Linux malware strengthens links between Lazarus and the 3CX supply-chain attack

4/20/2023

Similarities with newly discovered Linux malware used in Operation DreamJob corroborate the theory that the infamous North Korea-aligned group is behind the 3CX supply-chain attack

Similarities with newly discovered Linux malware used in Operation DreamJob corroborate the theory that the infamous North Korea-aligned group is behind the 3CX supply-chain attack

ESET researchers have discovered a new Lazarus Operation DreamJob campaign targeting Linux users. Operation DreamJob is the name for a series of campaigns where the group uses social engineering techniques to compromise its targets, with fake job offers as the lure. In this case, we were able to reconstruct the full chain, from the ZIP file that delivers a fake HSBC job offer as a decoy, up until the final payload: the SimplexTea Linux backdoor distributed through an OpenDrive cloud storage account. To our knowledge, this is the first public mention of this major North Korea-aligned threat actor using Linux malware as part of this operation.

Additionally, this discovery helped us confirm with a high level of confidence that the recent 3CX supply-chain attack was in fact conducted by Lazarus – a link that was suspected from the very beginning and demonstrated by several security researchers since. In this blogpost, we corroborate these findings and provide additional evidence about the connection between Lazarus and the 3CX supply-chain attack.

The 3CX supply-chain attack

3CX is an international VoIP software developer and distributor that provides phone system services to many organizations. According to its website, 3CX has more than 600,000 customers and 12,000,000 users in various sectors including aerospace, healthcare, and hospitality. It provides client software to use its systems via a web browser, mobile app, or a desktop application. Late in March 2023, it was discovered that the desktop application for both Windows and macOS contained malicious code that enabled a group of attackers to download and run arbitrary code on all machines where the application was installed. Rapidly, it was determined that this malicious code was not something that 3CX added themselves, but that 3CX was compromised and that its software was used in a supply-chain attack driven by external threat actors to distribute additional malware to specific 3CX customers.

This cyber-incident has made headlines in recent days. Initially reported on March 29th, 2023 in a Reddit thread by a CrowdStrike engineer, followed by an official report by CrowdStrike, stating with high confidence that LABIRINTH CHOLLIMA, the company's codename for Lazarus, was behind the attack (but omitting any evidence backing up the claim). Because of the seriousness of the incident, multiple security companies started to contribute their summaries of the events, namely Sophos, Check Point, Broadcom, Trend Micro, and more.

Further, the part of the attack affecting systems running macOS was covered in detail in a Twitter thread and a blogpost by Patrick Wardle.

Timeline of events

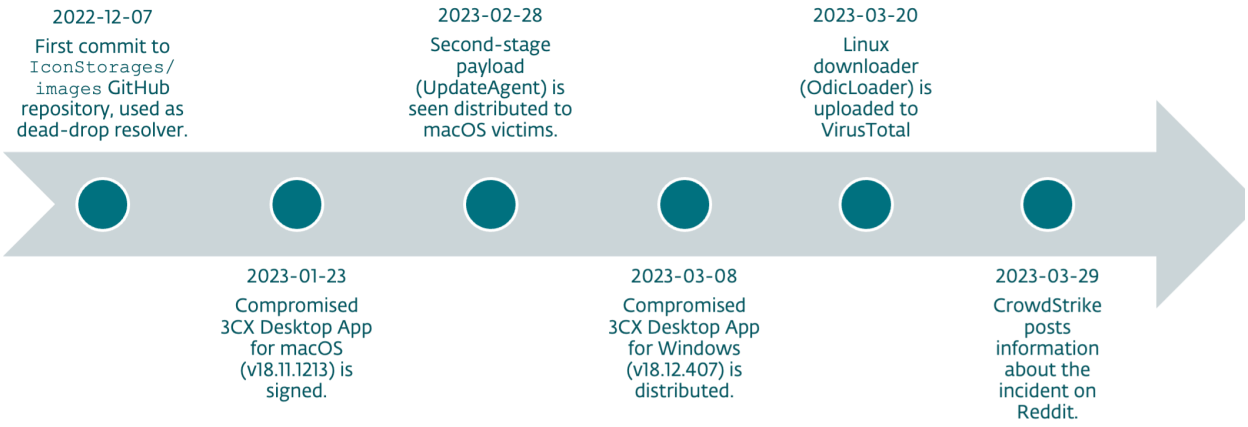

Figure 1. Timeline of events related to the preparation and distribution of 3CX trojanized applications

The timeline shows that the perpetrators had planned the attacks long before execution; as early as December 2022. This suggests they already had a foothold inside 3CX's network late last year.

