Chinese APT group targets India and Hong Kong using new variant of MgBot malware

blog.malwarebytes.com/threat-analysis/2020/07/chinese-apt-group-targets-india-and-hong-kong-using-new-variant-of-mgbot-malware

Chinese APT group targets India and Hong Kong using new variant of MgBot malware

Threat analysis

Posted: July 21, 2020 by Threat Intelligence Team

This blog post was authored by Hossein Jazi and Jérôme Segura

On July 2, we found an archive file with an embedded document pretending to be from the government of India. This file used template injection to drop a malicious template which loaded a variant of Cobalt Strike.

One day later, the same threat actor changed their template and dropped a loader called MgBot, executing and injecting its final payload through the use of Application Management (AppMgmt) Service on Windows.

On July 5, we observed yet another archive file with an embedded document borrowing a statement about Hong Kong from UK's prime minister Boris Johnson. This document used the same TTPs to drop and execute the same payload.

Considering the ongoing tensions between India and China, as well as the new security laws over Hong Kong, we believe this new campaign is operated by a Chinese state-sponsored actor. Based on our analysis, we believe this may be a Chinese APT group that has been active since at least 2014.

Active targeting with different lures

We were able to track the activities related to these threat actors over the succession of several days based on unique phishing attempts designed to compromise their target.

'Mail security check' with Cobalt Strike (variant 1)

This campaign was most likely carried out through spear phishing emails. The .rar file (*Mail security check.rar*) includes a document with the same name (Figure 1).

Figure 1: Mail security check.docx
The document uses template injection to download a remote template from the following URL (Figure 2).

Figure 2: Template injection
The downloaded template uses the dynamic data exchange (DDE) protocol to execute malicious commands, which are encoded within the document's content (Figure 3).

Figure 3: Encoded command
After decoding, we can see the list of commands that will be executed by DDE:

Figure 4: Decoded commands
As Figure 4 shows, the threat actors used *certutil* with *-urlcache -split -f* parameters to download a *com scriptlet* from its server and then used the *Squiblydoo* technique to execute the downloaded scriptlet via *regsvr32.exe* on the victim machine.

This scriptlet is stored in the *Documents* directory as "ff.sct". The scriptlet is an XML file that has embedded VBscript (Figure 5).

Figure 5: ff.sct snipplet
The scriptlet creates a VB macro and calls Excel to execute it. The macro has been obfuscated to bypass static security mechanism and is responsible for injecting the embedded payload into *rundll32.exe* using the reflective DLL injection method. The injected payload is a variant of Cobalt Strike.

The following diagram shows the overall process of this attack:

Figure 6: Overall process

'Mail security check' with MgBot (variant 2)

As we mentioned earlier, a day after the first attack, the APT group changed its remote template. In this new variant, the actors stopped using the Squiblydoo technique and Cobalt Strike as a payload.

Figure 7 shows the new encoded commands embedded within the template file.

Figure 7: Encoded command
Figure 8 shows the list of commands that will be executed by DDE.

Figure 8: Decoded commands
In this new template file, the *storm.sct* scriptlet was replaced with *storm.txt*. Similar to the previous version, *certutil* is used to download the storm.txt file which is an executable stored in the Documents directory as ff.exe.

The following diagram shows the overall execution process:

Figure 9: Overall execution process

"Boris Johnson Pledges to Admit 3 Million From Hong Kong" with MgBot (variant 3)

The last document used by the Chinese APT group in this campaign focused on issues happening in Hong Kong. The file was embedded within an archive file named "Boris Johnson Pledges to Admit 3 Million From Hong Kong to U.K.rar".

This document quotes the prime minister after a new security law was issued by China against Hong Kong (Figure 10).

Figure 10: Boris Johnson Pledges to Admit 3 Million From Hong Kong to U.K.
Similar to the other documents, it also uses template injection to download the remote template (Figure 11).

Figure 11: Remote template
The downloaded template (BNOHK.docx) is similar to ADIN.docx (variant 2) in which it uses DDE to download and drop its loader.

Payload analysis: MgBot (BLame, Mgmbot)

The dropped executable (ff.exe) is a new variant of a loader called MgBot that drops and loads the final payload. This loader pretends to be a *Realtek Audio Manager tool* (Figure 12).

Figure 12: File version information
It has four embedded resources in which two of them are in Chinese Simplified language. This is an indicator that suggests this campaign is likely operated by a Chinese APT group.

