3CX supply chain attack analysis

zscaler.com/security-research/3CX-supply-chain-attack-analysis-march-2023

On March 29th 2023, CrowdStrike published a <u>blog</u> outlining a supply chain attack leveraging the 3CXDesktopApp - a softphone application from 3CX. The ThreatLabz Team immediately started hunting for loCs on the Zscaler Cloud.

We observed infections dating back to **February 2023** for both the Windows as well as the MacOS variant of the Trojanized 3CXDesktopApp installers.



Fig.1 - Infections dating back to February 2023 in Zscaler Cloud

In this case the Threat Actors targeted various industry verticals such as:

- Technology
- Services
- Manufacturing and more

Further let's analyze the Infection Chain for the 3CX Supply Chain Attack:

Infection Chain:

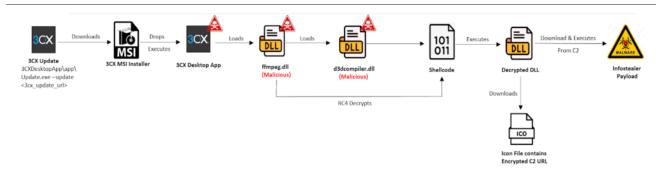


Fig.2 - Infection Chain

The Infection chain begins with the software update routine where the 3CXDesktopApp calls the "Update.exe --update <3cx_update_url>" from its bundle to fetch the updates. This then downloads the valid signed Malicious 3CX MSI installer and the Affected 3CX MAC Application as required in the form of an update package on the victim's machine as shown in the screenshot below.

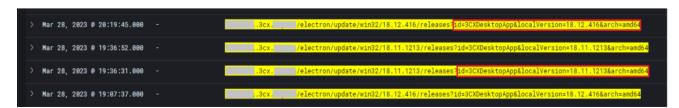


Fig.3 - Requests to 3CX domain to download the Affected 3CX MSI installer v18.12.416 & 3CX Mac App v18.12.416 as an Update Package

In this blog, we will take a look at the affected valid signed 3CX MSI Installer version 18.12.416 named "3CXDesktopApp-18.12.416.msi" which is signed on March 13, 2023.

	🕼 3CXDeskto	pApp-18.	12.416 Pr	operties			×
	Custom		Details		Previous Versions		
	General	Compa	atibility	Digital	Signatures	Security	
15	Signature list						
3CXD-sktopA pp-18.12.416	Name of	signer:	Digest a	lgorithm	Timestamp		
(g) accesses	3CX Ltd		sha1		Monday, Mar	ch 13, 2	
Digital Signature Details	?	💼 Certif	icate				
General Advanced		General	Details	Certificatio	n Path		
Digital Signature Information This digital signature is OK.		Show: Field	<all></all>		Value	~	
Signer information		E Ver	sion		٧3		
Name: 3CX Ltd			ial numbe			1df9c9a4d6ec	c8ed50c9
E-mail: webmaster@3cx.com			nature al nature ba	gorithm ash algorithn	sha25 n sha25		
Webinaster@JCX.com			uer	an agona in		o RSA Code Si	gning CA,
Signing time: Monday, March 13, 2023	12:04:04 PM	📴 Val	id from		Monda	ay, November	2, 2020 5
۰ ۱	View Certifica		id to			, November 3,	
			niect		3011	th 3CX1th 4	Markou D
Countersignatures Serial Numb	oer—	1566	11df9c	9a4d6ec	c8ed50c91	5917873	

Fig.4 - Signed 3CX MSI Installer

Upon execution the 3CX MSI installer extracts multiple files in the "AppData\Local\Programs\3CXDesktopApp" and then executes the valid signed **3CXDesktopApp.exe** as shown below in the screenshot.

