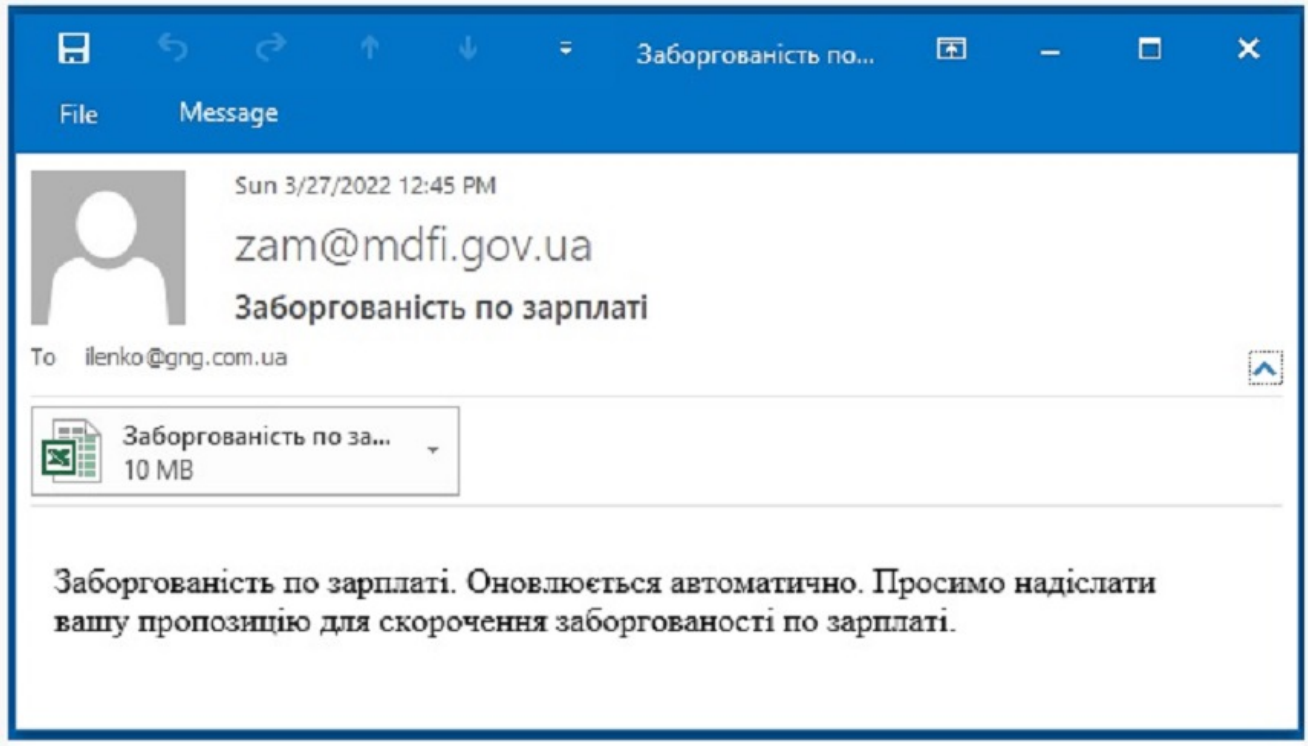
Cyberespionage Actor Deploying Malware Using Excel

govinfosecurity.com/cyber-espionage-actor-deploying-malware-using-excel-a-18830

Cybercrime , Cybercrime as-a-service , Cyberwarfare / Nation-State Attacks

Threat Actors Luring Ukrainian Phishing Targets to Download Malicious Files Prajeet Nair (@prajeetspeaks) • April 2, 2022



Example of email lure used by attackers (Source: Malwarebytes)
Researchers have found that cyberespionage actor UAC-0056, also known as SaintBear, UNC2589 and TA471, is now using a macro-embedded Excel document to target several entities in Ukraine, including ICTV, a private TV channel.

