

**UNITED STATES DISTRICT COURT  
EASTERN DISTRICT OF NEW YORK**

MICROSOFT CORPORATION, a Washington Corporation, FORTRA, LLC, a Delaware Limited Liability Company, and HEALTH-ISAC, INC., a Florida Corporation,

Plaintiff,

v.

JOHN DOES 1-2, JOHN DOES 3-4 (AKA CONTI RANSOMWARE GROUP), JOHN DOES 5-6 (AKA LOCKBIT RANSOMWARE GROUP), JOHN DOES 7-8 (AKA DEV-0193), JOHN DOES 9-10 (AKA DEV-0206), JOHN DOES 11-12 (AKA DEV-0237), JOHN DOES 13-14 (AKA DEV-0243), JOHN DOES 15-16 (AKA DEV-0504), Controlling Computer Networks and Thereby Injuring Plaintiffs and Their Customers,

Defendants.

Case No.

**FILED UNDER SEAL**

**DECLARATION OF JONATHAN M. GROSS IN SUPPORT OF APPLICATION FOR  
AN EMERGENCY *EX PARTE* TEMPORARY RESTRAINING  
ORDER AND ORDER TO SHOW CAUSE RE PRELIMINARY INJUNCTION**

I, Jonathan M. Gross, declare as follows:

1. I am a Principal Research Manager in Microsoft Corporation's Security Research Organization, primarily supporting Microsoft Defender Threat Intelligence ("MDTI"). I make this declaration in support of Microsoft's Application for An Emergency Temporary Restraining Order and Order To Show Cause Re Preliminary Injunction. I do so of my own personal knowledge or on information and belief where noted. If called as a witness, I could and would testify competently to the truth of the matters set forth herein.

2. I have been employed by Microsoft since October 2021, having joined the company through the acquisition of RiskIQ Inc., a network infrastructure intelligence company ("RiskIQ").<sup>1</sup> I lead a team of security researchers responsible for developing Microsoft's customer-facing threat intelligence products, including host and network indicators of compromise, methods of detecting malicious activity, and finished analysis of state-aligned and criminal threat actor groups and their behavior. I am also the team lead responsible for improving Microsoft's ability to detect malicious network infrastructure and integrating those detections into MDTI and Microsoft's other security solutions. Prior to joining Microsoft, I was Senior Director of Threat Intelligence at RiskIQ from 2020 to 2021. There, I led the company's transition into the external threat intelligence market by architecting a new cyber threat intelligence offering. From 2012 to 2018, I was Director of Threat Intelligence at Cylance, an antivirus company that was later acquired by BlackBerry. While at Cylance, I played a key role in the company's investigation of the 2015 breach of the Office of Personnel Management (OPM) and participated in the Congressional investigation that followed. Prior to that, from 2011 to 2012, I was a technical lead for RSA NetWitness' Incident Response team, managing multiple incident response engagements while performing host forensic analysis, malware analysis, and network forensic analysis. Before that, from 2009 to 2011, I helped launch and then led one of Mandiant's Incident Response teams, performing incident response and forensic analysis for Fortune 500, Fortune 100, and Fortune 50 companies. While there, I assisted

---

<sup>1</sup> Throughout this declaration I use RiskIQ when referring to the company prior to its acquisition by Microsoft in 2021. Comparatively, references to MDTI reflect the group's current activities.

in forensic litigation support for one of the largest theft-of-IP cases in recent history. I was also instrumental in Mandiant's discovery and detection of several of the Advanced Persistent Threat (APT) groups first identified by the company. From 2006 to 2007, while still a student in The George Washington University's forensic science graduate program, I was employed by the school's Information Security Office in various roles, including as Senior Computer Forensics and Information Security Systems Engineer. In October of 2007, before graduating, I was invited to teach forensic analysis to the United States Secret Service Electronic Crimes Task Force. I hold dual Bachelor of Science degrees from Tulane University in biomedical engineering and mathematics and a Master of Science in forensic science from The George Washington University. A true and correct copy of the current version of my curricula vitae is attached to this declaration as **Exhibit 1**.

