Billbug: State-sponsored Actor Targets Cert Authority, Government Agencies in Multiple Asian Countries

State-sponsored actors compromised a digital certificate authority in an Asian country during a campaign in which multiple government agencies were also targeted.

Symantec, by Broadcom Software, was able to link this activity to a group we track as Billbug due to the use in this campaign of tools previously attributed to this group. Billbug (aka Lotus Blossom, Thrip) is a long-established advanced persistent threat (APT) group that is believed to have been active since at least 2009. Symantec has previously published on this group's activity in 2018 and 2019 under the Thrip name, but following our 2019 investigation, we determined that Thrip and Billbug were most likely the same group so now track all activity under the Billbug name.

In activity documented by Symantec in 2019, we detailed how the group was using a backdoor known as Hannotog (Backdoor.Hannotog) and another backdoor known as Sagerunex (Backdoor.Sagerunex). Both these tools were also seen in this more recent activity.

The victims in this campaign included a certificate authority, as well as government and defense agencies. All the victims were based in various countries in Asia. Billbug is known to focus on targets in Asian countries. In at least one of the government victims, a large number of machines on the network were compromised by the attackers.

The targeting of a certificate authority is notable, as if the attackers were able to successfully compromise it to access certificates they could potentially use them to sign malware with a valid certificate, and help it avoid detection on victim machines. It could also potentially use compromised certificates to intercept HTTPS traffic. However, although this is a possible motivation for targeting a certificate authority, Symantec has seen no evidence to suggest they were successful in compromising digital certificates. Symantec has notified the cert authority in question to inform them of this activity.

This activity has been ongoing since at least March 2022.

Attack chain

There are some indications that the attackers are exploiting public-facing applications to gain initial access to victim networks.

The attackers use multiple dual-use tools in this attack campaign, as well as custom malware. Billbug's extensive use of dual-use and living-off-the-land tools was also notable in its previous campaigns. Among the dual-use tools leveraged in this recent activity are:

- **AdFind** – A publicly available tool that is used to query Active Directory. It has legitimate uses but is widely used by attackers to help map a network.
- **Winmail** – Can open winmail.dat files.
- **WinRAR** – An archive manager that can be used to archive or zip files - for example, prior to exfiltration.
- **Ping** – A tool that is freely available online that can allow users to determine if a specific location on a network is responding.
- **Tracert** – A network tool that can be used to determine the “path” packets take from one IP address to another. It provides the hostname, IP address, and the response time to a ping.
- **Route** – A path for sending packets through the internet network to an address on another network.
- **NBTscan** – Open-source command-line NetBIOS scanner.
- **Certutil** – Microsoft Windows utility that can be used for various malicious purposes, such as to decode information, to download files, and to install browser root certificates.
- **Port Scanner** – Allows an attacker to determine what ports are open on a network and could potentially be used to send and receive data.

Multiple files that are believed to be loaders for the Hannotog backdoor were spotted on victim machines. A backdoor was then deployed on the compromised system. This backdoor has multiple functionalities:

- It executes netsh to update the firewall settings:

  ```
  netsh advfirewall firewall add rule name="Core Networking - Router Solicitation (ICMP-In)" dir=in action=allow program="%s" enable=yes
  netsh firewall add portopening UDP 5900 @xpsp2res.dll,-22006 ENABLE ALL',0
  netsh firewall add allowedprogram name="SNMP Trap Service" program="%s" mode=enable
  ```

- Listens on port 5900
- Can create a service for persistence
- Can also stop services
- Can upload encrypted data
- Can execute *cmd.exe /c %s command* to gather system information
- Can download files to the machine

A tool called Stowaway Proxy Tool was also downloaded to victim machines. Stowaway is a multi-level proxy tool written in the Go language and intended for use by penetration testers. Users can use this program to proxy external traffic to the intranet through multiple nodes, break through intranet access restrictions, construct a tree-like node network, and easily implement management functions. It is not unusual to see penetration testing tools misused by threat actors. Cobalt Strike, which is a penetration testing framework, is considered commodity malware by many due to how often it is used by malicious actors.

Sagerunex - Technical details

The Sagerunex backdoor is fairly resilient and implements multiple forms of communication with its command-and-control (C&C) server. The analyzed sample had no hardcoded configuration, so it had to be dropped on the machine by a loader malware, such as Hannotog.

In the sample analyzed by Symantec, configuration is passed to the sample via a parameter of the exported function (called MainEntry). That configuration is decrypted with a simple XOR operation:

```
def simplecrypt(x):

   return xor(x, b"\xad" + x[:-1])
```

Next, the sample finds the explorer.exe process and uses it to change the token of the current thread. It then writes logs to a temporary file (*%TEMP%/TS_FB56.tmp*), but only if the file already exists. These logs are encrypted and the encryption algorithm used is AES256-CBC with 8192 rounds of SHA256:

```
def decrypt(datasample):

    key =
b'\x53\x12\x76\x23\x94\x89\x78\x45\x58\x31\x62\x83\x77\x95\x59\x17\x31\x47\x73\x50\x22\x34\x65\x89\x49\x12\x67\x41\x90\x3
```

```
  realkey = datasample[:0x10]+ b'\x00' * 0x10

   for i in range(0x2000):
     realkey = hashlib.sha256(realkey + key).digest()

  raw = aes.cbc.decrypt(realkey, datasample[:16], datasample[16:])

   print("checksum", raw[-32:].hex())
   return raw[:-32]
```

The encryption key is hardcoded, and was previously used in an older sample of this malware. The same encryption algorithm is used for network communication. The encrypted data structure is as follows:

```
struct encrypted_data {
   byte[16] IV;
   byte[N] message_data;  // always divisible by 16, padding added as necessary
   byte[32] sha256_checksum;
}
```

The sample stores configuration and state in the following file:

%appdata%/microsoft/protect/windows/DMI%X.DAT (where %X is variable and depends on the parameter with which sample was started).