Figure 13: Resource language
The loader starts its process by escalating privilege through a UAC bypass using the CMSTPLUA COM interface.

MgBot uses several anti-analysis and anti-virtualization techniques. The code is self modifying which means it alters its code sections during runtime. This makes static analysis of the sample harder.

MgBot tries to avoid running in known virtualized environment such as *VmWare*, *Sandboxie* and *VirtualBox*. To identify if it's running in one of these environments, it looks for the following DLL files: *vmhgfs.dll*, *sbiedll.dll* and *vboxogl.dll* and if it finds any of these DLLs, it goes to an infinite loop without doing any malicious activity (Figure 14).

Figure 14: Anti-VMs
It also checks for the presence of security products on the victim's machine and takes a different execution flow if a security product is detected. For example, it checks for *zhudongfangyu.exe, 360sd.exe, 360Tray.exe, MfeAVSvc.exe and McUICnt.exe* in different parts of the code (Figure 15). The malware does not perform all the checks at once and it rather checks a couple of them at different steps of its execution.

Figure 15: Security products checks

To invoke the required APIs, the malware does not call them directly but instead builds a function pointer table for the required APIs. Each request to an API call is made through the access to the relevant index of this table.

Figure 16: Building function pointer table
As an example, when the malware needs to invoke *WinExec*, it does so by invoking it through its index from the function pointer table.

Figure 17: Calling API through use of function pointer table
After building the required API calls table, the malware performs the following procedures:

- It calls *CreateFileW* to create *iot7D6E.tmp* (random name starting with iot) into the *%APPDATA%Temp* directory. This tmp file is a cab file that embedds the final payload.
- It calls *WriteFile* to populate its content
- It calls *CreateProcessInternalW* to invoke *expand.exe* to decompress the content of *iot7D6E.tmp* into *ProgramData\Microsoft\PlayReady\MSIBACF.tmp\tmp.dat* (the *MSIBACF.tmp* directory name is generated randomly and starts with MSI and then is followed by a combination of random numbers and characters)

Figure 18: Calling expand.exe
- It calls *CopyFileW* to copy tmp.dat into *pMsrvd.dll*
- It calls *DeleteFileW* to delete *tmp.dat*
- It drops *DBEngin.EXE* and *WUAUCTL.EXE* in the *ProgramData\Microsoft\PlayReady* directory. Both of these files are *rundll32.exe* that is used later to execute the dropped DLL.
- It modifies the registry hive of of *HKLM\SYSTEM\CurrentControlSet\Services\AppMgmt* registry location to make itself persistent. To perform this modification, it drops two registry files named iix*.tmp (random numbers have been added to iix) into the %APPDATA%Temp directory which are the old and new registry hives for the mentioned registry location.

To load the dropped DLL (*pMsrvd.dll*) the loader registers it as a service. To achieve this, it makes use of the already installed service, AppMgmt, to load the payload as shown in the following images:

Figure 18: ServiceDll
Figure 19: ImagePath
Finally, it executes the dropped DLL by running *net start AppMgmt*. After loading the DLL, the Loader creates a cmd file (*lgt*.tmp*.cmd) in the *%APPDATA%TEMP* directory with the content shown in Figure 20. Then it executes it to delete the cmd file and loader from the victim's machine.

Figure 20: cmd file

We were able to identify several different variants of this loader. In general, all the variants drop the final payload using *expand.exe* or *extrac32.exe* and then use "net start *AppMgmt*" or "net start StiSvc" to execute the dropped DLL with one of the following configurations:

- svchost.exe -k netsvcs -p -s AppMgmt
- svchost.exe -k netsvcs
- svchost.exe -k imgsvc

The dropped DLL is the main payload used by this threat actor to perform malicious activities. The following shows the file version information pretending to be a *Video Team Desktop App.*

Figure 21: File info
The creation time for this DLL appears to be "2008-04-26 16:41:12". However, based on Rich header data, we can assert that this might have been tampered with by the threat actor.

Figure 22: Rich header
This DLL has exported several functions as shown in the following Figure. The function *ServiceMain* is used by this DLL to install itself as a service.

Figure 23: Export functions
It can check the running services and based on that can run itself as a service using *WmiPrvSE.exe*.

Figure 24: Running as service using WmiPrvse.exe
Figure 25: RAT's DLL is installed as service using WmiPrvse.exe
It uses several anti-debugging and anti-virtualization techniques to detect if it's running in a virtualized environment or if it is being debugged by a debugger. It uses *GetTickCount* and *QueryPerformanceCounter* API calls to detect the debugger environment.

