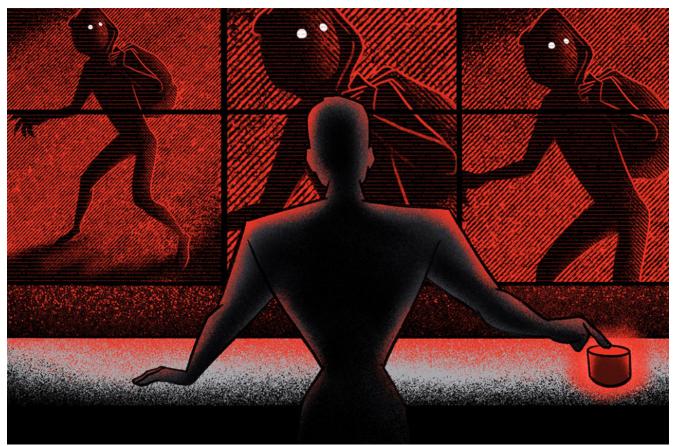
# HijackLoader Expands Techniques to Improve Defense Evasion

crowdstrike.com/blog/hijackloader-expands-techniques/

Donato Onofri - Emanuele Calvelli

February 7, 2024



- HijackLoader continues to become increasingly popular among adversaries for deploying additional payloads and tooling
- A recent HijackLoader variant employs sophisticated techniques to enhance its complexity and defense evasion
- CrowdStrike detects this new HijackLoader variant using machine learning and behavior-based detection capabilities

CrowdStrike researchers have identified a HijackLoader (aka IDAT Loader) sample that employs sophisticated evasion techniques to enhance the complexity of the threat. HijackLoader, an increasingly popular tool among adversaries for deploying additional payloads and tooling, continues to evolve as its developers experiment and enhance its capabilities.

In their analysis of a recent HijackLoader sample, CrowdStrike researchers discovered new techniques designed to increase the defense evasion capabilities of the loader. The malware developer used a standard process hollowing technique coupled with an additional trigger that was activated by the parent process writing to a pipe. This new approach has the potential to make defense evasion stealthier.

The second technique variation involved an uncommon combination of process doppelgänging and process hollowing techniques. This variation increases the complexity of analysis and the defense evasion capabilities of HijackLoader. Researchers also observed additional unhooking techniques used to hide malicious activity.

This blog focuses on the various evasion techniques employed by HijackLoader at multiple stages of the malware.

# HijackLoader Analysis

## Infection Chain Overview

The HijackLoader sample CrowdStrike analyzed implements complex multi-stage behavior in which the first-stage executable (streaming\_client.exe) deobfuscates an embedded configuration partially used for dynamic API resolution (using PEB\_LDR\_DATA structure without other API usage) to harden against static analysis.

Afterward, the malware uses WinHTTP APIs to check if the system has an active internet connection by connecting to https[:]//nginx[.]org. If the initial connectivity check succeeds, then execution continues, and it connects to a remote address to download the second-stage configuration blob. If the first URL indicated below fails, the malware iterates through the following list:

- https[:]//gcdnb[.]pbrd[.]co/images/62DGoPumeB5P.png?o=1
- https[:]//i[.]imgur[.]com/gyMFSuy.png;
- https[:]//bitbucket[.]org/bugga-oma1/sispa/downloads/574327927.png

Upon successfully retrieving the second-stage configuration, the malware iterates over the downloaded buffer, checking for the initial bytes of a PNG header. It then proceeds to search for the magic value C6 A5 79 EA, which precedes the XOR key (32 B3 21 A5 in this sample) used to decrypt the rest of the configuration blob.

