MI Battalion (Cyber) and the 11th CYB, who each completed the 11-month TDQC program taught in partnership with University of Maryland, Baltimore County (UMBC).

Graduates of the TDQC are proficient to an intermediate level in creating programs using the C and Python computer programming languages. The TDQC provides an education path for individuals to become experienced at approximately 90 percent of the identified critical developer requirements that an individual must be able to articulate and demonstrate through practical application to be certified as a Cyberspace Solution Engineer.

Congratulations to our TDQC graduates:

- SSG Alex Prufer, 11th Cyber Battalion, Distinguished Honor Graduate
- SSG Thomas Coop, 11th Cyber Battalion, Honor Graduate
- SGT Nicholas Barnes, 11th Cyber Battalion
- SGT Andrew Gort, 11th Cyber Battalion
- SPC Tyler Besler, 11th Cyber Battalion
- SPC Jackson Frink, 11th Cyber Battalion
- SGT Christopher Teal, 782nd Military Intelligence Battalion
- SGT James Lambert, 782nd Military Intelligence Battalion
- SGT Dawson Dittus, 782nd Military Intelligence Battalion

AvengerCon VII returns for a hybrid in-person and virtual event

COLUMBIA, Md. – AvengerCon VII, hosted every fall by Maryland Innovation and Security Institute to benefit the hackers of the U.S. Cyber Command community and the 780th Military Intelligence Brigade (Cyber), returned for a hybrid in-person and virtual event on November 30th and December 1st.

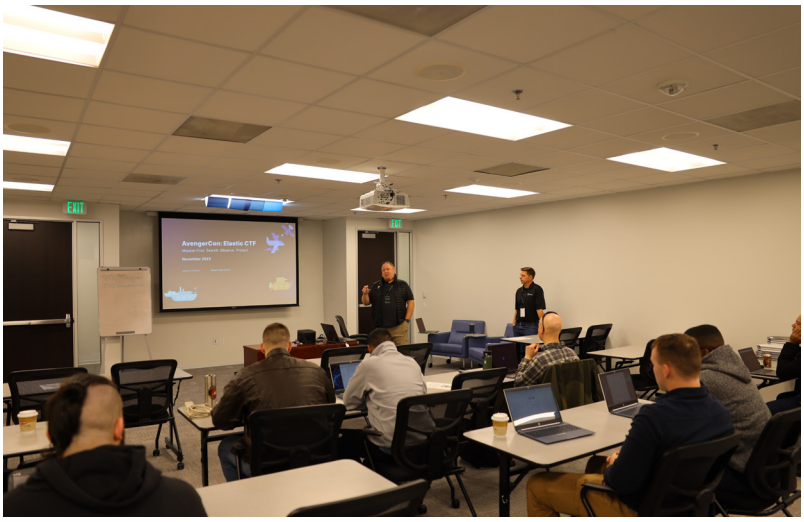
Training sessions on November 30 included: Intro to Reverse Engineering; Upgrading the Privacy of Android Smartphones; Provenance Tracking with Attack Graphs Using SysFlow; Using Containers to Analyze Malware at Scale; Rapid Prototyping of Machine Learning Solutions; Cyber Data Science for DCO (defensive cyberspace operations); Intro to Ghidra and Reverse Engineering; Developing Containerized Webapps in Python; and an Elastic Team CTF (capture-the-flag) event.

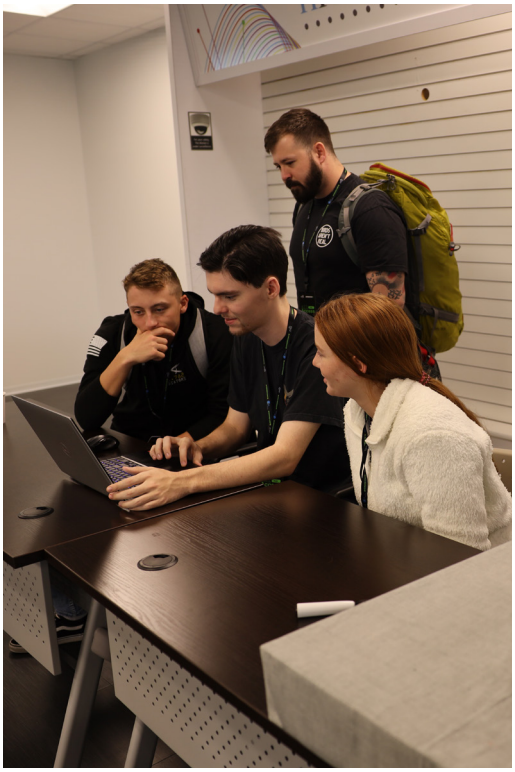
Peiter "Mudge" Zatko and Sarah Zatko were the keynote speakers for AvengerCon VII and kicked off day two; and the AvengerCon VII Panel on Crowdsourcing Security included: Katie Moussouris, CEO and Founder, Luta Security; TJ O'Connor, Professor, Florida Institute of Technology; and Colin Ahern, Chief Cyber Officer, New York State.

