Analysis of attack activities of APT-C-20 (APT28) using compound attack tactics

Advanced Threat Institute 360 Threat Intelligence Center October 10, 2024 12:07

APT-C-20

APT28

APT-C-20 (APT28), also known as Fancy Bear, Sofacy or Sednit, has been active in cyberspace for more than ten years. Its attack targets cover many countries and regions around the world, involving governments, military, media, energy, etc. a key area.

APT-C-20 (APT28)) is famous for its superb technical means and complex attack strategies. They are good at using a variety of attack vectors, such as spear phishing emails, waterhole attacks, zero-day vulnerabilities, etc., and combining them with a variety of technical means, such as malware, remote control tools, encrypted communication protocols, etc., to achieve target system control. Penetration and control. At the same time, APT28 also attaches great importance to covering up and disguising attack behaviors. It increases the difficulty of traceability and attribution by using proxy servers, forged file attributes and other means.

During the continuous tracking of APT28, 360 Advanced Threat Research Institute found that the organization used a variety of complex attack techniques to launch network attacks. This report will focus on analyzing and dissecting its three most active attack tactics, and deeply reveal APT28's intrusion path, tools used, techniques and tactics in recent attack activities, and the strategic intentions behind it. At the same time, this report will also evaluate the potential impact of these attack activities and propose corresponding defense suggestions and countermeasures. This research not only helps to deepen the understanding and knowledge of APT28, but also hopes to provide some experience and inspiration for dealing with increasingly complex network threats.

1. Analysis of attack activities

1.Category 1_Headlace

1.1 Attack process analysis

In a typical attack scenario, APT28 attackers first send carefully constructed phishing emails to target users. The body of the email usually contains links to malicious compressed files. Once the user downloads and opens the compressed file, Headlace Dropper will use some disguise methods to induce the user to execute the file, such as a file named Windows Update, a web link or an LNK file whose icon is disguised as a document. In some cases, attackers also use DLL hijacking techniques to load the Headlace Dropper when users open legitimate applications.

In addition to malicious compressed files, we also found that APT28 uses a variety of bait formats such as LNK shortcut files and malicious URLs to increase the success rate of the attack. Once the Headlace Dropper is successfully executed, it will further release the more powerful Headlace backdoor. The backdoor can establish communication with the attacker's command and control server and perform various malicious operations on the victim's system, such as stealing sensitive information, downloading additional malicious components, etc., and ultimately achieving long-term control of the target system.

Figure 1 Attack flow chart

1.2 Malicious payload analysis

Headlace-type attack campaigns typically begin by sending an email containing a malicious link to the target. Attackers carefully design email content to trick victims into clicking malicious links.

Figure 2 Email example

Before delivering the malicious payload to the victim, the attacker will use JavaScript to perform a series of verifications, such as checking whether the user agent contains "win" and does not contain "wow" (which may indicate a virtual machine), or checking whether the renderer name contains "vmware" ", "virtual", "google" or "engine" and even in some cases confirm the victim's geographical location to implement geofencing. This step can help attackers screen targets and improve the accuracy of attacks.

Geofencing is a tactic whereby attackers use customized scripts or malware to selectively target specific areas (such as countries or regions) for attacks and data theft based on the geographic location of the target victim.

Figure 3 Browser check code example

Figure 4 Geofence check code example

Once verified, the attacker will deliver malicious compressed files to the target. These files are carefully disguised, often under the guise of tempting content such as Windows updates or pictures of models.

Figure 5 Code example for delivering compressed files

In the early stages of the Headlace attack, the compressed files contained malicious CMD code.

Figure 6 Example of files within Headlace compression

The main function of the CMD code is to create a BAT file and a VBS file, and execute the BAT file through the VBS file. At the same time, the attacker will also open related bait websites such as explicit model websites, or display a fake update progress to deceive others.

Figure 7 cmd code example

Figure 8 Example of bait website

Figure 9 Example of false update progress

The malicious BAT file uses the "headless" mode of the Microsoft Edge browser to access the specified URL. It will create a file with the extension ".css" in the victim's "%USERPROFILE%\Downloads" directory, then move it to the "%PROGRAMDATA%" directory, change the extension to ".cmd", and execute, And delete it after the execution is completed to hide the traces of the attack.

Figure 10 bat code example

In the recently observed Headlace attacks, attackers also used DLL hijacking techniques to execute BAT files. They implanted a legitimate Calc.exe binary file that was vulnerable to DLL hijacking attacks into the compressed package, inducing users to click to execute and then load the malicious DLL file. The function of the malicious DLL file is still to download and execute CSS files.

