ELF_TSCookie - Linux Malware Used by BlackTech

blogs.jpcert.or.jp/en/2020/03/elf-tscookie.html

朝長 秀誠 (Shusei Tomonaga)

March 5, 2020

BlackTech

-
- Email

In the past blog articles, we have introduced TSCookie, PLEAD and IconDown, which are used by BlackTech. It has been identified that this group also uses several other types of malware. While the malware we have already described infects Windows OS, we have also confirmed that there are TSCookie and PLEAD variants that infect Linux OS.
This article describes TSCookie for Linux, used by BlackTech.

Difference between TSCookie for Windows and Linux

The function of the two are mostly the same, as many parts of the code are identical. Figure 1 shows the comparison of code in TSCookie for Windows and for Linux.

Figure 1: Comparison of code in TSCookie for Windows and Linux(Left: Windows Right: Linux)

While they are mostly the same in terms of the code, the Linux version operates differently with the following characteristics:

- Less configuration
- Supports custom communication protocol only
- Several functions available by default

The details are explained in the next sections.

Less configuration data

As it was described in the past blog entry (Appendix A: TSCookie Configuration), TSCookie for Windows has 17 sets of configuration within the 0xB78 data size. On the other hand, it is reduced to 5 in the Linux version, and the configuration on proxy communication and others have been excluded. See Appendix A for details.
In the Windows version, the configuration is RC4-encrypted and hardcoded in the malware. For the Linux version, however, information such as C&C server is copied as a plain text into a dedicated area in the memory and then RC4-encrypted.It is uncertain why the Linux version malware does not encrypt the configuration with RC4 from the beginning, but it is possible that coding some parts did not work when copying the code from the Windows version to the Linux one.

```
17  memset(&key_enc_config, 0, 0x2000u);
18  memset(&config, 0, sizeof(config));
19  strcpy((char *)&host, "app.dynamicrosoft.com@443;home.mwbsys.org@443");
20  mal_set_config_data((int)&config, (char *)&host);
21  config.rc4key.key2 = 0x23D;
22  key_data.data1 = 0x696D6461;
23  key_data.data2 = 0x216E;
24  key_data.data3 = 0;
25  key_data.data4 = 0;
26  config.rc4key.key1 = mal_ror4_hash(&key_data);
27  config.connect_mode = 0;
28  strcpy(config.id, "ATS-      );
29  v3 = &config_key;
30  v4 = 0x80;
31  if ( (unsigned __int8)&config_key & 2 )
32  {
33    *(_WORD *)&config_key = 0;
34    v3 = (char *)&v9;
35    v4 = 0x7E;
36  }
37  memset(v3, 0, 4 * (v4 >> 2));
38  v5 = &v3[4 * (v4 >> 2)];
39  v6 = &v3[4 * (v4 >> 2)];
40  if ( v4 & 2 )
41  {
42    *(_WORD *)v5 = 0;
43    v6 = v5 + 2;
44  }
45  if ( v4 & 1 )
46    *v6 = 0;
47  mal_create_rc4key(&config_key, 0x80);
48  memcpy(&key_enc_config, &config_key, 0x80u);
49  memcpy(&config_data, &config, 0xB78u);
50  mal_rc4(&config_data, 0xB78, &config_key, 0x80);
51  mal_main((int)&key_enc_config);
52  return 1;
53}
```

Allocate data area for the configuration data

Copy configuration data

Encrypt configuration data

Figure 2: Code for creating configuration

Supports custom communication protocol only

While TSCookie for Windows supports several communication protocols (HTTP, HTTPS and custom protocol), the Linux version only supports its custom protocol. Figure 3 shows a part of code for communication. It is clear that the code only covers the custom protocol.

```
int __cdecl mal_1st_request(struc *s)
{
  int result; // eax

  if ( s->mode_flag )
    result = mal_http_connect_set(s);
  else
    result = mal_tcp_connect_set(s);
  return result;
}
```

```
int __cdecl mal_1st_request(this2 *a1)
{
  int result; // [esp+1Ch] [ebp-Ch]

  result = -1;
  if ( !a1->mode_flag )
    result = mal_tcp_connect_set(a1);
  return result;
}
```

Figure 3: Comparison of communication in TSCookie for Windows and Linux(Left: Windows version Right: Linux version)

The payload itself is RC4-encrypted in both versions, and the format of the data as well as the commands received in reply remain mostly the same. (See Appendix B for details.)

Several functions available by default

TSCookie for Windows downloads modules and operates accordingly. The Linux version has the following functions by default, so it conducts malicious activities without downloading extra modules. (See Appendix C for details.)

- Execute arbitrary shell command
- Operate files (list, delete, move)
- Upload/Download files

In closing

It is assumed that the malware is embedded in a Linux server of a victim organisation by an attacker after intrusion. If you find any type of malware related to Blacktech in your network, it is recommended that you also check your Linux environment. Please see Appendix D for the list of C&C servers.

Shusei Tomonaga
(Translated by Yukako Uchida)

Appendix A: ELF_TSCookie Configuration

Table A: Configuration

Offset	Description	Remarks
0x000	Destination server and port number	Multiple hosts can be specified by listing with a semicolon ";"
0x400	RC4 key	Used for encrypting communication
0x40C	Campaign ID	
0x44C	Communication mode	Only supports a custom protocol
0x454	Not used	

Appendix B: Data exchanged by ELF_TSCookie

Table B-1: Format of sent data

Offset	Length	Contents
0x00	4	Number of received data (begins with 0xFFFFFFFF)
0x04	4	Length of data sent
0x08	4	Packet number (Used to divide data when the data length is larger than 65440)
0x0C	4	Command (begins with 0x7263BC02)
0x10	4	Whether the data after 0x20 is RC4-encrypted
0x14	4	Not used
0x18	4	0x3001
0x1C	4	RC4 key (random data)
0x20	-	Data to be sent (See B-2 for the first communication)

Up to offset 0x1C, the contents are encrypted with the RC4 key and random data in the configuration.

