G DATA
Whitepaper 2018 – paper 3

Analysis of

MSIL.Backdoor.SocketPlayer.A

Analysis by: https://twitter.com/RansomBleed

G DATA Software AG | Malware analysis | May 2018

Contents

1. Introduction

The twitter user @struppigel brought my attention to a post of @sudosev who found a RAT[1] on virustotal which looks like a new malware family. The RAT is written in .NET and the source-code can therefore be inspected easily. The RAT uses socket.io for communication which is not that typical for malware hence it captured my attention.

2. Initial routine of SocketPlayer

Firstly, the actual socket connection is created with the code line *Socket.socket = IO.Socket(URL).*

After that, the program checks whether the directory *C:\Users\USERNAME\Music\Player* exists. If it doesn't, it will create that folder. Next it creates a registry key to the *SOFTWARE\Microsoft\Windows\CurrentVersion\Run* path. Proceeding the autostart entry, the program copies itself to the location *C:\Users\USERNAME\Music\Player\Player.exe*

Lastly the actual socket connection is made with the *socket.Emit* function which passes the following information to the c2: the username, the current date, the antivirus product name (information about how this is achieved later) and the machine name. Some information like the username is sent twice and the machine name is even sent thrice – no idea why the author did that.

3. Commands

Command	Functionality
fdrive	Iterates through all drives that are ready and returns the name, the total size and again, the name. If a drive isn't ready, only the name is returned.
fdir	If the specified directory isn't on the system, false is returned. Otherwise the path of the subdirectories, the creation date of those, all files within the current directory, the file size and the creation date of those is returned.
mfdir	Returns the filename, file size, creation time and the full name of all the extensions that were specified recursively using the *sndflesi* method.
f1	If the specified parameter is longer than 3 and contains ":\\" it will return "f1|drive". Otherwise it will return the name of the parent directory.
strtsgnl	Returns the uid of the current running program.

fdowl	Using the *sndfle* function, a file is uploaded to the c2 using the path /cl/upld/.
fexc	Executes a file if it exists.
fdel	Deletes a directory or file if it exists.
procs	Returns information about currently running processes like the process name, whether the process responds, the window title and the process id.
prockil	Kills a running process by id. After that the same information as in *procs* is returned(to check if the process is terminated).
Gtscreen	A screenshot is made and returned.
upld	A specified file from the /uploads/ folder of the c2 is downloaded and stored to the temporary directory. After that it's executed.
upldex	Similar to *upld* but with specified location in the temporary directory and an autostart entry for it as well.
kylgs	Reads the contents of the file "klsetup.txt" in the temporary directory and returns them.
destt	Kills the active socket connection, removes the autostart key, removes the created file and the path from the initial routine and exits itself.

The command *kylgs* is possibly intended to serve as a keylogging feature as the name and the partial functionality suggests it. However, no actual keylogging functionality has been implemented and therefore the command is useless for now.

4. AvName() function

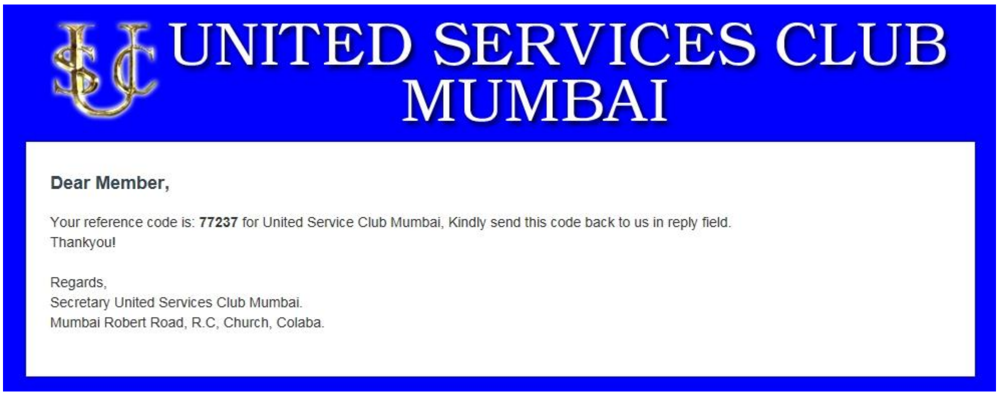
The function *AvName()* uses the *ManagementObjectSearcher* class in order to enumerate the currently installed antivirus product with the SQL query *"SELECT * FROM AntivirusProduct"*. As multiple values are returned, the program targets the property value *displayName* as this value contains the name of the antivirus product. The *displayName* value is returned. This function is used in 2. Initial routine.

5. SocketPlayer downloader

At the current time of writing there are two variants of the SocketPlayer downloader out there. The first variant is a ~100KB file which does exactly what a typical downloader does - downloading a file and executing it. The second one is a little bit more complex and is described in 7. variant 2.