While the trojanized 3CX macOS application shows it was signed in late January, we did not see the bad application in our telemetry until February 14th, 2023. It is unclear whether the malicious update for macOS was distributed prior to that date.

Although ESET telemetry shows the existence of the macOS second-stage payload as early as February, we did not have the sample itself, nor metadata to tip us off about its maliciousness. We include this information to help defenders determine how far back systems might have been compromised.

Several days before the attack was publicly revealed, a mysterious Linux downloader was submitted to VirusTotal. It downloads a new Lazarus malicious payload for Linux and we explain its relationship to the attack later in the text.

Attribution of the 3CX supply-chain attack to Lazarus

What is already published

There is one domain that plays a significant role in our attribution reasoning: journalide[.]org. It is mentioned in some of the vendor reports linked above, but its presence is never explained. Interestingly, articles by SentinelOne and ObjectiveSee do not mention this domain. Neither does a blogpost by Volexity, which even refrained from providing attribution, stating *"Volexity cannot currently map the disclosed activity to any threat actor"*. Its analysts were among the first to investigate the attack in depth and they created a tool to extract a list of C&C servers from encrypted icons on GitHub. This tool is useful, as the attackers did not embed the C&C servers directly in the intermediate stages, but rather used GitHub as a dead drop resolver. The intermediate stages are downloaders for Windows and macOS that we denote as IconicLoaders, and the payloads they get as IconicStealer and UpdateAgent, respectively.

On March 30th, Joe Desimone, a security researcher from Elastic Security, was among the first to provide, in a Twitter thread, substantial clues that the 3CX-driven compromises are probably linked to Lazarus. He observed that a shellcode stub prepended to the payload from d3dcompiler_47.dll is similar to AppleJeus loader stubs attributed to Lazarus by CISA back in April 2021.

On March 31st it was being reported that 3CX had retained Mandiant to provide incident response services relating to the supply-chain attack.

On April 3rd, Kaspersky, through its telemetry, showed a direct relationship between the 3CX supply-chain victims and the deployment of a backdoor dubbed Gopuram, both involving payloads with a common name, guard64.dll. Kaspersky data shows that Gopuram is connected to Lazarus because it coexisted on victim machines alongside AppleJeus, malware that was already attributed to Lazarus. Both Gopuram and AppleJeus were observed in attacks against a cryptocurrency company.

Then, on April 11th, the CISO of 3CX summarized Mandiant's interim findings in a blogpost. According to that report, two Windows malware samples, a shellcode loader called TAXHAUL and a complex downloader named COLDCAT,

were involved in the compromise of 3CX. No hashes were provided, but Mandiant's YARA rule, named TAXHAUL, also triggers on other samples already on VirusTotal:

- SHA-1: 2ACC6F1D4656978F4D503929B8C804530D7E7CF6 (ualapi.dll),
- SHA-1: DCEF83D8EE080B54DC54759C59F955E73D67AA65 (wlbsctrl.dll)

The filenames, but not MD5s, of these samples coincide with those from Kaspersky's blogpost. However, 3CX explicitly states that COLDCAT differs from Gopuram.

The next section contains a technical description of the new Lazarus malicious Linux payload we recently analyzed, as well as how it helped us strengthen the existing link between Lazarus and the 3CX compromise.

Operation DreamJob with a Linux payload

The Lazarus group's Operation DreamJob involves approaching targets through LinkedIn and tempting them with job offers from industry leaders. The name was coined by ClearSky in a paper published in August 2020. That paper describes a Lazarus cyberespionage campaign targeting defense and aerospace companies. The activity has overlap with what we call Operation In(ter)ception, a series of cyberespionage attacks that have been ongoing since at least September 2019. It targets aerospace, military, and defense companies and uses specific malicious, initially Windows-only, tools. During July and August 2022, we found two instances of Operation In(ter)ception targeting macOS. One malware sample was submitted to VirusTotal from Brazil, and another attack targeted an ESET user in Argentina. A few weeks ago, a native Linux payload was found on VirusTotal with an HSBC-themed PDF lure. This completes Lazarus's ability to target all major desktop operating systems.

On March 20th, a user in the country of Georgia submitted to VirusTotal a ZIP archive called HSBC job offer.pdf.zip. Given other DreamJob campaigns by Lazarus, this payload was probably distributed through spearphishing or direct messages on LinkedIn. The archive contains a single file: a native 64-bit Intel Linux binary written in Go and named HSBC job offer . pdf.

Interestingly, the file extension is not .pdf. This is because the apparent dot character in the filename is a leader dot represented by the U+2024 Unicode character. The use of the leader dot in the filename was probably an attempt to trick the file manager into treating the file as an executable instead of a PDF. This could cause the file to run when double-clicked instead of opening it with a PDF viewer. On execution, a decoy PDF is displayed to the user using xdg-open, which will open the document using the user's preferred PDF viewer (see Figure 3). We decided to call this ELF downloader OdicLoader, as it has a similar role as the IconicLoaders on other platforms and the payload is fetched from OpenDrive.