Figure 11 Example of DLL hijacking file

Figure 12 Malicious DLL code example

The purpose of this malicious script is to collect sensitive information on the user's computer and send it to a remote server. It mainly steals the file list of the user's home directory, desktop, downloads, documents and other directories, the information of the Program Files and Program Files (x86) directories, as well as the user's IP address and geographical location and other data.

Figure 13 Example of malicious bat code

Figure 14 Malicious CSS file code example

By continuously monitoring Headlace's activities, we also found that attackers used other types of initial payloads such as URLs and LNKs to use a multi-pronged approach to improve the success rate of attacks.

Figure 15 Code example for delivering compressed files

Figure 16 Example of files within Headlace compression

2.Category 2_Masepie

2.1 Attack process analysis

In another common attack method of APT28, attackers usually send phishing emails containing malicious links to target users. Once the user clicks on the link, they will be redirected to a bait page carefully designed by the attacker and induce the user to click on a specific button or link.

When users are fooled and click on the malicious button in the bait page, they will be further directed to a WebDAV server. On this server, the attacker pre-installed a malicious LNK shortcut file. Once the victim double-clicks these LNK files, a series of malicious activities will be triggered without their knowledge.

Through the LNK file, the attacker will use PowerShell commands to release multiple malicious components, including decoy documents used to confuse users, a Python interpreter used to execute malicious code, and a backdoor called MASEPIE.

Once the MASEPIE backdoor is executed on the victim system, the attacker can establish a stable remote control channel. Using this channel, the attacker can selectively deliver other attack components, such as STEELHOOK or OCEANMAP, to the victim system as needed to further expand control of the target environment. In addition, the MASEPIE backdoor also allows attackers to execute arbitrary commands on the victim system, which allows APT28 to flexibly adjust attack strategies to adapt to different target environments and attack needs.

Figure 17 Attack flow chart

2.2 Malicious payload analysis

In the initial phase of the attack, the attacker sends an email containing a link to a fake file to the target. When users click on these fake links, they see a vague image of a decoy document pretending to be related to the European Space Agency, enticing them to click a button to view the full document.

Figure 18 Example of bait URL

However, when the user clicks the button, the malicious code will actually use the characteristics of JavaScript and the search-ms application protocol to download an LNK file in the background. From the user's perspective, clicking the button only opens a File Explorer window, but in fact the malicious activity has already begun.

Figure 19 Page code example

This LNK file loads a remote decoy document and then executes malicious Python code through the remote Python interpreter.

Figure 20 LNK file code example

This malicious Python file belongs to the Masepie malware family. It is developed using the Python language and has functions such as file upload, download and command execution.

The sample first connects to the remote C2 server and sends a randomly generated AES key and system username.

Figure 21 main function code example

After the connection is established, the sample enters an infinite loop and continues to receive and execute commands issued by the server. These commands include:

- check: Send a "check-ok" message to confirm the connection status
- send_file: Start a thread and call the receive_file function to receive the file
- get_file: Upload a file from the victim computer to the server
- Other commands: Use os.popen directly on the victim computer to execute commands and return the results

Figure 22 receive function code example

The receive_file function is responsible for connecting to the C2 server, randomly generating an AES key and sending it to the server, then receiving the encrypted file name and size, sending confirmation information, and finally receiving the encrypted file content, decrypting it and saving it locally.

Figure 23 receive_file function code example

In subsequent attacks, attackers can use Masepie to deliver more types of malicious samples, such as OCEANMAP or STEELHOOK, to further expand control over the victim system.

Through the analysis of this APT attack, we can see how the attacker works step by step, from the initial bait email, to the malicious LNK file, to the Masepie malware, and finally may deliver other malicious tools. This multi-stage, multi-tool attack method increases the concealment and persistence of the attack, and brings great challenges to defense and elimination.

3.Category 3_Fishing

3.1 Attack process analysis

In APT28's phishing attack activities, attackers usually send well-designed phishing emails to target users, and the email attachment is usually a malicious compressed file. These compressed files contain attractive PDF documents or HTML files to attract the user's attention and entice them to open them.

When a curious user opens the compressed file and accesses the PDF document inside, they are further directed to a malicious HTML page. On this carefully disguised phishing page, users will be lured step by step to fill in their account credentials, such as usernames, passwords and other sensitive information.

Once a user enters their account credentials on a phishing page, this sensitive information can be silently stolen by the attacker. Attackers can use these stolen credentials to access various systems and resources within the organization as legitimate users, allowing for broader penetration and intelligence collection.