)rive.exe	6516	
V 🗠 3CXD	esktopApp.exe	3872	0.20
acx 3C	XDesktopApp.exe	6648	
acx 3C	XDesktopApp.exe	7308	0.01
acx 30	XDesktopApp.exe	1244	
acx 30	XDesktopApp.exe	2528	
	XDesktopApp.exe	4632	
	_/		
CPU Usage: 10	0.32% Physical memor	y: 2.16 GB (43	3.34%) Pro
To use 3CX Desktop App on your computer			
to use sex beaktop App on your computer	/		
 Open the Web Client and login with your 			
credentials.			
Manage app			
Share View Application Tools			
Share View Application Tools			
	sktopApp → app		
	sktopApp → app Date modified	Туре	
> AppData > Local > Programs > 3CXDe		Туре	
AppData > Local > Programs > 3CXDe Name Iocales		Type File fold	er
AppData > Local > Programs > 3CXDe Name Iocales resources			
AppData > Local > Programs > 3CXDe Name Iocales		File fold	er
 AppData > Local > Programs > 3CXDe Name Iocales resources 		File fold File fold	er tion
 AppData > Local > Programs > 3CXDe Name Iocales resources 3CXDesktopApp 		File fold File fold Applicat	er iion cument
 AppData > Local > Programs > 3CXDe Name locales resources 3CXDesktopApp 3CXDesktopApp.VisualElementsManifest chrome_100_percent.pak 		File fold File fold Applicat XML Do	er iion cument
 AppData > Local > Programs > 3CXDe Name Iocales resources 3CXDesktopApp 3CXDesktopApp.VisualElementsManifest chrome_100_percent.pak chrome_200_percent.pak 		File fold File fold Applicat XML Do PAK File PAK File	er tion cument
 AppData > Local > Programs > 3CXDe Name Iocales resources 3CXDesktopApp 3CXDesktopApp.VisualElementsManifest chrome_100_percent.pak 		File fold File fold Applicat XML Do PAK File PAK File Applicat	er iion cument

Fig.5 - Execution of 3CXDesktopApp

Further the 3CXDesktopApp.exe side loads the Backdoored signed DLL named "ffmpeg.dll" as based on the DLL search order mechanism if the DLL is present in the applications directory the DLL is loaded from there as shown in the screenshot.

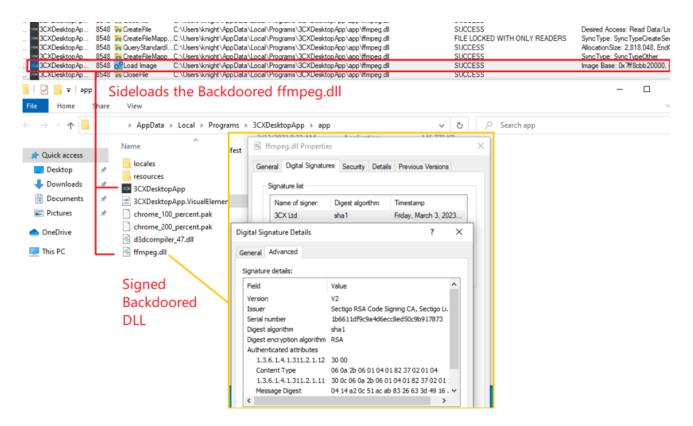


Fig.6 - 3CXDesktopApp sideloads the Backdoored "ffmpeg.dll"

Based on reports, the ffmpeg.dll was backdoored by the Threat Actors via manipulating the source code leading to the Supply Chain Attack. Once loaded into the virtual memory, the malicious "ffmpeg.dll" is commissioned to load the **d3dcompiler_47.dll** which contains the encrypted second stage payload. Initially the main function creates an event called "AVMonitorRefreshEvent" and checks if it already exists. If it does, it exits.



Fig.7 - Main function of ffmpeg.dll

After that it checks the current path in order to load the d3dcompiler_47.dll into memory and further loads the DLL into memory and checks if the DLL loaded correctly by comparing the starting byte of DLL.