3. My declaration concerns a commercially available framework called Cobalt Strike which is designed for use in penetration testing, red teaming, and adversary simulation. Included herein, is my investigation regarding the misuse of cracked versions of Cobalt Strike<sup>2</sup> and my conclusions related to my investigation.

## **I. COBALT STRIKE**

### **A. Background**

4. First released in 2012, Cobalt Strike was acquired by Fortra (formerly Help Systems) in 2020. It is marketed to give users "a post-exploitation agent and covert channels to emulate a quiet long-term embedded actor in your customer's network." Its many customizable features make it a popular choice for many adversary simulators and, ironically, actual adversaries as well. The latter frequently acquire cracked versions of the software through various means, such as theft or misappropriation.

5. Over the last decade, analysts at Microsoft and across the wider security community have repeatedly documented the use of cracked Cobalt Strike by threat groups, including those

---

<sup>2</sup> As used in this declaration and in others, cracked versions of Cobalt Strike refer to stolen, unlicensed, or otherwise unauthorized versions or copies of Cobalt Strike.

assessed to be acting in the interest of the Russian, Chinese, Iranian, and Vietnamese governments, as well as by criminal syndicates engaged in ransomware attacks and other financial schemes.

6. In a 2018 report published by the cyber security authorities of the governments of Canada, Australia, New Zealand, the U.S., and the U.K., these authorities warned that threat actors were using cracked Cobalt Strike, among other publicly available tools, to compromise organizations “across a wide range of critical sectors, including health, finance, government and defense.”<sup>3</sup>

7. Through the methodology that will be described below, my team’s research has established that nearly 90% of Cobalt Strike servers we were able to detect through daily scanning over the last several years almost certainly involved misuse of Cobalt Strike in the form of cracked copies of the product. Our analysis also shows that a significant subset of cracked Cobalt Strike servers detected within the last 30 days, more than 40%, can be directly tied to known or suspected malicious activity.

**B. Cobalt Strike Functionality**

8. As described on the company’s website, <https://www.cobaltstrike.com>, Cobalt Strike replicates “the tactics and techniques of an advanced adversary in a network.” The framework is comprised of several tools and features, including client-side reconnaissance, post-exploitation payloads, covert communication, spear phishing, attack packages, browser pivoting, attacker collaboration, and reporting and logging.

9. The following describes Cobalt Strike’s many features and components. To aid the Court in understanding these components, I have defined certain features and components as follows:

- a. **Cobalt Strike** is the Command & Control (“C2”) application itself, consisting of two principal components contained within the same Java executable (JAR)

---

<sup>3</sup> See *Alert (AA18-284A) Publicly Available Tools Seen in Cyber Incidents Worldwide*, CYBERSECURITY & INFRASTRUCTURE SECURITY AGENCY (Oct. 11, 2018), available at <https://www.cisa.gov/uscert/ncas/alerts/AA18-284A>.

file:

- i. **Team Server** refers to the server-side portion of Cobalt Strike. It accepts client connections, Beacon connections, and standard HTTP requests. By default, this executable listens on TCP port 50050.
  - ii. **Client** refers to the target-side portion of Cobalt Strike. This provides the primary means by which operators connect to a Team Server. Clients are commonly executed on the remote systems the operator wishes to control. Clients can be executed on several different operating systems including Windows, macOS, or Linux.
- b. **BEACON** is the name for Cobalt Strike's actual malware payload. This piece of code is responsible for making the connection to the Team Server. There are two types of BEACONS:
- i. **Stager** is an optional payload that can perform some basic checks and download and execute the fully-featured **backdoor** from the predefined C2 server.
  - ii. **Backdoor** can be directly downloaded and executed by a BEACON stager or by another malicious executable commonly called a "loader." The backdoor runs in memory and is responsible for establishing communication to the Team Server. This code allows the operator to execute commands on remote systems.