To detect if it is running in a virtual environment, it uses anti-vm detection instructions such as *sldt* and *cpid* that can provide information about the processor and also checks Vmware IO ports (VMXH).

Figure 26: Environment Detection
All the strings and used imports by this RAT are obfuscated using DES algorithm. Three different functions have been used to deobfuscate strings and imports:

- The first one has been used to decrypt unicode strings: In this function each string has a limit of 160 bytes and the key for decryption is: 25 00 57 50 8B C3 8A
- The second function is used to decrypt ASCII strings: In this function each string has a limit of 56 bytes and the decryption key is: A2 B1 1B B1 26 7B
- The last function is used to resolve API functions names in ASCII format. The decryption key is this case is 81 6E 00 E5 50 B2 D8.

This final piece of code bundled in MgBot is a Remote Administration Trojan with several capabilities such as:

- C2 communication over TCP (42.99.116[.]225:12800)
- Ability to take screenshots
- File and directory management (Enumerate files and directories and send the list to the server, Rename files, Create directories, Delete files)
- Process management
- Get drive type (FAT, FAT32, NTFS, CDFS) and free space

Infrastructure relations

The following shows the infrastructure used by this APT and relations between hosts used by this group. This APT group has used several different IP addresses to host its malicious payloads and also for its C2 communications.

What is interesting is that the majority of IP addresses used by this APT are located in Hong Kong and almost all of these Hong Kong-based IP addresses are used for C2 communication. Even in their past campaigns they mostly have used infrastructure in Hong Kong. The graph also shows the relationship between different IP addresses used by this APT group.

Figure 27: Infrastructure connections

Android RAT

We also found several malicious Android applications we believe are part of the toolset used by this APT group. Malwarebytes detects them as *Android/Trojan.Spy.AndroRat.KSRemote*.

Figure 28: Malicious Android APK
All these bogus applications contain a jar file named *ksremote.jar* that provides the RAT functionality:

- Recording screen and audio using the phone'ss camera/mic
- Locating phone with coordinates
- Stealing phone contacts, call log, SMS, web history
- Sending SMS messages

Figure 29: Contact grabbing capability
This RAT communicates with C&C servers using random port numbers within the 122.10.89.170 to 179 range (all in Hong Kong)

- 122.10.89[.]172:10560
- 122.10.89[.]170:9552
- 122.10.89[.]172:10560

TTPs in line with Chinese APTs

The lures used in this campaign indicate that the threat actor may be targeting the Indian government and individuals in Hong Kong, or at least those who are against the new security law issued by China.

The TTPs observed in these attacks have been used by several Chinese APT groups:

- Rancor APT is known to use Certutil to download their payload
- KeyBoy is known to have used DDE is its previous campaigns
- APT40 has utilized Squiblydoo and template injection in its previous campaigns.

Considering these factors we attribute this APT attack with moderate confidence to a new Chinese APT group. Based on the TTPs used by this APT group we were able to track back its activities to at least 2014. In all their campaigns the actor has used a variant of MgBot.

A threat actor with a long documented history

A Needle in a haystack blog post from 2014 detailed a campaign that drops a Trojan disguised as a legitimate MP3 encoder library. In this campaign the actor used CVE-2012-0158 to drop its Trojan. The rest of the TTPs including the methods used by the threat actor to execute MgBot and registry modifications are similar to this ongoing campaign.

In 2018, this group performed another operation in which they used a VBScript vulnerability (CVE-2018-8174) to initiate their attack to drop a variants of MgBot. In March 2020, an archive file (warning.rar) was submitted to VirusTotal that we believe is part of another campaign used by this actor.

We will continue this group's activities to see if their targeting or techniques evolve. Malwarebytes users are protected from this campaign thanks to our signature-less anti-exploit layer.