 00007FFFDDA36F6C			
COUNTEFEDDAJBERT COUNTEFEDDAJBERT COUNTEFEDDAJBERT COUNTEFEDDAJBERT	,	A Hide FPU	
COUDDYFFFDDAJD0759 48:89C7 imov rdi, rax imov rdi, rax imov rdi, rax imov rcx, rai imov rcx, rax imov rcx, rai imov rc			
 ¹ 00007FFFD0A30F7C ¹ 00007FFFD0A30F7C ¹ 00007FFFD0A30F7C ¹ 00007FFFD0A30F7C ¹ 00007FFFD0A30F7C ¹ 00007FFFD0A30F7C ¹ 00007FFFD0A30F7C ¹ 00007FFFD0A30F7C ¹ 00007FFFD0A30F7C ¹ 00007FFFD0A30F8C ¹ 0F8C300000 ¹ 0F8C3000000 ¹ 0F8C300000 ¹ 0F8C3000000 ¹ 0F8C300000000000000000000000000000000000		RAX 00000000005A4D	
• 00007FFFD0AJ0F7C • 00007FFFD0AJ0F7C • 00007FFFD0AJ0F7C • 00007FFFD0AJ0F8C • 000007FFFD0AJ0F8C • 000007FFFD		RBX 000002A34F717040 "wr "	
<pre></pre>		RCX 00000000000000	
 00007FFFD0A3D87/ 00007FFFD0A3D87/ 89C5		RDX 00007FF758950000 "M7	
COUNTFFEDAJORES SOLT COUNTFFEDAJORES COUNTFFEDAJO		RBP 000000004EDCD8	
CO007FFFD0A3DF88 Set1 mov eck;ex mov eck;ex mov eck;ex mov fbx;rax		RSP 00000062E96EAA0	
COUNTFFFDAJDFS COUNTFFFFDAJDFS COUNTFFFFFDAJDFS COUNTFFFFFDAJDFS COUNTFFFFFFFFFFFFFFFFF COUNTFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF		RSI 00000000000000 RDI 0000000000000 'ü'	
COUNTFFEDDAJDFS COUNTFFEDDAJDFS COUNTFFEDDAJDFA 48:802 COUNTFFEDDAJDFA 48:802 COUNTFFEDDAJDFA 48:802 COUNTFFEDDAJDFA 48:802 COUNTFFEDDAJDFA 48:802 COUNTFFEDDAJDFA 41:835 COUNTFFEDDAJDFA 41:835 COUNTFFEDDAJDFA COU		KD1 000000000000000000000000000000000000	
CO007FFFD0A3DF9C CO007FFFD0A3DF9C CO007FFFD0A3DF9C CO007FFFD0A3DF9C CO007FFFD0A3DF9C CO007FFFD0A3DF4A 48:89F9 CO007FFFD0A3DFAA 48:89F9 CO007FFFD0A3DFAA 48:89F9 CO007FFFD0A3DFAA 48:89F9 CO007FFFD0A3DFAA 41:89F8 CO007FFFD0A3DFAA 41:89F8 CO007FFFD0A3DFAA 41:83F8 CO007FFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFD0A3DFAA CO000FFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO0000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO0000FFFFD0A3DFAA CO0000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO0000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3DFAA CO000FFFFD0A3		R8 0000000000000 'à'	
COUNTFFEDDAJDFAL C:807C24 4C C:807C24 4C C:807C24 4C C:807C24 4C C:807C24 4C C:807C24 4C C:807C4 COUNTFFEDDAJDFAL 48:89F9 mov rdx, rax mov rdx, rdx mov rdx mov rdx, rdx mov rdx mov rdx, rdx mov mov rdx mov rdx		R9 00000062E96E968	
COUNTFFEDDAJDFAI 48:89F9 mov rcx rdx mov rdx rax mov rdx		R10 00007FF80FF0D0C4 ntdl1.0000	7FF80FF0D0c4
COUNTFFEDDAJDEA7 COUNTFFEDDAJDEA7 COUNTFFEDDAJDEA7 COUNTFFEDDAJDEA7 COUNTFFEDDAJDEA7 COUNTFFEDDAJDEA7 COUNTFFEDDAJDEA7 COUNTFFEDDAJDEA COUNTFFEDDAJDEC COUNTFFEDAJDA		R11 00000000000246 L'Z'	