10. Each BEACON file contains a unique **configuration** that is stored within the executable. When accessed, analysts can parse these configuration files to reveal a well-defined structure which, in turn, yields several key bits of information, including: C2 server information, IP address or domain names; the network port and protocol used in communicating with the C2; any unique headers used when communicating with the C2; process injection techniques used (if any) as well as supporting configuration information; the encryption key to use when securing client and server communication; and watermark designed to be unique and tied to each legitimate

customer.

11. In addition to the key pieces of information stored in the configurations listed above, numerous other fields of value to analysts can also be parsed from configurations. A non-exhaustive list is included below as **FIGURE 1**. The fields listed are those my colleagues and I have found to be useful in detecting Cobalt Strike servers on the Internet.

<b>Field Name</b>	<b>Purpose</b>
Sleep Time	The time interval to wait between client to server communication with C2
Jitter	The maximum percentage to vary the Sleep Time by. Jitter and Sleep Time together control how often the BEACON will communicate with the C2
User Agent	Sets the string value to use in HTTP requests in the User-Agent header field
Spawn To (x64 & x86)	Contains the location to physically store the PE file on disk
Pipe Name	Named pipes are used in Windows for inter-process communication (IPC). These names can be uniquely defined by the user

**FIGURE 1: ADDITIONAL COBALT STRIKE DATA STORED IN CONFIGURATIONS**

## **II. LEGITIMATE VS. ILLEGITIMATE USE OF COBALT STRIKE**

12. Today, both legitimate and illegitimate versions of Cobalt Strike are available to users. These versions include the following:

- a. **Licensed** versions of Cobalt Strike include more features and fewer embedded artifacts such as the EICAR string. It should be noted that the presence of a license is not an indication that an operator is using Cobalt Strike with the permission of the client / defender. Licenses can be and are stolen. Fortra typically revokes stolen licenses, which prevents the user from being able to receive updates. However, the stolen license will continue to work until its standard expiration.
- b. **Trial** versions of Cobalt Strike include numerous markers/signatures in them so that they can be more easily detected.

- c. **“Cracked”** which are comprised of stolen, unlicensed, or unauthorized versions of Cobalt Strike. Cracked versions of Cobalt Strike are usually distributed in online forums or via online chat programs such as Discord or Telegram. These are typically the result of someone modifying the contents of the original, legitimate JAR file itself to either bypass the license check entirely or create a fake authorization file that can then be bundled within the JAR.

13. Many versions of the authentic Cobalt Strike software, including “trial” versions, have ended up in the hands of cracked users because of the conduct of threat actors. In some cases, decompiled code has leaked, allowing threat actors the ability to further impede attribution. In November of 2020, for example, the press reported that the source code for Cobalt Strike 4.0 was leaked to GitHub and accessible to anyone. JAR files can also be trivially decompiled and recompiled by a seasoned reverse engineer. *See* Lawrence Abrams, *Alleged Source Code of Cobalt Strike Toolkit Shared Online*, BLEEPING COMPUTER (Nov. 11, 2020).<sup>4</sup>

14. From the perspective of an analyst or network defender, the only difference between sanctioned use of legitimate Cobalt Strike and malicious use of cracked Cobalt Strike is the authority or permission granted to the operator by the client. They otherwise function exactly the same way. Since it is not always possible to know when Cobalt Strike is being used with permission, analysts have therefore developed techniques to identify cracked versions of Cobalt Strike through other means. One such method involves the use of **watermarks**. For a discussion of how Microsoft used Cobalt Strike watermarks in their investigation *see* Declaration of Christopher Coy (“Coy Decl.”) filed concurrently with Plaintiffs’ Application for an Emergency *Ex Parte* Temporary Restraining Order and Order to Show Cause re Preliminary Injunction, ¶¶ 8, 20; and the Declaration of Jason Lyons (“Lyons Decl.”) filed concurrently with Plaintiffs’ Application for an Emergency *Ex Parte* Temporary Restraining Order and Order to Show Cause re Preliminary Injunction, ¶¶ 12-13, 20. Fortra utilizes these watermarks to distinguish legitimate