Figure 30: Malwarebytes Nebula blocking malicious Word document

MITRE ATT&CK techniques

Tactic	ID	Name	Details
Execution	T1059	Command-Line Interface	Starts CMD.EXE for commands execution
	T1106	Execution through Module Load	Loads dropped or rewritten executable - WUAUCTL.EXE - svchost.exe - rundll32.exe

	T1053	Rundll32	Uses RUNDLL32.EXE to load library
	T1064	Scripting	WScript.exe: Starts MSHTA.EXE for opening HTA or HTMLS files
	T1035	service execution	Starts NET.EXE for service management
	T1170	mshta	Starts MSHTA.EXE for opening HTA or HTMLS files
	T1086	PowerShell	Executes PowerShell scripts
Privilege Escalation	T1050	new service	Creates or modifies windows services through rundll32.exe
	T1088	Bypass UAC	Known privilege escalation attack through DllHost.exe
Persistence	T1031	Modify Existing Service	Creates or modifies windows services through rundll32.exe
	T1050	new services	Creates or modifies windows services through rundll32.exe
Defense Evasion	T1107	File Deletion	Starts CMD.EXE for self-deleting
	T1085	Rundll32	Uses RUNDLL32.EXE to load library
	T1088	bypass UAC	Known privilege escalation attack through DllHost.exe
	T1497	Virtualization/Sandbox Evasion	The Loader uses several anti-virtualization detections techniques
	T1221	Template Injection	Maldoc uses template injection to download remote template
	T1218	Signed Binary Proxy Execution	Use Squiblydoo to load executable
Discovery	T1012	Query Registry	Reads the machine GUID from the registry
	T1082	System Information Discovery	Reads the machine GUID from the registry
	T1007	System Service Discovery	Starts NET.EXE for service management

Lateral Movement	T1105	Remote File Copy	- certutil.exe: Downloads executable files from the Internet - cmd.exe: Starts CertUtil for downloading files
C&C	T1105	Remote File Copy	- certutil.exe: Downloads executable files from the Internet - cmd.exe: Starts CertUtil for downloading files

Table 1: Mitre Attack TTPs

IOCs

2a5890aca37a83ca02c78f00f8056e20d9b73f0532007b270dbf99d5ade59e2a Boris Johnson Pledges to Admit 3 Million From Hong Kong to U.K.docx

fc885b50892fe0c27f797ba6670012cd3bbd5dc66f0eb8fdd1b5fca9f1ea98cc BNOHK.docx.zip

3b93bc1e0c73c70bc8f314f2f11a91cf5912dab4c3d34b185bd3f5e7dd0c0790 Boris_Johnson_Pledges_to_Admit_3_Million_From_Hong_Kong_to_U.K.rar

ecf63a9430a95c34f85c4a261691d23f5ac7993f9ac64b0a652110659995fc03 Email security check.rar

1e9c91e4125c60e5cc5c4c6ef8cbb94d7313e20b830a1e380d5d84b8592a7bb6 Email security check.docx

3a04c1bdce61d76ff1a4e1fd0c13da1975b04a6a08c27afdd5ce5c601d99a45b ADIN.docx (storm.sct)

855af291da8120a48b374708ef38393e7c944a8393880ef51352ce44e9648fd8 ADIN.docx (storm.sct)

1e81fb62cb57a3231642f66fee3e10d28a7c81637e4d6a03515f5b95654da585 ff.exe (storm.txt)

99aee7ae27476f057ef3131bb371a276f77a526bb1419bfab79a5fac0582b76a cobalt strike

flash.governmentmm.com: This domain used by actor to host remote templates. It has been registered 3 month ago by someone in United States.

MgBot samples

2310f3d779acdb4881b5014f4e57dd65b4d6638fd011ac73e90df729b58ae1e0
e224d730e66931069d6760f2cac97ab0f62d1ed4ddec8b58783237d3dcd59468
5b0c93a70032d80c1f5f61e586edde6360ad07b697021a83ed75481385f9f51f
1e81fb62cb57a3231642f66fee3e10d28a7c81637e4d6a03515f5b95654da585
07bb016c3fde6b777be4b43f293cacde2d3aae0d4e4caa15e7c66835e506964f
7bdfabdf9a96b3d941f90ec124836084827f6ef06fadf0dce1ae35c2361f1ac6
8ab344a1901d8129d99681ce33a76f7c64fd95c314ac7459c4b1527c3d968bb4
f41bfc57c2681d94bf102f39d4af022beddafb4d49a49d7d7c1901d14eb698d2

45.77.245[.]0: This IP has been used by Cobalt Strike as a C&C server.

42.99.116[.]225: C&C server used by final Payload.

Android samples

b5304a0836baf1db8909128028793d12bd418ff78c69dc6f9d014cadede28b77
9aade1f7a1f067688d5da9e9991d3a66799065ffe82fca7bb679a71d89fec846
5f7f87db34340ec83314313ec40333aebe6381ef00b69d032570749d4cedee46

COMMENTS

RELATED ARTICLES

ABOUT THE AUTHOR

Threat Intelligence Team