| 40       B1       78       A2       EF       D0       EF       78       A2       EF       07       66       85       C6       35       C1       B0       D2       24       56       67       C2       C5       C5 <t< th=""><th></th><th></th></t<>   |   |   |
|---|---|---|
| EB 47 E0 D2 CF 31 03 EC 7D DA C4 6C 45 FB DF B5 14 25 D6 23 D5 B1 26 AC 41 0E 4D FF<br>64 A0 FC 69 A6 40 76 C7 E7 67 C8 D7 20 AC 66 49 66 D6 63 AF A2 50 25 2C 16 14 AF A8<br>0 10 10 46 47 05 71 E7 15 00 73 0B 73 DB 07 37 DB 02 C0 B4 CA 55 EF 7B 60 12 BA 2F 80 7F F4 2F 08 E8 80<br>C7 99 F7 30 DF F8 4D 99 D5 73 DB 02 C0 B6 2A 55 EF 7B 60 12 BA 2F 80 7F F4 2F 08 E8 80<br>C7 99 F7 36 A7 56 A4 50 F7 A0 20 20 D8 C0 A5 AE C7 0D A7 05 B4 A3 EA 50 71 E7 14 2F 08 7F 74 2F 08 4E 80<br>C7 99 F7 36 A8 56 44 50 F7 A0 58 40 C2 60 90 71 4 BD 43 EE 50 77 55 D9 67 28 44 80<br>C7 99 F7 36 A8 56 44 50 F7 A0 58 40 C2 60 90 71 4 BD 43 EE 50 77 55 D9 67 28 44 80<br>C7 99 F7 36 A8 56 44 50 F7 A0 58 40 C2 60 90 71 4 BD 43 EE 50 77 58 D8 72 80 43<br>C7 99 F7 36 A8 56 44 50 F7 A0 58 15 B3 21 A5 57 3D 80 62 E5 28 B3 31 A5 CF 31 08 00<br>B1 E7 7F 98 3F D0 00 00 00 48 44 41 56 06 A5 79 EA 32 B3 20 A4 32 DE 21 A5<br>C7 98 D7 50 2A 57 02L A3<br>C7 98 D7 50 2A 53 2D 32 D3 20 B3 20 A4 32 DE 21 A5<br>C7 98 D5 70 E2 LA5 15 B3 21 A5 06 F2 05 80 EC AN 59 EF 60 D 25 D8<br>C7 14 D0 10 93 23 14 5D A4 7C 13 LB B3 53 28 D3 20 B5 20 B5 20 B2 50 DF 50 DF 50 DF 50 BC 57 58 00 C A7 10 D 12 4F 73 4D A0<br>L4 55 45° 14° 14° 14° 14° 14° 14° 14° 14° 14° 14   | 40 B1 7B A2 EF DD EE F3 AC B6 37 6C 61 53 C1 BB BF FD FB 2A 33 52 18 02 A1 9D 28 45 |   |
| Cc 5A B3 94 Cc 73 Ec 72 F2 7C 6D F2 D3 AC 96 49 62 D6 39 3A FA 23 C2 2C 72 19 AE D7<br>1B 20 94 60 70 4B 4A 55 56 41 96 32 C1 BA AE AB 10 D1 B2 66 6A 6C D2 26 CA 26 EA 00 E71<br>7 86 47 BA D1 D6 AF 45 F3 91 AB 86 2D 18 47 B3 F4 71 50 12 02 2C 29 E1 66 2A 96 77 F4 2F CB 4E 0C<br>7 86 47 BA D1 D6 AF 45 F3 91 AB 86 2D 18 47 B3 F4 71 50 12 02 2C 29 E1 66 23 9B 53<br>7 86 47 BA D1 D6 AF 45 F3 91 AB 86 2D 18 47 B3 F4 71 50 12 02 2C 29 E1 66 2A 96 77 55 B5 75 16 2C 75 96 65 65 96 63 60 00 27 96 89 75 98 73 6A 33 64 45 0 FA 05 48 0 02 80 00 81 00 00 80 00 88 44 41 95 48 26 90 79 55 89 73 86 43 64 45 0 FA 05 48 0 02 80 00 81 00 00 80 00 88 44 41 56 6X 30 82 F4 50 10 00 81 64 14 55 6X 48 0 02 80 01 88 40 12 80 27 80 81 20 14 93 81 69 07 95 89 72 84 A2 0<br>10 0 0 0 37 0 F21 A5 32 83 29 85 15 B3 31 A9 45 1C 21 35 32 83 20 A4 32 