OdicLoader drops a decoy PDF document, displays it using the system's default PDF viewer (see Figure 2), and then downloads a second-stage backdoor from the OpenDrive cloud service. The downloaded file is stored in ~/.config/guiconfigd (SHA-1: 0CA1723AFE261CD85B05C9EF424FC50290DCE7DF). We call this second-stage backdoor SimplexTea.

As the last step of its execution, the OdicLoader modifies ~/.bash_profile, so SimplexTea is launched with Bash and its output is muted (~/.config/guiconfigd >/dev/null 2>&1).

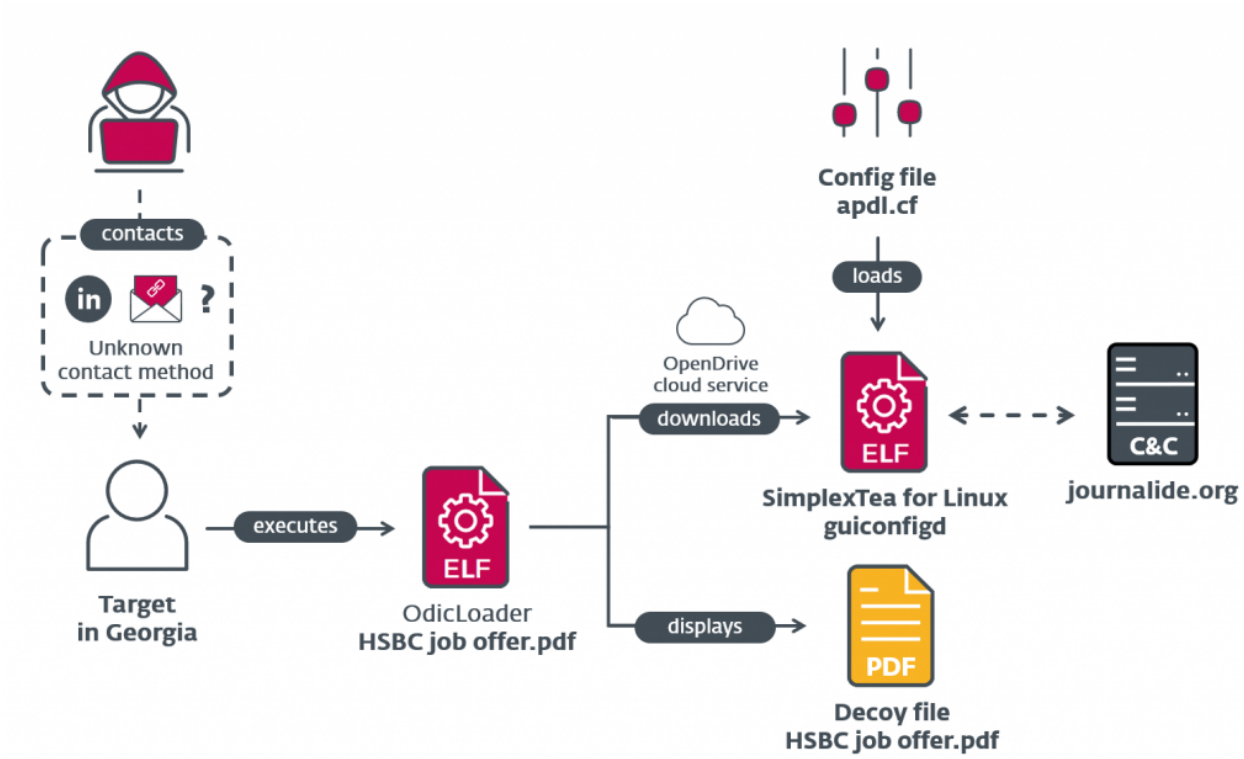

Figure 2. Illustration of the probable chain of compromise

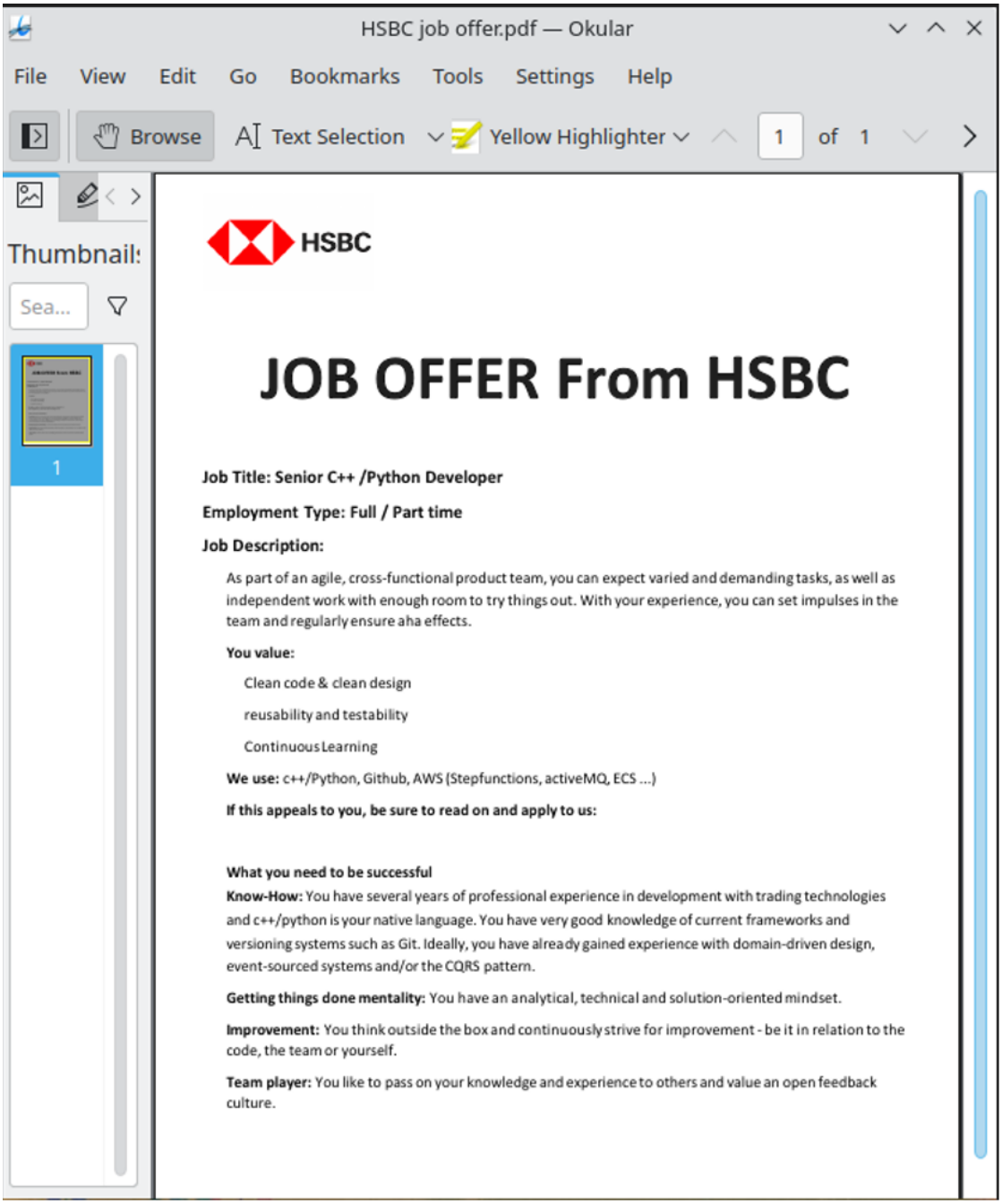

Figure 3. An HSBC-themed lure in the Linux DreamJob campaign

SimplexTea is a Linux backdoor written in C++. As highlighted in Table 1, its class names are very similar to function names found in a sample, with filename sysnetd, submitted to VirusTotal from Romania (SHA-1: F6760FB1F8B019AF2304EA6410001B63A1809F1D). Because of the similarities in class names and function names between SimplexTea and sysnetd, we believe SimplexTea is an updated version, rewritten from C to C++.

Table 1. Comparison of the original symbol names from two Linux backdoors submitted to VirusTotal

guiconfigd (SimplexTea for Linux, from Georgia)	sysnetd (BADCALL for Linux, from Romania)
C**MsgCmd**::Start(void)	**MSG_Cmd**
C**Msg**Secure**Del**::Start(void)	**MSG_Del**
C**MsgDir**::Start(void)	**MSG_Dir**
C**MsgDown**::Start(void)	**MSG_Down**
C**MsgExit**::Start(void)	**MSG_Exit**
C**MsgReadConfig**::Start(void)	**MSG_ReadConfig**
C**MsgRun**::Start(void)	**MSG_Run**
C**MsgSetPath**::Start(void)	**MSG_SetPath**
C**MsgSleep**::Start(void)	**MSG_Sleep**
C**MsgTest**::Start(void)	**MSG_Test**
C**MsgUp**::Start(void)	**MSG_Up**
C**MsgWriteConfig**::Start(void)	**MSG_WriteConfig**