COUNTFEFDOAJDFAA 41:89E8 mov r8d elp mov r8d		R12 00007FFFDDAAF220 <ffmpeg.en< td=""><td>tryPoint></td></ffmpeg.en<>	tryPoint>
COUNTFFEDAJOFAA COUNTFFEDAJOFABC COUNTFF		R13 0000000000000000	
• 00007FFFD0A3DFAD 000000FFFD0A3DFAD 000000FFFD0A3DFAD 000000FFFD0A3DFAD 000000FFFD0A3DFAD 000000FFFD0A3DFAD 0000000FFFD0A3DFAD 0000000FFFD0A3DFAD 0000000FFD0A3DFAD 0000000FFFD0A3DFAD 000000FFFD0A3DFAD 0000000FFFD0A3DFFFD0A3DFAD 0000000FFFD0A3DFAD 000000F		R14 000000000000000 R15 00000062E96EAEC "ØÜN"	
00007FFFD0A3DFB3 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 185 ZE028000 196 Thrpeq. /FFD0A3LFA7 185 ZE028000 196 Thrpeq. /FFD0A3LFA7 196 Thrpeq.		RT2 00000005560EVECRON	
00007FFFD0A3DFB3 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 00007FFFD0A3DFB7 185 ZE028000 196 Thrpeq. /FFD0A3LFA7 185 ZE028000 196 Thrpeq. /FFD0A3LFA7 196 Thrpeq.		RIP 00007FFFDDA3DFC5 ffmpeg.000	07FFFDDA3DFC
00007FFFD0A3DFED 00007FFFD0A3DFED 00007FFFD0A3DFED 00007FFFD0A3DFED 00007FFFD0A3DFED 00007FFFD0A3DFEC 00007FFFD0A3DFEC 00007FFFD0A3DFEC 00007FFFD0A3DFEC 00007FFFD0A3DFEC 000007FFFD0A3DFEC 000000FFFD0A3DFEC 000000FFFD0A3DFEC 000000FFFD0A3DFEC 000000FFFD0A3DFEC 000000FFFD0A3DFEC 0000000FFFD0A3DFEC 000000FFFD0A3DFEC 0000000FFFD0A3DF		Kir 0000/FFF00K50FC5 TTilpeg.000	5/ FFF DDA5DFC
OCCUPYEFEDDA JDECO 30. 405 A0000 Cep eax, 540 Intervention OUDDOFFEDDA JDECO > 0.85 2 C2 0000 Jum of these, FF1DA3 LF7 Jump 15 not taken FM=06 A113 S4DEC5 4403C5 FM=06 A113 S4DEC5 4403C5 Import fmpcg, FF1DA3 LF7 Intervention S6 2 C2 0000 CFF EDDA3 LF7 Itext: 000007FFEDDA3DEC5 ffmpcg, d11: S4DEC5 4403C5 Import fmpcg, FF1DA3 LF7 Itext: 000007FF1DDA3DEC5 ffmpcg, d11: S4DEC5 4403C5 Import fmpcg, FF1DA3 LF7 O00000224F712040 Import fmpcg, FF1DA3 LF7 Itext: 000007FF1DDA3DEC5 ffmpcg, d11: S4DEC5 4403C5 Import fmpcg, FF1DA3DEC5 Ff1DA3DEC5 Import fmpcg, FF1DA3DEC5 Hex 000000224F712040 Import fmpcg, FF1DA3DEC5 Import fmpcg, FF1DA3DEC5 FF1DA3DEC5 Import fmpcg, FF1DA3DEC5 FF1DA3DEC5 Import fmpcg, FF1DA3DEC5 Import fmpcg, FF1DA3DEC5 Impor		RELAGS 00000000000246	
Construction 30 405:A0000 Cemp eax, 540 Cemp eax, 540 Construction Construction Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Construction Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Construction Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Construction Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Dump 1 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Construction Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Jump 1 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Construction Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540 Cemp eax, 540		<	>
