MANDIANT
NOW PART OF Google Cloud

THREAT INTELLIGENCE
WHITE PAPER

Security Analyst Case Study: See and Stop Software Supply Chain Compromises

The content in this document was originally published in The Defender's Advantage Cyber Snapshot Issue 3.

Last year, Mandiant reported a significant increase in supply chain compromise–17% of intrusions over the course of 2021 started within the supply chain, up from <1% in 2020[1]. This increase is partially explained by the fact that 86% of the compromise intrusions Mandiant tracked were related to the SolarWinds breach and SUNBURST[2]. However, it also correlates to organizations maintaining technology relationships with an average of 244 vendors[3].

A software supply chain attack is nothing new. In 2017, the world was hit with the attack dubbed NotPetya. The malicious code, disguised as ransomware, exploited the NSA's leaked EternalBlue vulnerability to infiltrate networks and then systematically destroy data. The attackers behind NotPetya breached a financial services software company that was a supplier for the Ukrainian government.

In the same year, the utility CCleaner[4] suffered a breach and hackers were able to replace the legitimate version of the software with a malicious one, which resulted in the compromise of more than 2 million hosts.

In 2020, the aforementioned widespread attack leveraging a SolarWinds component was perpetrated by APT29 (previously UNC2452), a threat actor whose targeting is assessed to be consistent with Russian strategic interests[5]. The breadth of victims impacted by APT29 included government organizations and Fortune 500 companies. Once again, attackers targeted the software supply chain by injecting a backdoor code in the software component Orion, giving them access to the victims' internal environments and enabling them to deploy the SUNBURST malware after the updated code was distributed through a legitimate process.

Attackers have found a way to compromise the building blocks of our digital enterprise. By targeting and successfully compromising a popular package used by software developers, it is then easy to amplify the distribution of malicious code directly to victims themselves at scale. This approach leaves defenders asking if we are confident in our readiness to defend. Organizations worldwide are stretching to maintain visibility on their attack surface, and confidence in their detect and respond functions. Too often organizations are unsure of their ability to quickly see and stop cyber attacks within their software supply chain, in part because they lack properly trained defenders and don't activate–or respond–frequently enough to fine-tune their training and knowledge.

Software supply chain compromises are designed to abuse the trust in third-party providers to indirectly gain access to a victim's environment and can be difficult to detect. In the end, the security analyst's trained eyes and investigative process is the deciding factor in identifying and stopping advanced attacks.

> With supply chain attacks, a pre-established trust makes the malicious implant extremely difficult to detect directly. An activated and effective detection and response capability becomes even more critical as a suspicious event detected in the later stages of the same attack lifecycle allows analysts to discover the implant indirectly by rewinding the attackers actions through investigation.
>
> **— Steve Ledzian VP, CTO-APAC, Mandiant**

1. Mandiant, M Trends 2022
2. Mandiant, M Trends 2022
3. Mandiant, The Defender's Advantage Cyber Snapshot Issue 2, 2022
4. Mandiant Threat Intelligence, "CCleaner Supply-Chain Compromise Possibly Linked to Chinese Cyber Espionage Operators:, September 2017
5. Mandiant, "Highly Evasive Attacker Leverages SolarWinds Supply Chain to Compromise Multiple Global Victims With SUNBURST Backdoor", December 2020

Analyst Detection and Investigation of a Software Supply Chain Compromise

Starting in mid-October 2021, security analysts in Mandiant's managed detection and response service identified multiple events that appeared to be a poisoning of open-source repositories. The following case describes their detection and investigation process–and the questions they sought to answer–involving packages hosted on Node Package Manager (NPM), the package manager for the Node.js JavaScript platform.

A small team of Mandiant security analysts initially observed multiple alerts indicating that a native Windows utility **CERTUTIL.EXE** was being used to download payloads from a common URL **(hxxps://citationsherbe[.]at/sdd.dll)**. As more analysts in the Security Operations Center (SOC) started picking up similar alerts, the team began to work in coordination towards answers to their investigative questions.

What is it? How did it get downloaded onto the system?