Table B-2: Format of data sent in the first communication after offset 0x20

Offset	Length	Contents
0x00	4	0x9A65001F
0x04	4	Process ID
0x08	4	Command (0x7263BC02 at the beginning)
0x0C	4	Not used
0x10	4	Data size after offset 0x14
0x14	-	Random data

Up to offset 0x14, the contents are encrypted with RC4 key and random data in the configuration.

Table B-3: Format of received data

Offset	Length	Contents

0x00	4	Number of received data
0x04	4	Length of received data
0x0C	4	Command
0x10	4	Whether the data after 0x20 is RC4-encrypted
0x1C	4	RC4 key
0x20	-	Data

Up to offset 0x1C, the contents are encrypted with RC4 key in the configuration and another key in the received data.

Appendix C: ELF_TSCookie commands

Table C: Commands

Value	Contents
0x7200AC03	Launch remote shell
0x7200AC04	Send a command to remote shell
0x7200AC05	End remote shell
0x7200AC07	-
0x7200AC0B	Returns 0x7263BC06
0x7200AC0C	List files
0x7200AC0D	Download file
0x7200AC0E	Upload file
0x7200AC11	-
0x7200AC13	End bot
0x7200AC16	Delete file
0x7200AC1A	Move file
0x7200AC10	Execute command

Appendix D: C&C servers

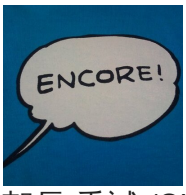
- app.dynamicrosoft.com
- home.mwbsys.org

Appendix E: Hash

fc863fbd71e22c99eaa2b1b0eb72d806cedeb536213e600afb03f0fbea9d2bb3

-
- Email

Author



朝長 秀誠 (Shusei Tomonaga)

Since December 2012, he has been engaged in malware analysis and forensics investigation, and is especially involved in analyzing incidents of targeted attacks. Prior to joining JPCERT/CC, he was engaged in security monitoring and analysis operations at a foreign-affiliated IT vendor. He presented at CODE BLUE, BsidesLV, BlackHat USA Arsenal, Botconf, PacSec and FIRST Conference. JSAC organizer.

Was this page helpful?

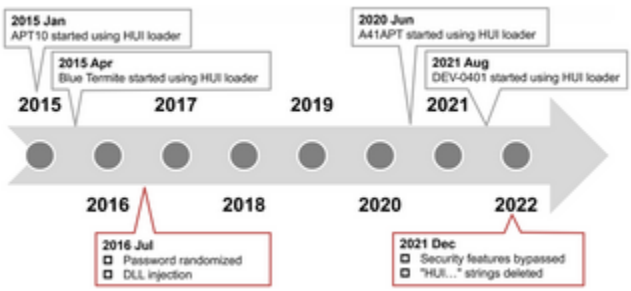
0 people found this content helpful.

If you wish to make comments or ask questions, please use this form.

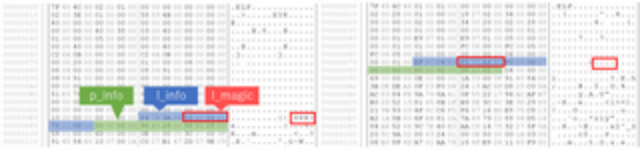
This form is for comments and inquiries. For any questions regarding specific commercial products, please contact the vendor.

please change the setting of your browser to set JavaScript valid. Thank you!

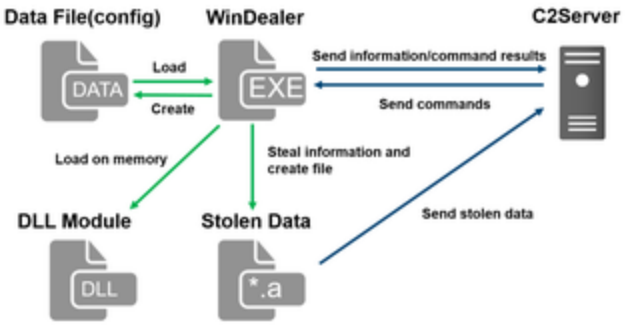
Related articles

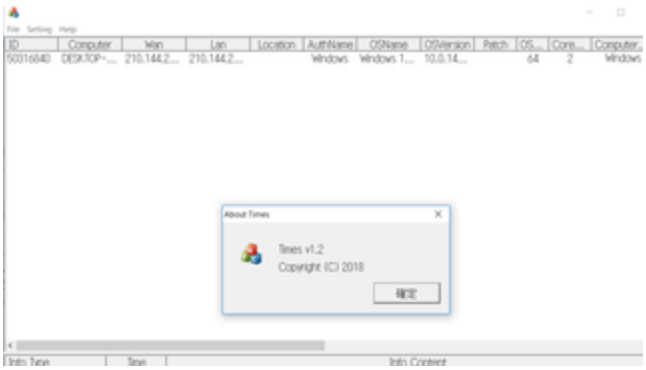
- 

Analysis of HUI Loader

- 

Anti-UPX Unpacking Technique
- 

FAQ: Malware that Targets Mobile Devices and How to Protect Them
- 

Malware WinDealer used by LuoYu Attack Group
- 
Malware Gh0stTimes Used by BlackTech

Back
Top
Next