Image: Strategy in the			
All C 3 13 Dump 1 Control Contect Contecontect Control Control Contecontect Control Contrecon		Default (x64 fastcall) • 5	Unlocked
ffmpeg.00007FFFDDA3DEC5 ffmpeg.dll:StDec5 #403c5 Itext:00007FFFDDA3DEC5 ffmpeg.dll:StDec5 #403c5 Image: Dump 1 Image: Dump 2 Image: Dump 3 Image: Dump 5		•	
ff#peg.00007FFFDDA3LE7 Ltext:00007FFFDDA3DeC5 ffmpeg.dll:S4DeC5 #4D3C5 ## Dump 1 ## Dump 2 ## Dump 4 ## Dump 5 ## Watch 1 [s=] Local 000000052896EAA0 000000000000000000000000000000000000	>	1: rcx 00000000000000	
ffmpeg.00007FFFDDA3DEC5 ffmpeg.dll:StDec5 #403c5 Itext:00007FFFDDA3DEC5 ffmpeg.dll:StDec5 #403c5 Image: Dump 1 Image: Dump 2 Image: Dump 3 Image: Dump 5		2. ndv_00007cc258950000 "M7 "	
Lext:0007FFFEDDAJDEC5 ffspeg.dll:StDec5 #4000000000000000000000000000000000000		3: r8 0000000000000000	
Bump 1 Bump 2 Bump 3 Bump 4 Bump 5 Witch 1 Ix=1 Locals COCOCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCCCCC/FE/GEAAC COCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCCC/FE/GEAAC COCCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC		4: r9 00000062E96E968	
Bump 1 Bump 2 Bump 3 Bump 4 Bump 5 Witch 1 Ix=1 Locals COCOCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCOCCCC/FE/GEAAC COCCCCC/FE/GEAAC COCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCCC/FE/GEAAC COCCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCCC/FE/GEAAC COCCCCCC/FE/GEAAC COCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC COCCCCCCCFE/GEAAC		5: Ersp+281_000000000000080	
Hex Dump 3 Hex Dump 4 Hex Dump 5 Watch Hex D0000062496EA8 D0000062496EA8 D0000062496EA8 D0000062596EA8 D0000062596EA8 D0000066296EA8 D00000066296EA8 D0000066296EA8 D0000062496EA8 D0000062486EA8		<	>
Address Hex Operating and the second			
000002A34F717050 B 80 00 00 00 00 00 00 00 00 00 00 00 00	<u> </u>		
000002A34F717050 B 80 00 00 00 00 00 00 00 00 00 00 00 00	\\Users\\	\\Desktop\\New folder\\d3dcompiler_4	/.dl1"
000002A34F717050 B 80 00 00 00 00 00 00 00 00 00 00 00 00	N"		
000002A34F717060 10 00 00 00 00 00 00 00 00 00 00 00 00			
000002A34F/17066 00 00 00 00 00 00 00 00 00 00 00 00			
000002A34F717070 00 00 00 00 00 00 00 00 00 00 00 00			
000002A34F717080 VE 1F 8A 0E 100 84 09 CD 21 88 01 4C CD 21 34 85			
000002/A34F/1/090 09 /3 20 /0 /2 0F 0/ /2 0F 0/ /2 01 00 20 03 01 0E 0E 0F 15 program canno 000000062E96EAE8 004EDCD800000000			
000002A34F7170B0 6D 6F 64 65 2F 0D 0D 0A 24 00 00 00 00 00 00 00 00 00 00 00 00 00			