	MSG_GetComInfo
CMsgHibernate::Start(void)	
CMsgKeepCon::Start(void)	
CMsgZipDown::Start(void)	
CMsgZip::StartZip(void *)	
CMsgZip::Start(void)	
CHttpWrapper::RecvData(uchar *&,uint *,uint,signed char)	
	RecvMsg
CHttpWrapper::**SendMsg**(_MSG_STRUCT *)	**SendMsg**
CHttpWrapper::SendData(uchar *,uint,uint)	
CHttpWrapper::SendMsg(uint,uint,uchar *,uint,uint)	
CHttpWrapper::SendLoginData(uchar *,uint,uchar *&,uint *)	

How is sysnetd related to Lazarus? The following section shows similarities with Lazarus's Windows backdoor called BADCALL.

BADCALL for Linux

We attribute sysnetd to Lazarus because of its similarities with the following two files (and we believe that sysnetd is a Linux variant of the group's backdoor for Windows called BADCALL):

- P2P_DLL.dll (SHA-1: 65122E5129FC74D6B5EBAFCC3376ABAE0145BC14), which shows code similarities to sysnetd in the form of domains used as a front for fake TLS connection (see Figure 4). It was attributed to Lazarus by CISA in December 2017. From September 2019, CISA started to call newer versions of this malware BADCALL (SHA-1: D288766FA268BC2534F85FD06A5D52264E646C47).

Figure 4. Similarities between a Windows and a Linux variant of BADCALL (a list of domains used as a front for a fake TLS connection)

- prtspool (SHA-1: 58B0516D28BD7218B1908FB266B8FE7582E22A5F), which shows code similarities to sysnetd (see Figure 5). It was attributed to Lazarus by CISA in February 2021. Note as well that SIMPLESEA, a macOS backdoor found during the 3CX incident response, implements the A5/1 stream cipher.

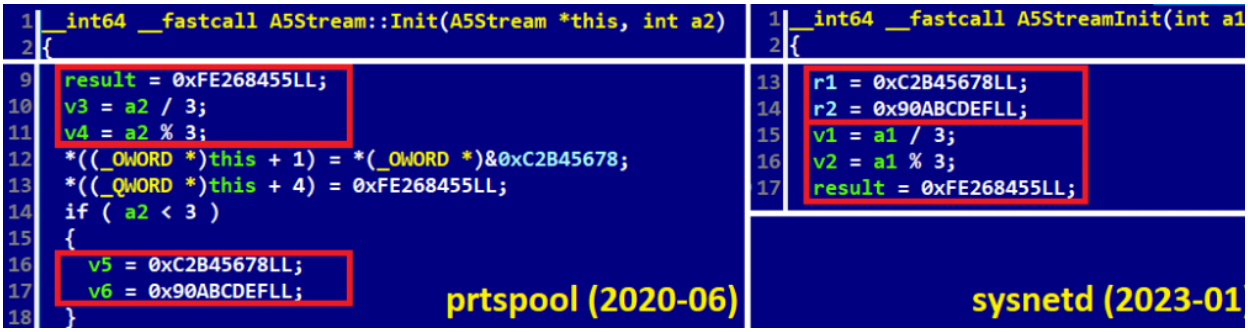

Figure 5. Similarities between AppleJeus for macOS and the Linux variant of BADCALL (the key for the A5/1 stream cipher)

This Linux version of the BADCALL backdoor, sysnetd, loads its configuration from a file named /tmp/vgauthsvclog. Since Lazarus operators have previously disguised their payloads, the use of this name, which is used by the VMware Guest Authentication service, suggests that the targeted system may be a Linux VMware virtual machine. Interestingly, the XOR key in this case is the same as one used in SIMPLESEA from the 3CX investigation.