6. Variant 1

According to virustotal version 1a[2] of the first variant was submitted on 2018-03-28. The variant 1b[3], which was submitted on 2018-03-31, is 5.5KB bigger. This is because the variant 1b contains HTML file within its resources. Below is a screenshot of the displayed file.



UNITED SERVICES CLUB
MUMBAI

Dear Member,

Your reference code is: **77237** for United Service Club Mumbai, Kindly send this code back to us in reply field.
Thankyou!

Regards,
Secretary United Services Club Mumbai.
Mumbai Robert Road, R.C, Church, Colaba.

Figure 2. HTML file from variant 1b[3] resources.

This image indicates that the author is planning to launch a phishing attack to club members. However, no email functionality is used in the sample and the HTML file makes therefore not much sense for now. The submission dates on virustotal indicate that the version 1a is an older and version 1b is the newer version of the downloader. The addition of the HTML file reinforces this assumption.

7. Variant 2

Same as variant 1, there is also an old version[4] and a new version[5].

Both versions have a similar initial routine as in 2. Initial routine. The old version only uses the *C:\Users\USERNAME\Music* path and downloads the data from *hxxp://173.249.39.7:1337/uploads/excutbls/* with the filename specified via socket.io from the server.

The new version checks if the file *hxxp://93.104.208.17:1337/uploads/excutbls/h/one/Player.exe* exists and then opens a socket connection. If the directory *C:\Users\USERNAME\Music\Audio Player* doesn't exist, it will be created. If the file *C:\Users\USERNAME\Music\Audio Player\Audio.exe* doesn't exist, an autostart key with the value "Audio Player" is created and the file copies itself to that path.

After that it does the same routine again for a different path. It checks if the directory *C:\Users\USERNAME\Music\Music Player* exists. If it doesn't, the directory will be created. If the file

C:\Users\USERNAME\Music\Music Player\Music.exe doesn't exist, it will create an autostart key with the value "Music Player". Next it will download the file *hxxp://93.104.208.17:1337/uploads/executbls/h/two/Music.exe* and save it to the path *C:\Users\USERNAME\Music\Music Player\Music.exe.*

Lastly the machine name is sent twice, the username is also sent twice and the date is sent to the c2.

8. Procedure of the latest SocketPlayer infection routine

The downloader[6] creates a connection to the host *hxxp://asdkajkjsdnaskjndjansdka.com*. This connection is made to detect sandboxed systems. Most sandboxed systems return true for any DNS query. So if this connection "succeeds" the malware stops itself. If the check has passed, the downloader downloads *hxxp://93.104.208.17:5156/uploads/executbls/h/Audio.exe,* decrypts that executable with the hardcoded string "*!##&aAwPs1337*" and uses the *Invoke* method to run the decrypted program[7] in memory.

The invoked program creates a socket connection to the host *hxxp://93.104.208.17:5156/socket.io* and creates the registry key *HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run* "Audio Player" with the command *C:\Users\USERNAME\Music\Audio.exe* and copies itself to the commands location. If the file *C:\Users\USERNAME\AppData\Roaming\Process Handler\Handler.exe* exists, it checks if the directory *C:\Users\USERNAME\AppData\Roaming\Process Handler* exists. If the directory doesn't exist, it will be created. Furthermore, an autostart key with the value "Handler" to the above executable path is created. It then downloads the file *hxxp://93.104.208.17:5156/uploads/excutbls/h/Bkdr.exe* and saves it to the location above.

It also downloads and executes the file *http://93.104.208.17:5156/uploads/excutbls/h/Cntrl.exe.* The Cntrl.exe[8] downloads Player.exe(SocketPlayer)[9], decrypts it[10] and runs the file in memory.

9. Bkdr.exe

The file Bkdr.exe[11] (possibly means backdoor.exe) just downloads the file hxxp://93.104.208.17:5156/uploads/excutbls/h/MAudio.exe[12], saves the file to the temporary directory and executes it. The executed file downloads the file hxxp://93.104.208.17:5156/uploads/excutbls/h/Audio.exe(Unfortunately no hash available at this point), decrypts it with the hardcoded key "!##&aAwPs1337" and invokes it in memory. The invoked file is similar to variant 2 of the SocketPlayer downloader.

10. Changes to SocketPlayer

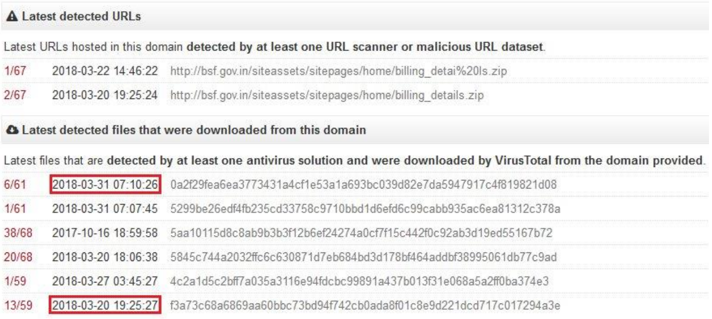
At the beginning of this report we had looked at the commands of the file[1]. With the latest sample[10] some things changed which are listed below.