Fig.8 - Load d3dcompiler_47.dll and check for starting byte of DLL

In this case the d3dcompiler_47.dll consisting of the RC4 encrypted shellcode and embedded DLL is valid signed by the Microsoft Digital certificate as shown in the screenshot below.

Digital Signature Details ? ×	G d3dcompiler_47.dll Properties
General Advanced	General Digital Signatures Security Details Previous Versions
Digital Signature Information	Signature list
This digital signature is OK.	Name of sign Digest algorit Timestamp Microsoft Cor sha256 Friday, May 7, 20
Signer information	Certificate
Name: Microsoft Corporation E-mail: Not available	General Details Certification Path
Signing time: Friday, May 7, 2021 6:31:17 PM	Show <all></all>
View Certificate	Field Value ^
Countersignatures	V3 Serial number 33000003dffb6ae3f427ec Signature algorithm sha256RSA
Name of sign E-mail addre Timestamp	Signature hash algorithm sha256
Microsoft Tim Not available Friday, May 7, 20	Issuer Microsoft Code Signing P Valid from Tuesday, December 15, 2
	Valid to Thursday, December 2, 2 Subject Microsoft Corporation, Mi
Details Serial Number —	33000003dffb6ae3f427ecb6a300000003df
	ц ц
ОК	

Fig.9 - Microsoft signed d3dcompiler_47.dll

Further in the infection chain, the ffmpeg.dll looks for the specific hex byte (**FE ED FA CE**) in the loaded d3dcompiler_47.dllwhich contains a second stage encrypted payload.

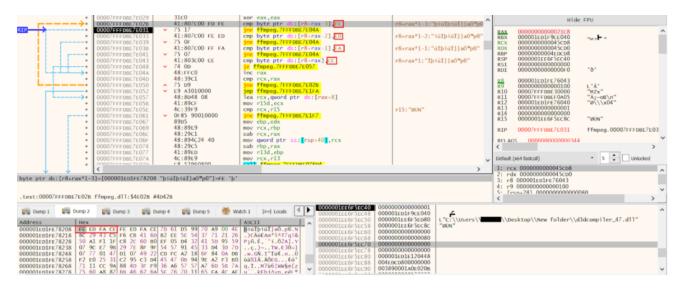


Fig.10 - Look for specific hex byte (FE ED FA CE) in loaded d3dcompiler_47.dll

After it locates the specific hex in loaded d3dcompiler_47.dll, it uses the RC4 decryption with the key "**3jB(2bsG#@c7**" to decrypt the second stage payload which is a shellcode with embedded DLL. The shellcode is responsible for calling the export function "**DIIGetClassObject**" of the second stage DLL to execute and download further stage payload.