---

<sup>4</sup> Available at <https://www.bleepingcomputer.com/news/security/alleged-source-code-of-cobalt-strike-toolkit-shared-online/>

and illegitimate use and maintains a list of watermarks for all legitimate versions of the framework today. See Declaration of Robert Erdman (“Erdman Decl.”) concurrently with Plaintiffs’ Application for an Emergency *Ex Parte* Temporary Restraining Order and Order to Show Cause re Preliminary Injunction ¶¶ 6-8.

15. Once the configuration file is decoded and accessed, the watermark can readily be modified or removed entirely by a skilled operator. Accordingly, the prevalence of a high number of repeated watermarks within the Cobalt Strike ecosystem allows one to easily infer which watermarks are shared and therefore likely linked to cracked copies. I have also observed that legitimately licensed operators typically cannot and do not use hundreds of Cobalt Strike servers simultaneously, so the appearance of the same watermark across a large number of servers at any given time is another indication of likely illegitimate use. Leveraging these observations, my team and I have been able to link certain watermarks directly with specific, known threat actors (*see* **FIGURE 2** below). We have also been able to assess, with high confidence, that the following watermarks are directly associated with cracked versions, based on their prevalence across our data over the past several years.

<b>Watermark Value</b>	<b>Unique Beacons (Collected by MDTI)</b>
0	41,443
305419896	32,636
1234567890	23,243
426352781	20,076
1359593325	15,304
1580103824	9,370
100000	5,646
1	3,512
1873433027	3,238
206546002	2,387
6	2,339
391144938	2,091
999999	1,541
Non-Numeric Value	1,515
1711276032	1,473
666666	1,386
1580103814	1,302
305419776	878



16777216	829
987654321	566
574247	449
388888888	417
8848	394
668694132	386
3	334
12345	329
2130772225	300

**FIGURE 2: WATERMARKS ASSOCIATED WITH CRACKED COBALT STRIKE**

### **III. INVESTIGATION INTO CRACKED COBALT STRIKE**

16. In late 2020, MDTI began to actively track cracked versions of Cobalt Strike on a daily basis. Our methodology (described *infra* at ¶¶ XX), was informed by several conversations with colleagues at Rapid7. Rapid7 had been using RiskIQ data to seed their own Cobalt Strike scans. Rapid7 was just one of many security companies devoting attention to finding and stopping cracked versions of Cobalt Strike.

17. By this point, the security community had been publishing research documenting the misuse of cracked versions of Cobalt Strike by state-aligned and criminal syndicate groups. Misuse of cracked Cobalt Strike had led to ransomware attacks that had evolved in both frequency and sophistication. These destructive attacks were affecting a whole range of organizations in the business, government, and non-profit sectors, impacting critical infrastructure, schools, and even entire governments. *See* Lyons Decl. ¶¶ 26, 38.

18. Many of these ransomware attacks involved the use of cracked Cobalt Strike for lateral movement through victim environments and/or deployment of ransomware. Because of the penetration testing functionalities of the legitimate versions of Cobalt Strike, threat actors soon found that use of cracked versions (illegitimate versions of the same software with the same functionality) provided an immediate advantage to ransomware operators.

19. My team’s attention was further drawn to Cobalt Strike in October 2020, when

press reports indicated that ransomware operators had begun to focus their efforts on hospitals and the healthcare industry. *See also* Declaration of Errol Weiss (“Weiss Decl.”) filed concurrently with Plaintiffs’ Application for an Emergency *Ex Parte* Temporary Restraining Order and Order to Show Cause re Preliminary Injunction, ¶¶ 14-15.