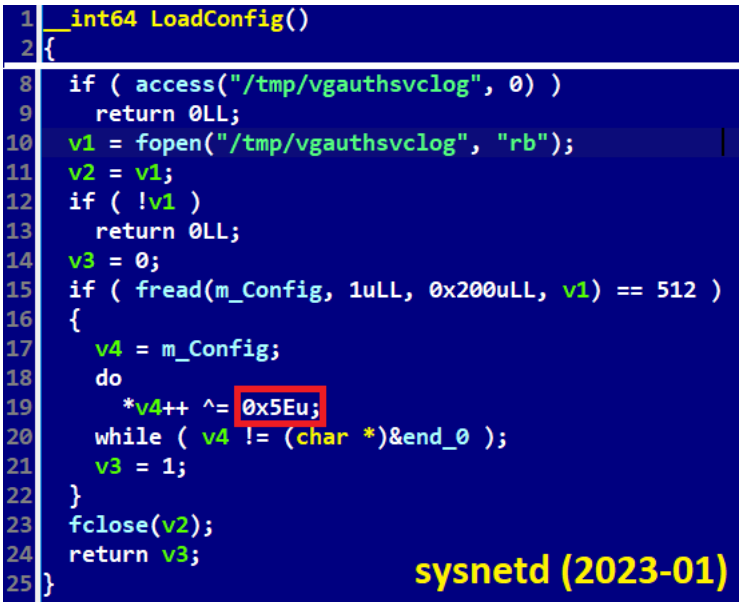

Figure 6. Loading a configuration file by BADCALL for Linux, cf. Figure 8

Taking a look at the three 32-bit integers, 0xC2B45678, 0x90ABCDEF, and 0xFE268455 from Figure 5, which represent a key for a custom implementation of the A5/1 cipher, we realized that the same algorithm and the identical keys were used in Windows malware that dates back to the end of 2014 and was involved in one of the most notorious Lazarus cases: the cybersabotage of Sony Pictures Entertainment (SHA-1: 1C66E67A8531E3FF1C64AE57E6EDFDE7BEF2352D).

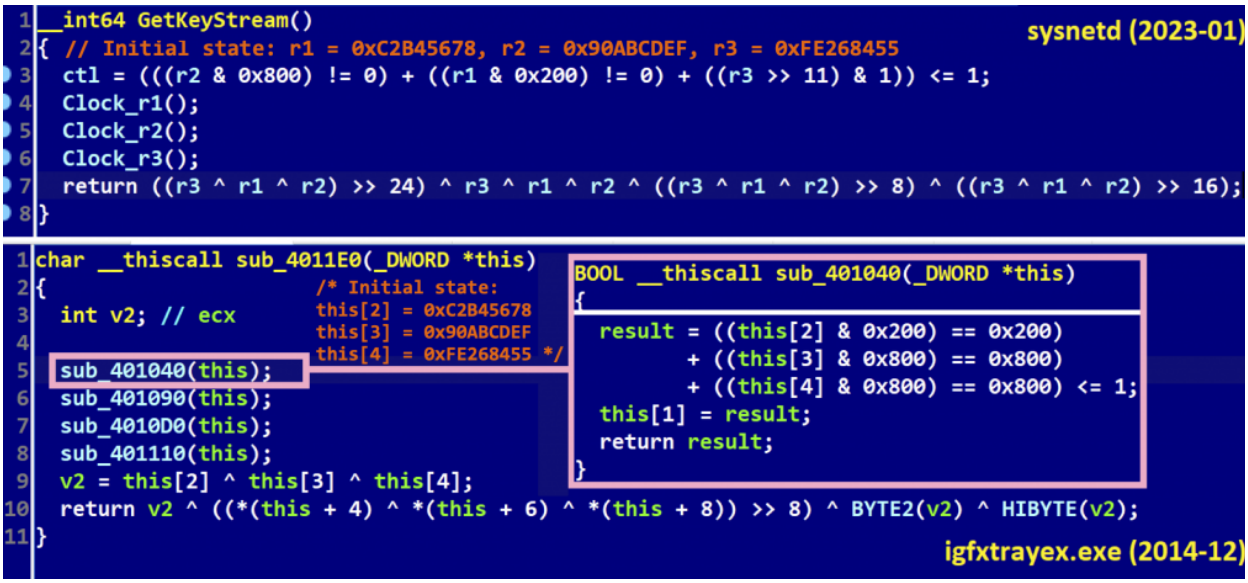

Figure 7. The decryption routine shared between the BADCALL for Linux and targeted destructive malware for Windows from 2014

Additional attribution data points

To recap what we've covered so far, we attribute the 3CX supply-chain attack to the Lazarus group with a high level of confidence. This is based on the following factors:

1. Malware (the intrusion set):
    1. The IconicLoader (samcli.dll) uses the same type of strong encryption – AES-GCM – as SimplexTea (whose attribution to Lazarus was established via the similarity with BALLCALL for Linux); only the keys and initialization vectors differ.
    2. Based on the PE Rich Headers, both IconicLoader (samcli.dll) and IconicStealer (sechost.dll) are projects of a similar size and compiled in the same Visual Studio environment as the executables iertutil.dll (SHA-1: 5B03294B72C0CAA5FB20E7817002C600645EB475) and iertutil.dll (SHA-1: 7491BD61ED15298CE5EE5FFD01C8C82A2CDB40EC) reported in the Lazarus cryptocurrency campaigns by Volexity and Microsoft. We include below the YARA rule RichHeaders_Lazarus_NukeSped_IconicPayloads_3CX_Q12023, which flags all these samples, and no unrelated malicious or clean files, as tested on the current ESET databases and recent VirusTotal submissions.