DE 21 A5 54<br>52 78 80 10 D9 32 31 45 DA 4F C1 58 B3 33 80 6F E0 58 D6 65 FC 21 F2 04 87 70 C6 67<br>F5 31 CF 35 34 59 13 83 21 40 87 31 33 80 6F E0 58 D6 65 FC 12 F2 04 87 70 C6 67<br>F5 31 CF 35 34 59 13 83 21 C2 14 A5 73 83 80 2F 80 58 D6 65 FC 12 F2 04 87 70 C6 67<br>F5 31 CF 35 34 59 13 B3 21 42 C3 41 DB A1 D1 5 FD F0 C1 C5 ED F2 54 95 25 B7 F6 DD 70 L1 247 73 4D A0<br>C4 34 A4 35 B3 2C CB BA A0 21 A5 B9 FF 01 59 63 8C F9 58 DB CA AF0 12 47 73 4D A0<br>C4 34 A4 35 B3 2C CB BA A0 21 A5 B9 FF 01 59 63 8C F9 58 DB CA AF0 12 47 73 4D A0<br>C4 76 4 CF 1A 71 L2 BA 1A 27 F8 31 DF 7 12 F8 H3 28 32 C1 A5 02 F1 12 67 12 F7 16 C2 F1 F2 06 102 F1 F2 06 F1 F1 F1 F2 F2 F1 F2 06 F1 F1 F2 F2 F1 F  | A6 4E 75 78 68 B8 18 3D 6B B6 7D DF 8C 64 35 32 BB A4 CA 1C B0 8D 07 F5 C5 FC 32 CF |   |
| 1B 20 94 60 70 4B 4A 58 56 41 98 32 C1 BA AE AB 10 D1 B2 66 6A 6C CD 28 6E A0 0E 71,       ""pRJXWA2A5es, 8*f31(n, q)         04 A0 F6 C3 00 BF F4 09 90 57 30 B0 6C 4A 55 EF 7B 60 12 BA 2F 88 7F F4 2F CB 4E 6C,       "psD200x601x4*2, 4*1, 4*1, 4*1, 4*1, 4*1, 4*1, 4*1, 4*1  | EB 47 E0 D2 CF 31 03 EC 7D DA C4 6C 45 FB DF B5 16 25 D6 23 D5 B1 26 AC 44 0E 4D 81 |   |
| 84 A0 F6 C9 30 BF F8 40 99 D5 73 DB 0E 4C A5 EE 78 60 12 BA 2F 88 7F F4 2F C8 4E 8C       , otio_Quercian       otio_Quercian       MAGIC. C0 A5 79 EA         76 64 71 8A D1 D6 AF 45 73 91 A8 66 2D 164 71 85 12 02 02 29 E1 65 29 B5 53.       fc:NoPC 56 vietsGidQ:,viets.sc;       fc:NoPC 56 vietsGidQ:,vietsSc;       fc:NoPC 56 vietsGidQ:,viets.   | C8 9A B3 94 C4 73 E6 7E F8 7C 8D F2 D3 AC 96 49 68 D6 39 3A FA 25 02 52 CE 19 AE D  |   |
| 77       68       47       BA       D       BA       F4       75       91       AB       74       71       50       12       22       22       22       21       65       29       B5       33       .16*MO       26/star)       As       As       C       59       97       30       80       60       60       60       71       48       30       21       AS       CF       10       00       00       00       60       44       12       14       10 <td>18 20 94 60 70 48 4A 58 56 41 98 32 C1 BA AE AB 10 D1 B2 66 6A 6C CD 28 6E A0 0E 71</td> <td>1 . "`pKJXVA~2Á°®«.ѰfjlÍ(n .q MACIC: C6 AE 70 EA</td>  | 18 20 94 60 70 48 4A 58 56 41 98 32 C1 BA AE AB 10 D1 B2 66 6A 6C CD 28 6E A0 0E 71 | 1 . "`pKJXVA~2Á°®«.ѰfjlÍ(n .q MACIC: C6 AE 70 EA          |