"AvengerCon is both space for people new to the InfoSec community to have a beginner-friendly area to see and experience the wider community and also allows the more experienced folks to have a spot to come together, proverbially let their hair down, talk to some of their peers in other organizations, and maybe connect afterwards," said Capt. Jacob Heybey, 780th MI Brigade, and one of the primary AvengerCon VII conference organizers.

For more information visit the AvengerCon website at https://avengercon.com/.


CAUTION


EXIT
AvengerCon: Elastic CTF




Sidebar: The Rule of Five
Choose a random sample of five from a population
There is a 93.75% chance that the median value of that population is between the smallest and largest values in the random sample of five.

781st Military Intelligence Battalion (Cyber)

FORT INDIANTOWN GAP, Pa. – The 781st Military Intelligence Battalion (Cyber) conducted a weapons qualification range under the leadership of 1LT Anna Devries, range OIC, and SSG Hayden Brown, range NCOIC, February 23. LTC Donald Sedivy, the battalion commander, awarded both Soldiers with an Army Achievement Medal for their outstanding work and devotion throughout the planning and execution of the range.

UBI CETERI NON POSSUNT

Vanguard... *When Others Cannot*

780th Military Intelligence Brigade (Cyber) - Strong Bonds

WASHINGTON – The 780th Military Intelligence Brigade Unit Ministry Team facilitated a couples and Families training event to build strong and ready teams at the Hyatt Place Hotel, March 10. Chaplain (Major) Frances Igboeli, brigade chaplain, led the training event using Stephen R. Covey's The 7 Habits of Highly Effective Families as a guide.

"The UMT (brigade unit ministry team) facilitated the 7 Habits of Highly Effective Families in order to transform lives. The curriculum is a great way to equip families and Soldiers with habits/tips to succeed in their respective Families and roles. Successful Families don't just happen; they take every bit of energy, vision, prioritization, etcetera."

Chaplain (MAJ) Igboeli

782nd Military Intelligence Battalion (Cyber Legion) - Strong Bonds

AUGUSTA, Ga. – The 782nd Military Intelligence Battalion (Cyber Legion) chaplain, Ch. (Captain) Joshua Calmes, hosted a family Strong Bonds retreat at the Partridge Inn for 36 adults and 15 children, February 26. The retreat included family events and childcare during the adults-only portions, and the adults were able to have a "date night" after the retreat event concluded.


EXIT

782nd Military Intelligence Battalion (Cyber) - Vigilant Wellness Event

AUGUSTA, Ga. – The 782nd Military Intelligence Battalion (Cyber) staff held a vigilant wellness event led by Chaplain (CPT) Joshua Calmes, the battalion chaplain, at the Farmhaus Burger in downtown Augusta, January 27. Under U.S. Army Intelligence and Security Command Holistic Health and Fitness guidance, CH Calmes led a discussion on renewal with 18 members of the battalion staff during a "Staff Renewal Lunch." It was a time of fellowship and an opportunity for the battalion staff to get to know one another. In addition to the Executive Officer, there was representation from the S1 (Personnel), S2 (Intelligence), S3 (operations), S4 (Logistics), and Civilian Personnel Office. Cyber Legion, Silent Victory.

FORT GEORGE G. MEADE, Md. – A Company (Avengers), 781st Military Intelligence Battalion (Cyber) Change of Command ceremony whereby CPT Phillip Arnold relinquished command to CPT Nolan Hedglin in a ceremony hosted by LTC Donald Sedivy, the battalion commander, at the MG Baron DeKalb Army Reserve Center, March 16. Vanguard…When Others Cannot!.