- The c2 port has changed from 3000 to 7218
- The file location changed from C:\Users\USERNAME\Music\Player\Player.exe to C:\Users\USERNAME\Music\Media Player\Player.exe

- The information that is sent in the initial routine changed a bit. In the old version[1] the author sends the string ",1.1,1" to the c2. In the new version the program sends ",1.2,1", telling the c2 that the new version runs on the machine.
- To the commands Fdrive,fdir,smfdir,procs,prockil,gtscreen and kylgs the variable *susrid* is added to be sent to the server. This is done to identify the infected systems better.
- The functionality *stscrnpercnt* is added. This feature assists the gtscreen function to set the quality of the image.
- The gtscreen function additionally to the *susrid* also sends the computer name with the picture.
- The storage location of *upldex* is changed to *C:\Users\User\AppData\Roaming\Microsoft\Windows\Templates*. An autostart key to the registry is added. The downloaded file is also executed.
- The *kylgs* function also switched to use the above path to read the file *klsetup.txt*.
- The *destt* function additionally checks if the following path and files are available. If so, it deletes them. *C:\Users\USERNAME\AppData\Roaming\Process Handler C:\Users\USERNAME\AppData\Roaming\Process Handler\Handler.exe,* C:\Users\User\AppData\Roaming\Microsoft\Windows\Templates\Image.exe *and* C:\Users\User\AppData\Roaming\Microsoft\Windows\Templates\Media.exe.

11. Spreading

It appears that the website *https://bsf.[gov].in* which is the border security force of India has been used to spread malware including SocketPlayer downloader as you can see from figure 2 Below. Currently the website does not seem to spread malware again.



⚠ Latest detected URLs

Latest URLs hosted in this domain **detected by at least one URL scanner or malicious URL dataset**.

1/67	2018-03-22 14:46:22	http://bsf.gov.in/siteassets/sitepages/home/billing_detai%20ls.zip
2/67	2018-03-20 19:25:24	http://bsf.gov.in/siteassets/sitepages/home/billing_details.zip

Latest detected files that were downloaded from this domain

Latest files that are **detected by at least one antivirus solution and were downloaded by VirusTotal from the domain provided**.

6/61	2018-03-31 07:10:26	0a2f29fea6ea3773431a4cf1e53a1a693bc039d82e7da5947917c4f819821d08
1/61	2018-03-31 07:07:45	5299be26edf4fb235cd33758c9710bbd1d6efd6c99cabb935ac6ea81312c378a
38/68	2017-10-16 18:59:58	5aa10115d8c8ab9b3b3f12b6ef24274a0cf7f15c442f0c92ab3d19ed55167b72
20/68	2018-03-20 18:06:38	5845c744a2032ffc6c630871d7eb684bd3d178bf464addbf38995061db77c9ad
1/59	2018-03-27 03:45:27	4c2a1d5c2bff7a035a3116e94fdcbc99891a437b013f31e068a5a2ff0ba374e3
13/59	2018-03-20 19:25:27	f3a73c68a6869aa60bbc73bd94f742cb0ada8f01c8e9d221dcd717c017294a3e

Figure 2. Virustotal URL detection

12. File hashes and resources

[1] de38e74b2cd493d0f014fc6ca5d2834cea213778c2e056a7c84e9547fe275889

[2] 0136b20b48cb3178fad12c1bd8d5d2779f3932f2c98de09e08eeddb3a84b4176

[3] 1cf67c25274473724a6cf614d45217edc16edbccebb3fa212f7b17e127362ea7

[4] 8b158aaf7215aea9766843219c2749d7ecb44986262107b38e234756738b3e7c

[5] 1fd13875bf3df273acc893c6a0fe0144a05d5534624471baaa48f014757301ef

[6] c446cd91b2f5e0ca77716ec361e75da03c8d4c1cbf4d83fe927ec04bb1c78a83

[7] 8c7f9824580ea6c2286604c2677e1d27b95c4b3f9b2151b17a463f94bb68aa66

[8] d38292508697e7cd8b38d8e0d159e6e67e7191305f883aec94aeb95887748b2f

[9] ac1d717e345a831766f7d00c22db54121c502c9de5723bcc95bff8bb040157c8

[10] 3d0e90b82c57abfb22ed76c020ab34449efbcfb1c3e20ae2a3874a4ab2cc1743

[11] 4f21bb54b372a932fa3d13c9dcfaee06fc7e691a9dd86af3a4e47d6d003e020f

[12] ace5a17702239cebf4ebc9ae034f3b2878e471978b895d3f461ef38fce7b1a40

If you want to stay updated about malware, be sure to follow these accounts:

RansomBleed - My personal twitter account about the latest malware reports.

GDataSoftwareAG – G DATAs twitter company account.

Blog – The G DATA blog about all kinds of security-related news.