| 77       68       47       BA       D       BA       F4       75       91       AB       74       71       50       12       22       22       22       21       65       29       B5       33       .16*MO       26/star)       As       As       C       59       97       30       80       60       60       60       71       48       30       21       AS       CF       10       00       00       00       60       44       12       14       10 <td>84 A0 F6 C9 30 BF FB 4D 99 D5 73 DB 0E 4C A5 EE 7B 60 12 BA 2F 88 7F F4 2F CB 4E 80</td> <td>C "öÉ0¿ûM™ŐsÜ.L¥î{`.º/^.ô/ËNE NIAGIC. COAS /9 EA</td>  | 84 A0 F6 C9 30 BF FB 4D 99 D5 73 DB 0E 4C A5 EE 7B 60 12 BA 2F 88 7F F4 2F CB 4E 80 | C "öÉ0¿ûM™ŐsÜ.L¥î{`.º/^.ô/ËNE NIAGIC. COAS /9 EA          |
| CCE 59 9F 39 6A 83 64 45 0F FA 05 48 0C 26 09 C7 14 BD 43 BE 69 07 95 B9 67 28 4A 2D<br>E0 88 1E 77 F9 B3 EB 00 00 80 00 19 44 41 84 16 26 A5 79 EA 32 B3 21 A5 CF 31 08 00 63<br>A <ul> <li>uhr 20, 10 09 32 07 07 21 A5 32 B3 28 15 B3 21 A9 45 1C 21 35 32 B3 20 A4 32 DE 21 A5 4</li> <li>uhr 21, 10 20 00 19 32 14 5 DA 17 15 B3 58 06 E 50 58 D6 56 FC 21 F2 04 77 DC 68 2F F1 32</li> <li>uhr 21, 10 20 00 18 10 93 21 45 DA 17 18 38 31 40 15 17 B3 89 F4 32 E3 21 A7 76 B3 2F F1 32</li> <li>uhr 21, 10 20 00 18 10 93 21 45 DA 17 18 38 30 66 E 50 52 12 F2 04 77 DC 68 2F F1 32</li> <li>uhr 21, 10 20 10 93 21 45 DA 17 18 38 31 40 15 F DF 0F C1 5E DF E4 94 32 E6 AA 49 B1</li> <li>uhr 21, 10 20 14 50 97 12 1A 58 97 10 15 96 35 B7 F5 80 BC AA 70 12 4F 73 4D 07</li> <li>Uhr 21, 10 14 14 14 12 04 avoid + 000 14 50 9F F0 15 96 53 DE C5 AD 86 DC 5A AF 01 12 4F 73 4D 07</li> <li>Uhr 21, 10 14 14 14 12 04 avoid + 000 14 50 9F 16 13 33 33 32 E BA 07 21 8F C7 83 19 F1 20 51 95 65 12 2.</li> <li>Uhr 21, 10 14 14 14 14 14 14 14 14 14 14 14 14 14</li></ul>  | 7F 86 47 BA D1 D6 AF 45 F3 91 AB B6 2D 18 47 B3 F4 71 50 12 02 2C 29 E1 65 29 BB 53 | 3 .†G°ÑÖ Eó`≪¶G°ôqP,)áe)≫S                                |
| E0 8 1E 77 F9 83 E8 00 00 80 00 <mark>89 44 41 54 C6 A5 79 EA 32 83 21 A5 CF 31 08 00 63<br/>At .wt *ec.TATEXP\$2* H2FT<br/>13 42 A5 03 83 15 95 32 83 21 C1 31 1B 30 A5 17 B3 89 E4 32 E3 21 A7 76 B3 2F F1 32<br/>B5 27 80 10 D9 32 31 45 DA 4F C1 58 B3 53 80 6E E0 58 D6 65 FC 21 F2 04 87 7D C6 5F<br/>p*e. U2EDOA(*SenaXodeit6.*)#<br/>p*e. U2EDOA(*SenaXodeit6.*)#<br/>DECRYPT from: 37 0F 21 A5<br/>DECRYPT </mark>   | 80 80 EB 9A F3 ED B6 D2 F5 B6 D2 ED B6 2A 05 AE CF 0D A7 05 B4 A3 EA 5C 18 0C C5 F3 | 9 €€ëšói¶Òõ¶Òi¶*.®Ï.§.′£ê\Åù                              |