See Also: 2021 Cost Of A Data Breach

"Unlike previous attacks that were trying to convince victims to open a URL and download a first-stage payload or distributing fake translation software, in this campaign the threat actor is using a spear-phishing attack that contains macro-embedded Excel documents," researchers at cybersecurity firm Malwarebytes say.

The UAC-0056 group, which cybersecurity firm SentinelOne recently reported was targeting Ukrainians with fake translation software, is known to have performed a wiper attack in January 2022 on multiple Ukrainian government computers and websites.

In March, Cert-UA reported the group targeting state organizations in Ukraine using malicious implants called GrimPlant, GraphSteel and Cobalt Strike Beacon.

The group is also known to have performed the WhisperGate disruptive attack against the Ukrainian government entities in early 2022.

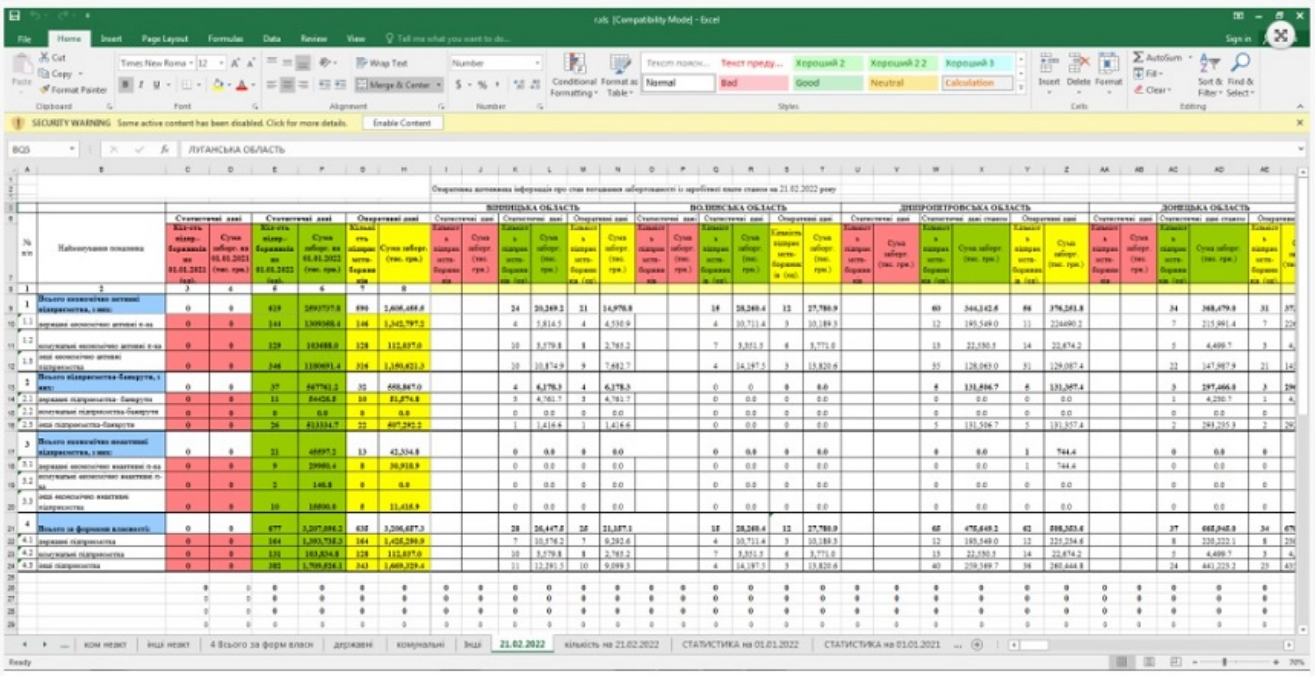
Technical Analysis

The attack starts with a phishing email in which a document attachment containing a malicious macro drops an embedded payload. Then, further payloads are downloaded from the attacker server in Base64 format.