3. SimplexTea payload loads its configuration in a very similar way to the SIMPLESEA malware from the 3CX official incident response. The XOR key differs (0x5E vs. 0x7E), but the configuration bears the same name: apdl.cf (see Figure 8).

```
__int64 __fastcall CEnv::LoadConfig(CEnv *this)
{
  v1 = getenv("HOME");
  sprintf(v7, "%s/.config/apdl.cf", v1);
  if ( access(v7, 0) )
    return 0;
  v4 = fopen(v7, "rb");
    if ( fread((char *)this + 8, 1uLL, 0x336uLL, v4) == 822 )
    {
      do
      {
        *(_BYTE *)v6 ^= 0x7Eu;
        v6 = (CEnv *)((char *)v6 + 1);
      }
      while ( v6 != (CEnv *)((char *)this + 830) );
      v2 = 1;
    }
    fclose(v5);
  }
```

Figure 8. Loading a configuration file by SimplexTea for Linux, cf. Figure 6

2. Infrastructure:
    1. There is shared network infrastructure with SimplexTea, as it uses https://journalide[.]org/djour.php as it C&C, whose domain is reported in the official results of the incident response of the 3CX compromise by Mandiant.

```
__int64 __fastcall CEnv::SetConfig(CEnv *this)
{
  v2 = 0LL;
  v3 = 0;
  CFunctions::GenerateUID((CFunctions *)&gf, (char *)&v2);
  strcpy((char *)this + 8, (const char *)&v2);
  result = (__int64)this + 40;
  strcpy((char *)this + 40, "https://journalide.org/djour.php");
  *((_DWORD *)this + 6) = 60;
  *((_DWORD *)this + 5) = 0;
  *((_DWORD *)this + 7) = 3;
  *((_DWORD *)this + 8) = 200;
  *((_DWORD *)this + 9) = 0;
  return result;
}
```

Figure 9. A hardcoded URL in SimplexTea for Linux

Conclusion

The 3CX compromise has gained a lot of attention from the security community since its disclosure on March 29th. This compromised software, deployed on various IT infrastructures, which allows the download and execution of any kind of payload, can have devastating impacts. Unfortunately, no software publisher is immune to being compromised and inadvertently distributing trojanized versions of their applications.

The stealthiness of a supply-chain attack makes this method of distributing malware very appealing from an attacker's perspective. Lazarus has already used this technique in the past, targeting South Korean users of WIZVERA VeraPort software in 2020. Similarities with existing malware from the Lazarus toolset and with the group's typical techniques strongly suggest the recent 3CX compromise is the work of Lazarus as well.

It is also interesting to note that Lazarus can produce and use malware for all major desktop operating systems: Windows, macOS, and Linux. Both Windows and macOS systems were targeted during the 3CX incident, with 3CX's VoIP software for both operating systems being trojanized to include malicious code to fetch arbitrary payloads. In the case of 3CX, both Windows and macOS second-stage malware versions exist. This article demonstrates the existence of a Linux backdoor that probably corresponds to the SIMPLESEA macOS malware seen in the 3CX incident. We named this Linux component SimplexTea and showed that it is part of Operation DreamJob, Lazarus's flagship campaign using job offers to lure and compromise unsuspecting victims.

For any inquiries about our research published on WeLiveSecurity, please contact us at threatintel@eset.com.

ESET Research offers private APT intelligence reports and data feeds. For any inquiries about this service, visit the ESET Threat Intelligence page.

IoCs

Files

SHA-1	Filename	ESET detection name	Description
0CA1723AFE261CD85B05C9EF424FC50290DCE7DF	guiconfigd	Linux/NukeSped.E	SimplexTea for Linux.
3A63477A078CE10E53DFB5639E35D74F93CEFA81	HSBC_job_offer.pdf	Linux/NukeSped.E	OdicLoader, a 64-bit downloader for Linux, written in Go.
9D8BADE2030C93D0A010AA57B90915EB7D99EC82	HSBC_job_offer.pdf.zip	Linux/NukeSped.E	A ZIP archive with a Linux payload, from VirusTotal.
F6760FB1F8B019AF2304EA6410001B63A1809F1D	sysnetd	Linux/NukeSped.G	BADCALL for Linux.