| E0 8 1E 77 F9 83 E8 00 00 80 00 <mark>89 44 41 54 C6 A5 79 EA 32 83 21 A5 CF 31 08 00 63<br/>At .wt *ec.TATEXP\$2* H2FT<br/>13 42 A5 03 83 15 95 32 83 21 C1 31 1B 30 A5 17 B3 89 E4 32 E3 21 A7 76 B3 2F F1 32<br/>B5 27 80 10 D9 32 31 45 DA 4F C1 58 B3 53 80 6E E0 58 D6 65 FC 21 F2 04 87 7D C6 5F<br/>p*e. U2EDOA(*SenaXodeit6.*)#<br/>p*e. U2EDOA(*SenaXodeit6.*)#<br/>DECRYPT from: 37 0F 21 A5<br/>DECRYPT </mark>   | CE 59 9F 39 6A 83 64 45 0F FA 05 48 0C 26 09 C7 14 BD 43 BE 69 07 95 B9 67 28 4A 21 | D ÎYŸ9jfdE.ú.H.&.Ç.%C%i.*°g(J XOR KEY: 32 B3 21 A5        |
| B3 42 A5 03 B3 15 95 32 83 21 C1 31 1B 3D A5 17 B3 89 E4 32 E3 21 A7 76 B3 2F F1 32<br>B5 27 80 10 D9 32 31 45 DA 4F C1 58 B3 53 80 6E 00 58 D6 65 FC 21 F2 04 87 7D C6 5F<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*SenaXOdeuto.+)z<br>pre.021E00Å(*Sen  | E0 8B 1E 77 F9 B3 EB 00 00 80 00 49 44 41 54 C6 A5 79 EA 32 B3 21 A5 CF 31 08 00 63 |   |
| BS 27 80 10 D9 32 31 45 DA 4F C1 5B B3 53 80 6E E0 5S D6 65 FC 21 F2 04 87 7D C6 SF       µ*€.021E00Å[*3EnAXOGE16.+)E       DECRYP1 from: 37 0F 21 A5         D7 0F BD 57 C8 44 92 60 93 42 C8 41 DB A1 D1 5F DF 0F C1 SE DF E4 94 32 E6 AA 49 B1       **WED***BEÀ0Å[Å a.Å*Ba*2æ*14       **WED***BEÀ0Å[Å a.Å*Ba*2æ*14         SF 31 CF 36 B3 49 40 33 B3 21 40 BC 67 21 A5 32 30 E5 AD BB F6 DD 25 B8 F6 DD 71 B8 F6 DD 71 B5 F0 D7 11 AF 37       SB 76 DD 71 BA 57       **WED***BEÀ0Å[Å a.Å*Ba*2æ*14         SF 31 SF 31 CF 10 A5 47 A7 48 67 33 33 32 BE FA BA AF 01 12 4F 73 4D A0       La#5*\$1*! ¥*D.*C[X*4±40.08M         F5 23 AB BF 62 1 55 B1 CE D1 A5 47 A7 48 67 33 33 32 D1 3E FE A3 BA AC 33 20 25 2B D9 51 25 12       Z.¥680U2a0±35Å]D±0*.3(\$+D0*         38 71 E7 16 4C F1 A7 12 4B A1 A2 7F 83 D9 F4 DA 30 21 E4 B0 9D 96 33 6F 14 825 14       8qc.Lfs.K;e.fD00/8*.Zie#sh0]D±6*.004*         47 60 65 31 FE D5 F4 DA 84 E4 B7 C6 3A 61 80 BB B1 61 A9 B3 72 C1 E7 16 33 D5 2C 78       G'elbô0Å]*.#Eame*rác, 30,x         58 10 CE D7 33 20 CF B2 AF AC E8 E2 E2 C9 GD 30 85 01 AD F5 F6 F1 B9 72       IleEl [021bA301.co04]*.dea#*10±A642+         CE 61 5D CD C6 24 4C FD 73 32 CF B2 AF AC E8 E2 E2 C9 GD 30 85 01 AD F5 F6 F1 B9 72       IleEl [021bA301.co04]*.dea#*1.4504.ac840+         FB 01 D0 5B B2 0F E0 EA B3 1A E7 16 C7 7F 62 77 47 20 E4 E1 D9 29 28 7F 5F 70 4D 34       Lo016.co14**4±-00.AB 400+0064*         FB 01 D0 5B B2 0F E0 EA B3 1A E7 16 C7 7F 62 77 47 20 E4 E1 D9 29 28 7F 5F 70 4D 34       Lo016.co14**4±-00.AB 400+0064*         FB 01 D0 5B B2 0F E0 EA B3 1A E7 16 C7 7F 62 77 47 72 C2 FF F  | 1D OC 00 37 OF 21 A5 32 B3 29 B5 15 B3 21 A9 45 1C 21 35 32 B3 20 A4 32 DE 21 A5 54 | 47.!¥2")μ."!@E.!52" ¤2Þ!¥T                                |