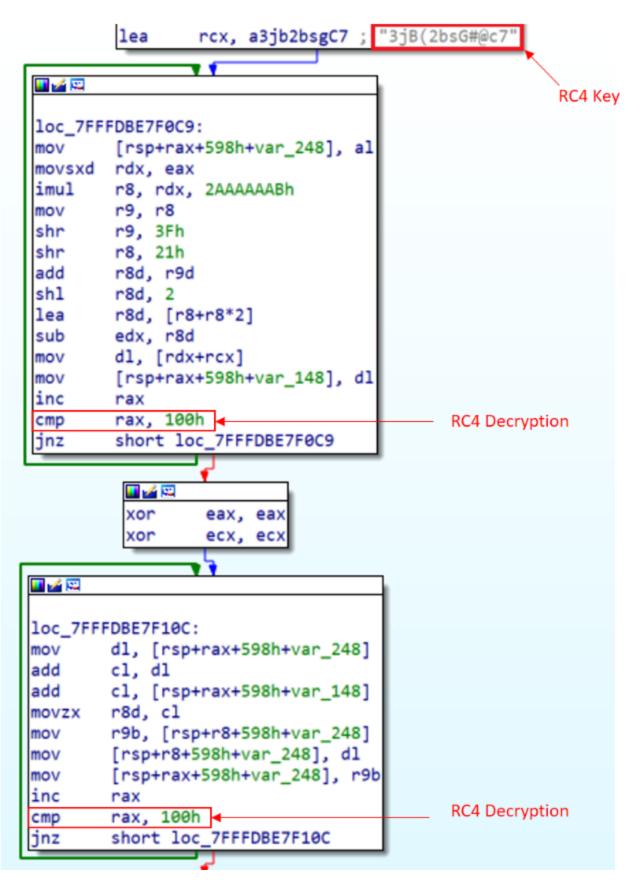


Fig.11 - Decryption of second stage payload

 00007FFFDBE7E1C5 00007FFFDBE7E1C8 	4C:89EA 41:88 40000000	mov rdx,r13 mov r8d,40	40: '@'	^	Hide	P FPU
COUNTERPORTATION COUNTERPORTATION COUNTERPORTATION COUNTERPORTATION	r:15 sc:3z2400 85C0 74 22 4:89r0 resson resson F:15 z78:2800 42:89r0 4:89r1 42:89r1 42:89r1 4:89r1 45:31r6 43:450 4:89r1 45:31r6 48:8508 4:839r3 85:85 74:08 4:839r3 85:856 74:08 4:839r6 43:85r6 74:08	<pre>call quored ptr ds:[#WirtualProtects] test eax,eax is [frace</pre>	edue faziEn edue faziEn edue faziEn		x 0000012303A4040 x 0000012531C481c0 x 0000017531C481c0 x 000001767748590000 p 000000000000000000000000000000000000	"wry ⊨ " "wry ⊨ " "à" "à" "à" "à" "à" "à" "à" "à" "à" "
eax=1				3:	rdv 00007ce7ae890000 "N	u7 E "
.text:00007FFFDBE7E1D4 ffmpeg.d11:\$4E1D4	#4D5D4			4:	r9 00000160B0FEF28	>
Address Hex 0000012b3ClB45CB 33 C0 48 80 5C 24 20 0000012b3ClB45B5 55 C340 5A 90 00 33 C 0000012b3ClB45B5 00 00 88 00 00 00 00 0000012b3ClB45B 00 00 00 00 00 00 0000012b3ClB45B 00 00 00 00 00 00 00 0000012b3ClB45B 00 00 00 00 00 00 00 0000012b3ClB45B 00 00 00 00 00 00 00 0000012b3ClB45B 66 67 42 0 0000012b3ClB46B 00 00 00 00 00 00 00 0000012b3ClB46B 00 00 00 00 00 00 00 0000012b3ClB46B 00 00 00 00 00 00 00 0000012b3ClB46B 00 00 00 00 00 00 00 00 0000012b3ClB46B 00 00 00 00 00 00 00 00 0000012b3ClB46B 00 00 00 00 00 00 00	Dump 4 Image Dump 5 Image V Image V	Ascili Concolic double Ascili Concolic double 0 Ascili 1 Concolic double 1 Concolic double<	0000012/030x494040 0000012/030x494040 0000012/0606F-07/C 00000016/00000000 00000016/00000000 00000016/00000000 00000000000000 00000000000000 00000000000000 000000000000000 00000000000000 00000000000000 00000000000000 0003000000000000000000000000000000000	windows\\Start	Menu\\Programs\\Tools"	

Fig.12 - Decrypted second stage payload

The Stage-2 DLL further downloads the Icon file from the following Github repository as shown below. We observed in some cases that the second stage decrypted DLL would sleep for more than 7 days before communicating with the C2 server.



Fig.13 - Second Stage payload downloads icon files from GitHub Repository

The github repository consists of multiple icon files as shown below. These icons are been downloaded by the Stage-2 DLL.

☐ IconStorages / images			
<> Code Issues Pull requests		✓ Insights	
	🐉 main 👻 1 branch 💿 0 tags		Code -
	lconStorages Add files via upload		e934e60 2 weeks ago 🕥 17 commits
	🗅 README.md	Create README.md	3 months ago
	🗋 icon0.ico	Add files via upload	3 months ago
	🗋 icon1.ico	Add files via upload	3 months ago
	🗋 icon10.ico	Add files via upload	2 weeks ago
	icon11.ico	Add files via upload	2 months ago
	icon12.ico	Add files via upload	2 months ago
	icon13.ico	Add files via upload	2 months ago
	🗋 icon14.ico	Add files via upload	2 months ago
	icon15.ico	Add files via upload	2 months ago
	🗋 icon2.ico	Add files via upload	3 months ago
	🗋 icon3.ico	Add files via upload	3 months ago

Fig.14 - Github Repository hosting multiple icon files.