20. My team at RiskIQ published an article, enumerating a list of servers deploying cracked Cobalt Strike that ultimately delivered Ryuk ransomware, the same strain used in the attacks targeting healthcare providers. *See* Team RiskIQ, *RiskIQ Has Released Its Corpus of Infrastructure and IOCs Related to Ryuk Ransomware*, RISKIQ BLOG (Oct. 30, 2020)<sup>5</sup> and *Ryuk Ransomware: Extensive Attack Infrastructure Revealed*, RISKIQ THREAT INTEL PORTAL.<sup>6</sup>

## INVESTIGATIVE METHODOLOGY

21. In connection with tracking cracked Cobalt Strike C2 infrastructure daily, my team relied on, in part, investigations from other cyber security companies who were also investigating and tracking cracked Cobalt Strike C2 infrastructure. For example, Fox-IT had already been tracking cracked Cobalt Strike and released some of the findings publicly in 2019, and then a fulsome set of their historic Cobalt Strike scanning data in 2022. Additionally, my team relied on public data available from other cybersecurity companies such as Censys, Shodan, ZoomEye, and Quake.

22. Because one of the functionalities of legitimate Cobalt Strike is penetration testing, the software needs to be designed in such a way to avoid detection. Because cracked Cobalt Strike relies on the same functionalities, a problem we were facing was how to detect the cracked C2 infrastructure—in fact, it was common for cybersecurity companies tracking cracked Cobalt Strike to only see 60 to 70% of the relevant infrastructure. To address this, MDTI sought to account for these gaps and to identify additional edge cases. My team identified a number of patterns common in the network protocol response that Cobalt Strike Team Servers served when a specially crafted

---

<sup>5</sup> Available at <https://web.archive.org/web/20201031235721/https://www.riskiq.com/blog/external-threat-management/ryuk-ransomware-indicators/> (last accessed Feb. 13, 2023). Because RiskIQ was subsequently acquired by Microsoft, the previous RiskIQ Blog website has since been sunsetted. Thus, the link is a Web Archive version of the same blogpost that my team published in October 2020.

<sup>6</sup> Available at <https://community.riskiq.com/article/0bcefe76> (last accessed Feb. 13, 2023).

HTTP request was made, as was done in our daily scans.

23. MDTI Team specially crafted our HTTP requests in these scans, which included what is commonly referred to as a “Checksum8” request. Checksum8 is the name of a particular function within the Team Server JAR that is responsible for checking valid requests to the server. It works by adding up the ASCII (American Standard Code for Information Interchange) values of each character and then taking the least significant byte. By default, this function only accepts the request if the number corresponds to 92 (in the case of x86) or 93 (in the case of x64). More sophisticated operators can readily modify the code of this function to deter automated extraction of the payload from the Team Server, but most do not.

24. In general, the default setting of most webservers elicits the same set of responses when an HTTP request is made. We observed that webservers used by threat actors for malicious purposes often return responses that differ from the default response in unique ways. An example of such a unique response is included below as **FIGURE 3**. These unique responses are often repeated across all the servers they control and act as a fingerprint which, when combined with certain other aspects of tradecraft, can establish a pattern used to identify the attackers that are leveraging cracked Cobalt Strike for malicious purposes.

```
HTTP/1.1 404 Not Found
Date:
Content-Type: text/plain
Content-Length: 0

and

HTTP/1.1 404 Not Found
Date:
Server: Apache
Content-Length: 0
Keep-Alive: timeout=10, max=100
Connection: Keep-Alive
Content-Type: text/plain
```