| D7 OF BD 57 CB 44 92 60 93 42 C6 41 DB A1 D1 5F DF OF C1 5E DF E4 94 32 E6 AA 49 B1 *.\#ED'."BÈÀŪ(À D.A^DA"24*1±<br>57 31 CF 36 B3 49 4D 33 B3 21 40 BC 67 21 A5 32 30 E5 AD BB F6 DD 25 B9 F6 DD F0 A<br>116*1*4**4* 1*20A.ao%***050<br>4C 34 A4 35 B3 25 CD BA A0 21 A5 B9 FE 01 59 63 5B CF 95 B0 BC AA F0 12 4F 73 4D A0<br>57 23 AB BB F6 21 55 B1 CE D1 A5 47 A7 4B 87 33 33 82 2E BA 07 21 8F C0 62 31 96 F2<br>58 8F A5 53 63 87 45 53 23 86 38 DB 15 32 3D 13 23 D1 35 28 D1 87 23 BA AC 33 22 85 D9 51 22.<br>58 67 A5 53 23 86 38 D0 H 53 23 D1 32 3D 13 23 D1 37 23 20 E5 AD AA C3 32 82 52 B9 51 22.<br>58 67 A5 68 23 64 27 45 23 28 63 BD B1 52 30 D1 35 23 D1 32 3D 13 28 22 B0 95 13 26 12 2.<br>58 67 A5 87 A5 32 38 63 BD B1 52 02 60 C A2 36 67 20 C1 85 67 31 D9 11 6F 20 KG*U30*U3*.0c6AH*.<br>48 E7 E5 5A B2 32 24 3E 31 16 17 95 F0 26 E0 CA 23 66 67 20 C1 85 67 31 D9 11 6F 20 KG*U30*U3*.0c6AH*.<br>47 60 65 31 FE D5 F4 DA 84 E4 B7 C6 3A 61 80 BB D1 61 A9 B3 72 C1 E7 16 33 D5 2C 76 G*e1D00U3*.2c6AH*.<br>58 10 A9 23 B3 62 H9 FB 21 85 63 D9 52 52 E7 BB AA 81 B0 17 E1 B1 06 3E 11 6F 21 38 (D* 2*14*)*64ÅÅBÅDÅ.A%ÅTÅÅZ.<br>58 10 A9 23 B3 62 H9 FB 21 85 63 D9 25 22 E7 BB AA 81 B0 17 E1 B1 06 3E 11 E8 11 (.00*14*.0*04**.4*0*±ÅC*350**56űA0<br>59 4Å*0***********************************  | B3 42 A5 03 B3 15 95 32 83 21 C1 31 1B 3D A5 17 B3 89 E4 32 E3 21 A7 76 B3 2F F1 32 | 2 'B¥.'. • 2f!Å1.=¥. * ##2#!\$v'/#2                       |
| D7 OF BD 57 CB 44 92 60 93 42 C6 41 DB A1 D1 5F DF OF C1 5E DF E4 94 32 E6 AA 49 B1 *.\#ED'."BÈÀŪ(À D.A^DA"24*1±<br>57 31 CF 36 B3 49 4D 33 B3 21 40 BC 67 21 A5 32 30 E5 AD BB F6 DD 25 B9 F6 DD F0 A<br>116*1*4**4* 1*20A.ao%***050<br>4C 34 A4 35 B3 25 CD BA A0 21 A5 B9 FE 01 59 63 5B CF 95 B0 BC AA F0 12 4F 73 4D A0<br>57 23 AB BB F6 21 55 B1 CE D1 A5 47 A7 4B 87 33 33 82 2E BA 07 21 8F C0 62 31 96 F2<br>58 8F A5 53 63 87 45 53 23 86 38 DB 15 32 3D 13 23 D1 35 28 D1 87 23 BA AC 33 22 85 D9 51 22.<br>58 67 A5 53 23 86 38 D0 H 53 23 D1 32 3D 13 23 D1 37 23 20 E5 AD AA C3 32 82 52 B9 51 22.<br>58 67 A5 68 23 64 27 45 23 28 63 BD B1 52 30 D1 35 23 D1 32 3D 13 28 22 B0 95 13 26 12 2.<br>58 67 A5 87 A5 32 38 63 BD B1 52 02 60 C A2 36 67 20 C1 85 67 31 D9 11 6F 20 KG*U30*U3*.0c6AH*.