Further the Stage-2 DLL reads the icon file and parses the encrypted string present at the end of the downloaded icon file and passes it to the ico_decryption() function.

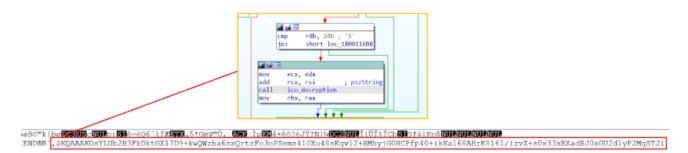


Fig.15 - Parsing of the Encrypted string in the ICON File

The encrypted string from the icon file is base64 decoded and then passed to a decryption routine as shown below in the screenshot. The decrypted string in this case is the C2 URL: https[:]//glcloudservice[.]com/v1/console

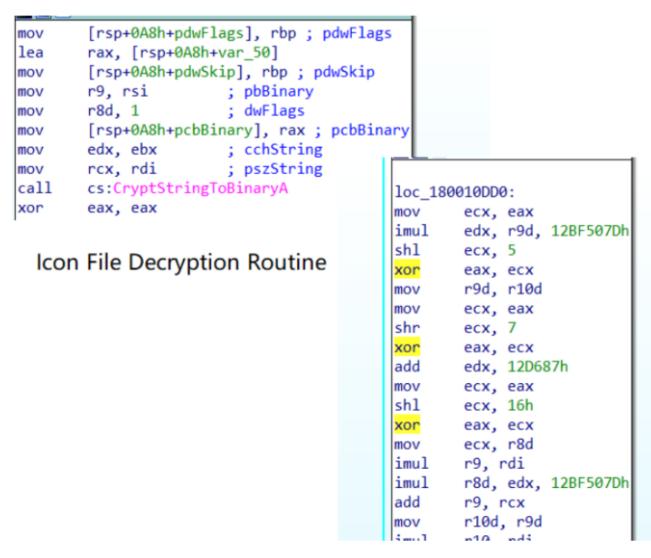


Fig.16 - Decryption of C2 URL from the encrypted string parsed via the ICON File

Further the malware performs HTTPS requests to the C2 URL as shown in the screenshot below from the Zscaler Cloud.



Fig.17 - HTTPS Requests to the C2 URL seen in the Zscaler Cloud

At the time of analysis the C2 Domains were down. The expected response would be in JSON format consisting of encrypted data which is then decrypted by the decryption routine before the final payload is executed on the infected machine.

Based on the <u>blog</u> published by Sentinel One, the final payload delivered on the target machines in the supply chain attack was an Infostealer with capabilities such as collecting system information and browser information such as saved credentials from the Brave,

Affected 3CX Versions:

Following are the affected versions <u>announced</u> by 3CX:

Affected 3CX Electron Windows App Versions:

- 18.12.416
- 18.12.407

Affected Electron Mac App versions:

- 18.11.1213
- 18.12.402
- 18.12.407
- 18.12.416

loCs:

File Name	Md5
3CXDesktopApp-18.12.416.msi	0eeb1c0133eb4d571178b2d9d14ce3e9
3CXDesktopApp.exe	704db9184700481a56e5100fb56496ce
ffmpeg.dll	cb01ff4809638410a531400a66376fa3
d3dcompiler_47.dll	82187ad3f0c6c225e2fba0c867280cc9

C2 Domains:

akamaicontainer[.]com

akamaitechcloudservices[.]com

azuredeploystore[.]com

azureonlinecloud[.]com

azureonlinestorage[.]com

dunamistrd[.]com

glcloudservice[.]com

journalide[.]org

msedgepackageinfo[.]com

msstorageazure[.]com

msstorageboxes[.]com

officeaddons[.]com

officestoragebox[.]com

pbxcloudeservices[.]com

pbxphonenetwork[.]com

pbxsources[.]com

qwepoi123098[.]com

sbmsa[.]wiki

sourceslabs[.]com

visualstudiofactory[.]com

zacharryblogs[.]com

msedgeupdate[.]net