**FIGURE 3: UNIQUE HEADER RESPONSE EXAMPLE FROM SERVERS ASSOCIATION WITH CRACKED COBALT STRIKE**

25. In the case of illegitimate users of Cobalt Strike, those patterns included use of a

default SSL certificate, 6ece5ece4192683d2d84e25b0ba7e04f9cb7eb7c, that operators routinely neglected to change. My team used these patterns, and others, to then seed additional Internet Protocol (IP) address and port combinations to our custom scanner and expand our search. We looked for these patterns across publicly available data (e.g., those that had been published by Censys and Shodan) as well as our own internet scans. This allowed us to identify even more cracked Cobalt Strike Team Server patterns and in turn search for them across all available data sets. MDTI stored and analyzed the unique response to the HTTP requests, including associated metadata. We observed over time that these server response patterns would regularly change, and concluded that malicious actors were most likely to modify banner headers when regularly deploying publicly available code. For example, we found operators deploying Ryuk ransomware used a publicly available, cracked version Cobalt Strike concealer to attempt to hide their illicit Cobalt Strike deployments. Our investigation thus focused on identifying as many publicly available concealers as possible. We found some examples on GitHub.

26. We used certain characteristics contained within the configuration profiles they create or the response patterns they employ to readily detect them. We observed cracked Cobalt Strike copies are regularly used with these types of concealers. As an example, **FIGURE 4** shows the top ten watermarks contained within profiles matching the use of Forty North’s C2 concealer

Watermark (Forty North C2 Concealer)	Frequency
0	2,013
1580103824	1,132
426352781	975
1359593325	776
1580103814	686
305419896	580
305419776	525
206546002	352
1	219
1234567890	195

**FIGURE 4: TOP 10 WATERMARKS ASSOCIATED WITH CRACKED COBALT STRIKE & RATE OF USE**

27. In investigating the above watermarks further, my team and I found that many of the individual Cobalt Strike BEACON configurations subsequently collected and associated with

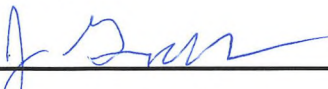
them could be definitively linked to specific, known ransomware groups or even individual operators themselves. *See also* Lyons Decl. ¶ 33.

28. MDTI continues to regularly update detection patterns of cracked versions of Cobalt Strike to keep pace with malicious actors. As of February 2023, we have collected 204,071 unique parsed configurations over the past few years. Using these, we can reliably, and with high confidence, continue to identify anywhere from 1,500 to 2,500 servers deploying cracked Cobalt Strike across the internet on any given day. Of the 204,071 configurations, 184,230 of the configurations contained a watermark corresponding to a cracked copy of Cobalt Strike. Over our multi-year collection efforts, 90% of the cracked Cobalt Strike team servers MDTI identified on the internet corresponded to illicit versions of the software.

29. To determine clear instances of illicit usage over a period of 30 days, MDTI looked specifically at the C2 servers used, the hosting provider for the IP address, length of time the IP was active, as well as unique facets of the configurations. Of the 4,363 profiles corresponding to cracked copies of Cobalt Strike, we assess that 1,878 of them almost certainly represented malicious activity. This means that within the last 30 days of this writing, at least 43% of the cracked Cobalt Strike Team Server configurations MDTI identified were highly likely to have been malicious or were associated with known threat actors.

I declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct to the best of my knowledge.

Executed this 30th day of March, 2023, in New York, New York.

  
\_\_\_\_\_  
Jonathan M. Gross

# **EXHIBIT 1**

**Professional Experience:**

**Research & Development – Microsoft**

Oct 2021 - Present

***Principal Research Manager***

- Led the Microsoft Defender Threat Intelligence (MDTI) team responsible for developing Microsoft’s customer-facing threat intelligence products, including host and network indicators of compromise, methods of detecting malicious activity, and finished analysis of state-aligned and criminal threat actor groups and their behavior.
- Produced actionable and impactful threat intelligence for customers of the MDTI product on a wide array of threat actors, platforms, and topics.
- Created several novel internal systems utilizing a new class of network-based indicator of compromise to aid in tracking and monitoring targeted threat activity across the internet.
- Collaborated with Microsoft’s MSTIC team to produce meaningful public facing threat research.
- Led the internal team responsible for improving Microsoft’s ability to detect malicious network infrastructure and integrated those detections into MDTI as well as Microsoft’s other security solutions.