The first investigative questions to be answered are, "What malware is present and what are its capabilities?" and "How did it arrive on the system?" Analysts acquired the payload from initial hosts to determine the functionality and capabilities of the suspicious binary. Triage analysis indicated that the binary was a variant of the DANABOT malware, which targets credentials for theft through communication with an attacker-controlled command and control (C2) server. Using the malware's C2 address, analysts began to further scope the environment by identifying other systems communicating with the attacker infrastructure. This process allows analysts to determine if the same or similar malware may have been deployed on other systems without a corresponding alert. Once the payload is confirmed malware, the analyst team proceeded to contain the compromised hosts remotely or by initiating the incident response team to act.

How did it get there?

To understand how the malware was deployed, analysts typically rely on data collected by endpoint detection and response (EDR) technologies. By reviewing EDR telemetry, the analysts traced the activity to legitimate commands executed by users to update NPM packages.

Thorough investigation revealed that each of the affected hosts had a similar file written to the **UA-PARSER-JS PACKAGE** directory, which led the analysts to believe it was compromised and distributing malware. The malicious change to the JS Package directory added a preinstall step to the package installation process, which downloaded the malware. In reviewing the compromised script, the analysts found that it also downloaded and deployed coinminers (also called cryptocurrency miners) to the host. Analysts checked the GitHub issues for the package repository and found a question where someone had asked if the package was very recently compromised. According to a GitHub issue raised on October 22, 2021, at approximately 12:15 UTC, the NPM package **"ua-parser-js**", a popular Node.js library that amassed over 7 million downloads per week, was compromised to deliver malware. The threat actor was able to publish three malicious versions of the package by hijacking the author's NPM account. According to the repository's Git log, on October 22, between 16:14 UTC and 16:25 UTC, the package author committed a sanitized version of the malicious packages to stop further compromises.

DANABOT is a backdoor written in Delphi that communicates using a custom binary protocol over TCP. The backdoor implements a plug-in framework that allows it to add capabilities via downloaded plugins. DANABOT's capabilities include full system control using a VNC or RDP plugin, video and screenshot capture, keylogging, arbitrary shell command execution, and file transfer. DANABOT's proxy plugin allows it to redirect or manipulate network traffic associated with targeted websites. This capability is often used to capture credentials or payment data. DANABOT can also extract stored credentials associated with web browsers and FTP clients.

"ua-parser-js" is a lightweight, small footprint package deployed within a web application or server-side application to extract and filter the relevant data needed to parse a User Agent string (i.e., Browser, Engine, OS, CPU, and Device).

What other activity was performed by this threat actor?

After the hosts were contained, the analysts continued researching to determine the root cause of the attack. Reviewing the git log of the package repository, the analysts found timestamps for when the malicious change was pushed and when the fix was applied a few hours later. Further analysis of attacker TTPs allowed the team to link additional NPM packages, compromised by the same attacker, and scope the extent of the activity performed by the threat actor. The team was able to attribute the activity with reasonable confidence to UNC3379, analyze the malware, document attacker behavior, and develop detection techniques to thwart future activity.

For more information on this software supply chain compromise, review the research blog, No Unaccompanied Miners: Supply Chain Compromises Through Node.js Packages.

Trust analyst instinct, critical thinking and experience

Regardless of the scale of the investigation, time is of the essence. Mandiant relies on our analysts' knowledge, training, and critical thinking in investigation and response. Our team operates like detectives, leveraging the clues, evidence, and forensic artifacts to uncover the story behind each incident. The goal of the investigative process is to answer key questions about the attack to determine:

- Scope of the intrusion
- Whether it is still ongoing
- Earliest date of compromise and cause of the intrusion
- Type and extent of data exposed
- Threat actor Identity and motives

Understanding these facts about the intrusion will guide your containment, eradication, and recovery. Through frontline experience, simulation, and training, Mandiant recommends empowering your analysts to lead investigations and make key decisions around the timing and execution of containment and eradication. "It's common for organizations performing their own incident investigation and response to prematurely jump to remediation," says Eric Scales, Vice President Mandiant. "The more that you understand about the attack, the greater the success in eradication and recovery."

In the case presented here, Mandiant MDR's analyst-led investigation developed key atomic indicators related to the activity, performed triage of the malware deployed to determine the appropriate remediation actions, and, using our in-depth knowledge and research about the threat group, successfully scoped the environments of our customers to discover additional malicious activity related to this campaign that was not detected by their EDR products.

Read more articles from **The Defender's Advantage Cyber Snapshot.**