Figure 24 Attack flow chart

3.2 Malicious payload analysis

During our observations, we discovered multiple phishing attacks targeting Ukraine. In these campaigns, attackers send malicious emails to targets, often with a malicious zip file as an attachment. These compressed files may contain phishing HTML files or decoy PDF documents, with the content disguised as ukr.net login or password modification.

When the victim clicks the button in the bait PDF document, it will jump to a phishing webpage hosted on Mocky. This webpage is disguised as the login page of ukr.net, with the purpose of stealing the user’s login credentials.

Figure 25 Example of phishing PDF document

In addition to bait PDF documents, attackers also use phishing HTML files. The main purpose of these files is again to collect user credentials, but they also contain a range of malicious code.

Figure 26 Example of phishing page

Figure 27 Example of phishing page

These malicious codes mainly use the XMLHttpRequest JavaScript object to send the captured user credentials to the remote C2 server, and then wait for the server's response. Based on the string , the phishing page will display specific dynamic web content to the victim:

- "Finaly": Indicates that the identity verification is completed, and the page will cancel the fuzzy occlusion, or guide the user to enter a new password and other subsequent operations.
- "Redirect": The page will redirect to the real "http://mail.ukr.net/" to conceal the phishing behavior.
- "AGAIN": implies that the server requires resending data.
- "BAD": The page will display an error message, suggesting that the user entered incorrect credentials.
- "DATA=": The server will return JSON data, and the page will parse the data and dynamically update the web page content.

In order to confuse users more effectively, the server will also return data such as verification codes to make the phishing page look more authentic and credible.

Figure 28 Phishing page code example

By analyzing these phishing attack activities, we can see how attackers carefully design phishing emails and web pages, using various carriers such as compressed files, PDF documents, HTML files, etc., combined with malicious JavaScript code, to dynamically interact with the C2 server to achieve theft Purpose of user credentials. This complex phishing attack method puts forward higher requirements for user education and security defense.

2. Attribution research and judgment

As early as the beginning of APT28's widespread attacks on Europe and the Caucasus, foreign security agencies issued relevant notices, pointing out that the APT28 organization used malicious components such as Headlace and Masepie to carry out a series of network attacks [1][2] . Through continuous monitoring and analysis of this organization, we found that APT28’s attack activities have expanded to other countries in Europe and the Caucasus region.

The phishing techniques used in these attacks are completely consistent with the techniques and tactics consistently used by APT28. In addition, attackers used compromised Ubiquiti Edge routers to collect user credentials. This method has been previously disclosed by multiple security research institutions and highly matches the attack characteristics of APT28 [3] .

Based on the above analysis, we have every reason to believe that this series of attacks were planned and implemented by the APT28 organization. APT28's activity level and attack scale in the region indicate its strong interest and continuous penetration attempts in the region's network assets and intelligence information. This trend deserves continued attention and vigilance from the global cybersecurity community.

3. Suggestions for prevention and investigation

Based on the analysis of three typical attack activities of APT28, we recommend that organizations take the following prevention and troubleshooting measures:

1. Regularly carry out network security education and training to improve employees’ ability to identify and prevent phishing emails, malicious attachments and suspicious links.

2. Establish clear security policies and procedures and require employees to handle emails and attachments from unknown or suspicious sources with caution.

3. Conduct specialized email security training for employees to improve their ability to identify and report suspicious emails.

4. Deploy and update 360 Security Guard on all terminal devices, enable the automatic update function of the operating system and applications, and promptly patch known vulnerabilities.

5. Limit the management rights of ordinary users and reduce the potential impact of malware.

6. Regularly conduct comprehensive security assessments and vulnerability scans of the organization's networks, systems, and applications.

7. Establish a professional security incident response team and equip it with necessary personnel, technology and resources.

8. Implement strict access control and encryption protection for key information assets to minimize potential leak risks.

9. Establish a backup and recovery mechanism for key information assets to ensure timely restoration of business continuity when a security incident occurs.

The above suggestions are designed to help organizations comprehensively improve their network security defense capabilities and resist complex network threats such as APT28. At the same time, we also recommend that organizations flexibly adjust and optimize the above measures based on their own business characteristics and security needs, and continue to invest resources to keep pace with the evolving network security situation.

Reference link

[1] https://cert.gov.ua/article/6276894

[2] https://cert.gov.ua/article/5702579

[3]https://www.justice.gov/opa/pr/justice-department-conducts-court-authorized-disruption-botnet-controlled-russian