**Research & Development – RiskIQ**

Oct 2020 – Oct 2021

***Senior Director of Threat Intelligence***

- Identified shortcomings and gaps within RiskIQ’s internet-scale collections. Drove modifications to these collection systems which enabled actionable threat intelligence to be more reliably delivered at scale.
- Created and developed a new class of threat indicator with the express purpose of reliably detecting attacker controlled C2 across the internet as well as through time.
- Designed, implemented, and deployed robust and comprehensive systems to analyze, collect, and attribute malware C2 data across the internet.
- Created and led RiskIQ’s Team Atlas which delivered numerous far-reaching and impactful public threat reports, including one that was personally addressed by the Russian Ministry of Foreign Affairs. Delivered timely and actionable insights to our customers via the PassiveTotal product.

**Project Management – RiskIQ**

Feb 2020 – Oct 2020

***Project Manager***

- Directed the development and engineering team responsible for RiskIQ’s PassiveTotal product.
- Enabled metrics collection across the platform at a scale that was not previously possible. Developed and created new avenues for growth and profit which immediately increased revenue by sixty percent in six months.
- Identified key personnel and external individual contributors inside the platform and devised ways to incentivize them to continue contributing despite the implementation of more restrictive community controls.
- Designed & developed the RiskIQ Intelligence Portal and all the associated content as well as mechanisms to collect and integrate open-source intelligence across platforms on the internet. Architected a new cyber threat intelligence offering.

**R&D/Marketing – APT Marketing LLC**

Apr 2018 - Present

***Founder & Principal Researcher***

- Founded my own company to deliver customized intelligence products to a variety of different clients with a primary focus on exposing novel targeted threats. Strived to achieve maximum marketing penetration and potential through direct interaction with clients’ internal marketing as well as internal and external public relations teams.
- Enabled clients to speak clearly and concisely about delivered intelligence products in public news outlets and top industry security conferences, including Black Hat. Established several clients as experts in their respective fields across a breadth of security-related topics.

**Research & Development – Cylance**

Dec 2012 – Apr 2018

***Director of Threat Intelligence***

- Architected and maintained autonomous systems for tracking advanced threats. Created predictive threat modeling based upon domain registration characteristics and anomalies.

- Performed malware analysis across a variety of platforms in support of the professional services team and for research and development purposes.
- Discovered and investigated new families and other variants of unknown targeted threats. Delivered comprehensive reports to trusted parties on these undiscovered threats.
- Researched and analyzed malware command and control infrastructure and network protocols. Created decryption routines to decode and decrypt these custom malware C2 channels and stored configuration data.
- Created a new method of rapid key extraction for variable key size XOR encoded binary data. Used this technology to pinpoint and study malicious office documents associated with targeted threats.
- Played a key role in the company's investigation of the 2015 breach of the Office of Personnel Management (OPM) and participated in the Congressional investigation that followed.

**Professional Services – NetWitness/RSA**

Mar 2011 – Nov 2012

***Principal Security Consultant/Senior Practice Consultant***

- Functioned as a technical lead for RSA's IR team. Validated the team's technical findings across active engagements and oversaw the delivery of final work product to clients.
- Led and managed incident response engagements while performing host forensic analysis, malware analysis, and network forensic analysis.
- Reverse engineered malware and malicious documents using static and dynamic techniques in support of incident response. Created custom network-based indicators to identify generic and targeted threats.
- Researched and analyzed targeted malware command and control infrastructure and protocols. Created decryption routines to decode and decrypt custom malware C2 channels.
- Developed a comprehensive threat intelligence program following several threat groups currently targeting the United States and other related economic interests.

**Commercial Consulting Services – Mandiant**

Aug 2009 – Feb 2011

***Security Consultant***

- Performed incident response and forensic analysis for Fortune 500, Fortune 100, and Fortune 50 companies. Analyzed the methodologies and malware associated with the advanced persistent threat (APT) and PCI breaches.
- Reverse engineered malware using static and dynamic techniques in support of incident response to create host and network-based signatures to scan the enterprise for similar targeted threats.
- Identified and analyzed newer host-based infection techniques and persistence mechanisms which avoided creating any artifacts within the Windows registry.
- Assisted in forensic litigation support for one of the largest theft of IP cases in recent history.
- Developed novel host-based forensic techniques which expedited uncovering evidence of data theft.