Network

IP address	Domain	Hosting provider	First seen	Details
23.254.211[.]230	N/A	Hostwinds LLC.	N/A	C&C server for BADCALL for Linux
38.108.185[.]79 38.108.185[.]115	od[.]lk	Cogent Communications	2023-03-16	Remote OpenDrive storage containing SimplexTea (/d/NTJfMzg4MDE1NzJf/vxmedia)
172.93.201[.]88	journalide[.]org	Nexeon Technologies, Inc.	2023-03-29	C&C server for SimplexTea (/djour.php)

MITRE ATT&CK techniques

Tactic	ID	Name	Description
Reconnaissance	T1593.001	Search Open Websites/Domains: Social Media	Lazarus attackers probably approached a target with a fake HSBC-themed job offer that would fit the target's interest. This has been done mostly via LinkedIn in the past.
Resource Development	T1584.001	Acquire Infrastructure: Domains	Unlike many previous cases of compromised C&Cs used in Operation DreamJob, Lazarus operators registered their own domain for the Linux target.
	T1587.001	Develop Capabilities: Malware	Custom tools from the attack are very likely developed by the attackers.
	T1585.003	Establish Accounts: Cloud Accounts	The attackers hosted the final stage on the cloud service OpenDrive.
	T1608.001	Stage Capabilities: Upload Malware	The attackers hosted the final stage on the cloud service OpenDrive.
Execution	T1204.002	User Execution: Malicious File	OdicLoader masquerades as a PDF file in order to fool the target.
Initial Access	T1566.002	Phishing: Spearphishing Link	The target likely received a link to third-party remote storage with a malicious ZIP archive, which was later submitted to VirusTotal.
Persistence	T1546.004	Event Triggered Execution: Unix Shell Configuration Modification	OdicLoader modifies the victim's Bash profile, so SimplexTea is launched each time Bash is stared and its output is muted.
Defense Evasion	T1134.002	Access Token Manipulation: Create Process with Token	SimplexTea can create a new process, if instructed by its C&C server.
	T1140	Deobfuscate/Decode Files or Information	SimplexTea stores its configuration in an encrypted apdl.cf.

Tactic	ID	Name	Description
	T1027.009	Obfuscated Files or Information: Embedded Payloads	The droppers of all malicious chains contain an embedded data array with an additional stage.
	T1562.003	Impair Defenses: Impair Command History Logging	OdicLoader modifies the victim's Bash profile, so the output and error messages from SimplexTea are muted. SimplexTea executes new processes with the same technique.
	T1070.004	Indicator Removal: File Deletion	SimplexTea has the ability to delete files securely.
	T1497.003	Virtualization/Sandbox Evasion: Time Based Evasion	SimplexTea implements multiple custom sleep delays in its execution.
Discovery	T1083	File and Directory Discovery	SimplexTea can list the directory content together with their names, sizes, and timestamps (mimicking the ls -la command).
	T1071.001	Application Layer Protocol: Web Protocols	SimplexTea can use HTTP and HTTPS for communication with its C&C server, using a statically linked Curl library.
Command and Control	T1573.001	Encrypted Channel: Symmetric Cryptography	SimplexTea encrypts C&C traffic using the AES-GCM algorithm.
	T1132.001	Data Encoding: Standard Encoding	SimplexTea encodes C&C traffic using base64.
	T1090	Proxy	SimplexTea can utilize a proxy for communications.
Exfiltration	T1041	Exfiltration Over C2 Channel	SimplexTea can exfiltrate data as ZIP archives to its C&C server.

Appendix

This YARA rule flags the cluster containing both IconicLoader and IconicStealer, as well as the payloads deployed in the cryptocurrency campaigns from December 2022.

```
/*
The following rule will only work with YARA version >= 3.11.0
*/
import "pe"
rule RichHeaders_Lazarus_NukeSped_IconicPayloads_3CX_Q12023
{
meta:
description = " Rich Headers-based rule covering the IconicLoader and IconicStealer from the 3CX supply chain incident, and also payloads from the cryptocurrency campaigns from 2022-12"
author = "ESET Research"
date = "2023-03-31"
hash = "3B88CDA62CDD918B62EF5AA8C5A73A46F176D18B"
hash = "CAD1120D91B812ACAFEF7175F949DD1B09C6C21A"
hash = "5B03294B72C0CAA5FB20E7817002C600645EB475"
hash = "7491BD61ED15298CE5EE5FFD01C8C82A2CDB40EC"
condition:
pe.rich_signature.toolid(259, 30818) == 9 and
pe.rich_signature.toolid(256, 31329) == 1 and
pe.rich_signature.toolid(261, 30818) >= 30 and pe.rich_signature.toolid(261, 30818) <= 38 and
pe.rich_signature.toolid(261, 29395) >= 134 and pe.rich_signature.toolid(261, 29395) <= 164 and
pe.rich_signature.toolid(257, 29395) >= 6 and pe.rich_signature.toolid(257, 29395) <= 14
}
```