It is also encrypted, but with RC4. This key was hardcoded in older versions of Sagerunex, but recent samples started to read the key from configuration instead. The config file modification date will always be in the year 2011 – the “file last edit” year is changed by the malware to 2011.

Network communication

In normal mode, the sample will try all the following supported connection modes in this order. In all cases, HTTPS is used, with user agent equal to: Mozilla/5.0 (compatible; MSIE 7.0; Win32).

- *1: httpsviaconfigproxy: HTTPS with configured proxy*

- *2: httpswpadproxy: instead of using configured proxy, use proxy provided by WPAD mechanism (web proxy autodiscovery)*

- *3: httpsviaiexproxy: self-explanatory. Use proxy from \\Software\\Microsoft\\Windows\\CurrentVersion\\Internet Settings\\ProxyServer*

- *4: httpsviafirefoxproxy - get proxy from \\Mozilla\\Firefox\\profiles.ini (one of FF config files)*

- *5: httpsviaautoproxy - use proxy obtained from WinHttpGetIEProxyConfigForCurrentUser*

- *6: httpspreconfig - try to connect without proxy*

The network packet is composed of two parts: the header and the payload. Both are encrypted separately.

```
struct network_packet {
   byte[64] encrypted_header;  // see encrypted_data above
   byte[N] encrypted_payload;  // see encrypted_data above
}
```

The structure of the decrypted header is as follows:

```
struct header {
   int32 command_id;
   int16 packet_length;
   int32 packet_crc32;
}
```

The structure of the decrypted payload mostly depends on the command ID. The list of supported commands includes:

- 7: Return the list of currently configured proxies.
- 9: Execute a program, DLL or shell command. There are three supported subcommands: "runexe" to run an executable, "rundll" to run a DLL file, and anything else for arbitrary shell command.
- 11: Steal a local file (gets a file name specified in the command payload).
- 15: Get a configured file path (configured by command 18).
- 17: Drop a file to a specified path – but only if the specified path was previously selected by command 18.
- 18: Select a file path for commands 15 and 17.

Motivation

While we do not see data being exfiltrated in this campaign, Billbug is widely regarded as being an espionage actor, indicating that data theft is the most likely motivation in this campaign. The victims in this campaign – government agencies and a certificate authority – also point to an espionage and data-theft motive. The targeting of the government victims is most likely driven by espionage motivations, with the certificate authority likely targeted in order to steal legitimate digital certificates, as mentioned in the introduction.

This is potentially very dangerous, as if Billbug is able to sign its malware with a valid digital certificate it may be able to bypass security detections on victim machines. The ability of this actor to compromise multiple victims at once indicates that this threat group remains a skilled and well-resourced operator that is capable of carrying out sustained and wide-ranging campaigns. Billbug also appears to be undeterred by the possibility of having this activity attributed to it, with it reusing tools that have been linked to the group in the past.

Protection

For the latest protection updates, please visit the Symantec Protection Bulletin.

Indicators of Compromise

072022b54085690001ff9ec546051b2f60564ffbf5b917ac1f5a0e3abe7254a5

0cc6285d4bfcb5de4ebe58a7eab9b8d25dfcfeb12676b0c084e8705e69f6f281

148145b9a2e3f3abdc6c2d3de340eabc82457be67fb44cfa400a5e7bd2f88760

2a4302e61015fdf5f65fbd456249bafe96455cd5cc8aefe075782365b9ae3076

3585a5cbbf1b8b3206d7280355194d5442ed997f61e061fd6938a93163c79507

37fe8efe828893042e4f1db7386d20fec55518a3587643f54d4c3ec82c35df6d

3c35514b27c57a46a5593dbbbfceddbc49979b20fddc14b68bf4f0ee965a7c59

3dd7b684024941d5ab26df6730d23087037535783e342ee98a3934cccddb8c3e

64c546439b6b2d930f5aced409844535cf13f5c6d24e0870ba9bc0cf354d8c11

79f9f25b15e88c47ce035f15dd88f18ecc11e1319ff6f88568fdd0d327ad7cc1

7fe67567a5de33166168357d663b85bd452d64a4340bdad29fe71588ad95bf6f

80a8a9a2e91ead0ae5884e823dca73ef9fce59ff96111c632902d6c04401a4fe

861d1307913d1c2dbf9c6db246f896c0238837c47e1e1132a44ece5498206ec2

8f7c74a9e1d04ff116e785f3234f80119d68ae0334fb6a5498f6d40eee189cf7

a462085549f9a1fdeff81ea8190a1f89351a83cf8f6d01ecb5f238541785d4b3

adb61560363fcda109ea077a6aaf66da530fcbbb5dbde9c5923a59385021a498

bcc99bc9c02e1e2068188e63bc1d7ebe308d0d12ce53632baa31ce992f06c34a

b631abbfbbc38dac7c59f2b0dd55623b5caa1eaead2fa62dc7e4f01b30184308 c4a7a9ff4380f6b4730e3126fdaf450c624c0b7f5e9158063a92529fa133eaf2

e4a460db653c8df4223ec466a0237943be5de0da92b04a3bf76053fa1401b19e

f7ea532becda13a1dcef37b4a7ca140c56796d1868867e82500e672a68d029e4

f969578a0e7fe90041d2275d59532f46dee63c6c193f723a13f4ded9d1525c6b

fea2f48f4471af9014f92026f3c1b203825bb95590e2a0985a3b57d6b598c3ff