**Information Security Office – ISS, The George Washington University**

Jun 2007-Aug 2009

***Senior Computer Forensics and Information Security Systems Engineer***

- Analyzed compromised university systems through live and offline analysis to determine how the systems were exploited, if data was exfiltrated, and if a backdoor was placed onto the system.
- Reverse engineered current malware to determine its capabilities and generate network-based signatures.
- Led and taught a small team of engineers in analyzing custom IDS alerts, writing new signatures, and verification of potential compromises to eliminate false positives.
- Redesigned current IPS and IDS implementations to correspond with rapid changes in the threat landscape and underlying network infrastructure changes.
- Responsible for finding and testing new equipment to further heighten the university's security posture.
- Implemented NetWitness to provide extensive network-based forensic capabilities.

**Information Security Office – ISS, The George Washington University**

Sep 2006-Jun 2007

***Computer Forensics and Information Security Systems Engineer***

- Worked with FTK and EnCase for the purpose of university-related forensic investigations and system analysis/recovery, including acquisition, authentication, and analysis of digital media.
- Designed custom IDS systems including complete signature bases for Enterasys Dragon and Snort.
- Analyzed network traffic using several different means for the purpose of identifying and analyzing compromised university systems and servers.



- Performed live and offline incident response on compromised university systems and wrote formal reports detailing the findings.
- Conducted remote Web Application/Server assessments using Web Inspect and passive enumeration techniques.
- Analyzed and identified emerging threats to university systems which included an in-house analysis of the worm Curious Yellow before public disclosure in mid-December.
- Worked on undetectable/stealth malware and rootkit identification, automated disassembly of packed executables, and streamlining signature identification/design.

**Information Security Office – ISS, The George Washington University**

May 2006-Sep 2006

***Information Security Administrator***

- Worked with FTK and EnCase for the purpose of university-related forensic investigations.
- Designed and deployed a new IDS architecture based on Dragon 7.2.2.
- Assisted as part of a university wide penetration test which incorporated the use of tools such as Nmap, Amap, Nbtscan, Retina and Nessus.
- Conducted remote Web Application/Server assessments using Web Inspect, AppScan, and Paros.
- Identified common backdoor methodologies and developed unique methods for detecting anomalous traffic.

**Independent Malware and Vulnerability Researcher**

2001-Present

***Contract Security Consultant***

- Developed custom detection methods for identifying generic Trojans and malware through unique polymorphic and metamorphic executable packing signatures.
- Analyzed numerous public web exploitation kits to design novel detection methods of both obfuscated exploits and payload delivery.
- Researched common propagation and centralized control methods of recent internet worms and similar IRC/Web-based backdoors.
- Identified current methods used in antiviral heuristic analysis of unknown executables and analyzed their current effectiveness and weaknesses.
- Developed network-based signatures to generically identify backdoors and anomalous client-server traffic.

**Education:**

**Masters of Forensic Science**

High-Technology Crime Investigation. GPA 3.8

The George Washington University

December 2007

**Bachelors of Science**

Biomedical Engineering & Mathematics

Tulane University New Orleans, LA

Cum Laude w/ Departmental Honors May 2005

**Technical Skills:**

- Programming Languages: C, Perl, x86 Assembly, Python, Go, Rust, JavaScript, Java

**Papers and Presentations:**

United States Secret Service Electronic Crimes Task Force Quarterly Meeting: “The Evolving Threat Landscape (‘Real’ Case Studies)”. October 2007

Microsoft Digital Crimes Consortium (DCC): “Department of Pre-Crime: Cobalt Strike Profiling”. October 2022

**Additional Information:** Available Upon Request