FORT GORDON, Ga. – Bravo Company, 782d Military Intelligence Battalion (Cyber) conducted a morale-boosting Combatives tournament at Victory Fitness Center, February 3. Eight personnel competed for the coveted title of “best fighter in the company” and the Birds of Prey company coin. To make it interesting, as the matches progressed, weight classes combined for tougher challenges to fight through. 14 grueling matches, over the course of two hours, with minimal rest, resulted in SGT Dooley (1st place), CPL Williams (2nd place), and SSG Bagley (3rd place).

FORT GEORGE G. MEADE, Md. – Soldiers and Civilians from the Headquarters & Headquarters Company, 780th Military Intelligence Brigade (Cyber), participated in a board game day event supporting Vigilant Wellness which promotes holistic wellness in the Brigade Annex, January 20.

FORT GEORGE G. MEADE, Md. – Soldiers and Civilians representing the Headquarters & Headquarters Company, 780th Military Intelligence Brigade (Cyber), participated in a video game day as part of the U.S. Army Intelligence and Security Command Resiliency program in the Brigade Annex, February 24.

FORT GEORGE G. MEADE, Md. – The 780th Military Intelligence Brigade (Cyber) with support from United States Navy Chief's Mess hosted a retirement ceremony to honor the service of Sergeant First Class Prince Yohannes in the Post Theater, March 17.

WHEELER ARMY AIRFIELD, Hawaii. – With support from the 25th Infantry Division Combat Aviation Brigade, the Soldiers of Detachment Hawaii, 782nd Military Intelligence (MI) Battalion (Cyber), and senior leaders representing the 780th MI Brigade (Cyber), attended the promotion ceremony of SFC Justin Riopelle, January 31. The oath was administered by COL Ben Sangster, the 780th MI Brigade commander.

FORT GEORGE G. MEADE, Md. – Soldiers, Family, and friends were on hand for the promotion of SGM Richard Harrison, the Brigade S3 Operations Sergeant Major, 780th Military Intelligence Brigade (Cyber), in a ceremony hosted by LTC Dave Raser, Brigade S3, in the Brigade Annex, January 12.

WASHINGTON. – Soldiers and Army Civilians from the 780th Military Intelligence Brigade (Cyber) visited Arlington National Cemetery and took a group tour of the Pentagon, encountering a special guest, Sergeant Major of the Army, SMA Michael Grinston, on March 17.

JOINT BASE SAN ANTONIO, Texas. – The Soldiers and Army Civilians of Detachment Texas, 782d Military Intelligence Battalion (Cyber Rangers), hosted a retirement ceremony for Chief Warrant Officer 4 Kevin McKee to honor his selfless service and his Family's sacrifices. Soldier for life!

In Memoriam CPT Hunter Phillips

CPT Hunter Ian Phillips was born one of two children on May 10th, 1994, in Gelnhausen Germany, to Colonel and Mrs. Travis Phillips. Growing up in the Army, Hunter lived at various Army installations before graduating from Palos Verde High School in Rancho Palos Verde, California in 2012. CPT Phillips received a Regular Army Commission from the United States Military Academy in 2016, as an Infantry Officer.

After completion of the Infantry Officer Basic Course, Hunter was assigned to 1-32nd Infantry Battalion, 1st Brigade, 10th Mountain Division, where he would deploy to Djibouti Africa as a Rifle Platoon Leader in 2018 and earned the coveted Ranger tab in 2019. Hunter later served with distinction as a Rifle Company Executive Officer and a Battalion Assistant Operations Officer in 1-32nd Infantry Battalion. Upon transferring to the Military Intelligence Branch, Hunter was assigned to the 11th Cyber Battalion where he would serve as a Signals Intelligence Officer, deploying to Kuwait in 2022.

During his service, CPT Phillips was awarded the Meritorious Service Medal, Army Commendation Medal with one oak leaf cluster, Army Achievement Medal with one oak leaf cluster, National Defense Service Medal, Global War on Terrorism Expeditionary Service Medal, Global War of Terrorism Service Medal, Army Service Ribbon, Army Ranger Tab, Expert Infantryman Badge and the Air Assault Badge.

CPT Phillips is survived by his wife, Robin, his beloved sister, Alma, and his parents, Travis and Catarina (Sanchez) Phillips.


PHILLIPS

NEXT QUARTER'S BYTE IS focused on the Brigade's Commissioned Officers. As in other issues of the BYTE magazine, the command encourages your contribution to drive the Cyber and Information Advantage conversation. If you have an article to share, write a synopsis and send it to steven.p.stover.civ@army.mil NLT May 15, 2023. Final articles are due May 31.