The researchers observed phishing emails being distributed from at least March 23 to March 28, with the subject "wage arrears" and with the body of all the emails containing a similar message: "Wage arrears. Updated automatically. Please send your offer to reduce your salary arrears." The attached document contains a similar message to the email body: "This document contains an embedded macro that drops the first stage payload called 'base-update.exe'. The payload has been saved in a 'very hidden sheet' named 'SheetForAttachedFile,'" the researchers say.

Malwarebytes researchers found that this sheet contains the filename, the date the payload is attached (March 21, 2022), the file size and the content of the attached file in hex format.

"The macro reads the content of the embedded file in the hidden sheet and writes it into the defined location for this payload which is the 'AppDataLocalTemp' directory. The macro used by the actor is taken from a website that described and provided code for a method to attach and extract the files from an Excel workbook," the researchers say.



Example of decoy Excel document (Source: Malwarebytes)

Extracted Files

Elephant Dropper

Researchers say that the Elephant dropper is the initial executable deployed in this attack; it is a simple dropper that deploys further stages. This dropper is written in the Go programming language and is signed with a stolen Microsoft certificate.

"The strings in the binary suggest that it was actually named as Elephant Dropper by the attackers themselves," the researchers say. "It checks if the 'C:Users{user}.java-sdk' directory exists on the system and creates it if it does not. The strings in the binary are encoded and are only decoded when they are required to be used."

The dropper also decodes the command-and-control address from a string and then downloads a Base64 encoded binary from the C2 and writes it to "C:Users{user}.java-sdkjava-sdk.exe."

Elephant Downloader

Elephant Downloader, which is also written in the Go programming language, is executed by the Dropper. The purpose of this payload is to maintain persistence and to deploy the next two stages of the attack.

"The strings in this executable are encoded in the same way as in the Dropper. It makes itself persistent through the auto-run registry key," the researchers say. "The downloader is responsible for getting the implant and the client; the URL paths for the payloads are stored in encoded form in the binary. It downloads the implant and the client."

In the next stage, the Elephant downloader decodes the file names, which are also stored in an encoded format and create a file. The file name of the implant is oracle-java.exe, and the client is microsoft-cortana.exe.

Elephant Implant

Elephant Implant, also tracked as GrimPlant backdoor, seems to be one of the most important payloads in this attack, the researchers say. They describe how it communicates with the C2 on port 80 and gets the C2 address encrypted from its parent process.

"The implant makes use of gRPC to communicate with the C2, it has a TLS certificate embedded in the binary and makes use of SSL/TLS integration in gRPC. This allows the malware to encrypt all the data that is being sent to the C2 via gRPC," the researchers say.

This implant also uses the MachineID library to derive a unique ID for each machine and gets the IP address of the machine by making a request to https://api.ipify.org/.

The implant collects information related to the OS in a function named GetOSInfo. As part of this, the malware collects the hostname, OS name and number of CPUs in the system, and a function named GetUserInfo collects name, username and path to Home directory of the current user.

Elephant Client

The last payload that the researchers detailed is named elephant_client by the actor. It is also tracked as the GraphSteel backdoor. This final payload is a data stealer, the researchers say.

"Similar to other payloads in this attack chain, this payload receives the C2 server as a parameter in Base64 format which is AES encrypted format of the server. Decoding the Base64 string gives the C2 IP address in AES encrypted format. The actor uses a key to AES decrypt (ECB-NoPadding mode) the C2 address," the researchers say.

Upon successful connection with its C2 server, it starts collecting data and exfiltrating it into the server.

Initially, it collects basic information about the users and sends it to the server. The collected data is Base64 encoded and includes hostname, OS name(windows), number of CPUs, IP address, Name, Username and home directory.

Once this is finished, the client tries to steal credentials from the victim's machine. The actor steals data from these services: Browser credentials, Wi-Fi information, Credentials manager data, Mail accounts, Putty connections data and Filezilla credentials.