<br>48 E7 E5 5A B2 32 24 3E 31 16 17 95 F0 26 E0 CA 23 66 67 20 C1 85 67 31 D9 11 6F 20 KG*U30*U3*.0c6AH*.<br>47 60 65 31 FE D5 F4 DA 84 E4 B7 C6 3A 61 80 BB D1 61 A9 B3 72 C1 E7 16 33 D5 2C 76 G*e1D00U3*.2c6AH*.<br>58 10 A9 23 B3 62 H9 FB 21 85 63 D9 52 52 E7 BB AA 81 B0 17 E1 B1 06 3E 11 6F 21 38 (D* 2*14*)*64ÅÅBÅDÅ.A%ÅTÅÅZ.<br>58 10 A9 23 B3 62 H9 FB 21 85 63 D9 25 22 E7 BB AA 81 B0 17 E1 B1 06 3E 11 E8 11 (.00*14*.0*04**.4*0*±ÅC*350**56űA0<br>59 4Å*0***********************************  | B5 27 80 10 D9 32 31 45 DA 4F C1 5B B3 53 80 6E E0 58 D6 65 FC 21 F2 04 87 7D C6 51 | E μ'€.Ů21EÚOÁ['S€nàXÖeü!ò.‡}E = DECRYPI from: 37 OF 21 A5 |
| 4C       34       A4       35       B3       25       CD       BA       A0       21       A5       B9       FC       01       59       B0       BC       AA       FO       12       4F       73       4D       A0       1485*8f°       14**4*6.0eM         F5       23       AB       BF       62       155       B1       CE       10       A5       47       A7       4B       77       33       38       22       BA       C1       64       Ax       01       Ax       Ax       18       C       21       16       C       64       Ax       18       C       18       A       C       18       A       A       C       18       A       S       18       A       S       18       A       C       18       A       S       18       18   | D7 OF BD 57 CB 44 92 60 93 42 C8 41 DB A1 D1 5F DF OF C1 5E DF E4 94 32 E6 AA 49 B1 | l ×.⅓WËD'``BÈAÛ;Ñ_B.Á^Bă″2æ*I±                            |
| F5       23       AB       BE       F6       21       55       B1       CE       D1       AS       47       74       86       73       33       33       33       32       DE       A0       71       16       CE       21       95       12       C       Add and and and and and and and and and a  | 5F 31 CF 36 B3 49 4D 33 B3 21 4D BC 87 21 A5 32 30 E5 AD BB F6 DD 25 B9 F6 DD F5 D2 |   |
| SA 8F A5 36 38 74 55 32 38 63 BD 81 53 23 D1 3E FE A3 BA AC 33 28 25 2B D5 51 25 12       2.¥66tU28cH±5Å>pte <sup>a</sup> -3(4+00€.         38 71 E7 16 4C F1 A7 12 4B A1 A2 7F 83 D9 F4 DA 30 21 E4 B0 9D 96 36 71 48 12 14       bqc.Lñ5.K;c.f0d00;4*.ûceñH4.         BE 7 E5 5A B2 32 24 3E 31 16 17 95 F0 26 E0 CA 25 66 72 0C 1 85 67 31 D9 11 66 16       kqc.Lñ5.K;c.f0d00;4*.ûceñH4.         47 60 65 31 FE D5 F4 DA 84 E4 B7 C6 3A 61 80 BB B1 61 96 F2 38 C4 F8 F1 71 62 95 12       S*åÅ'0-t*°∞2**±5Å'>pte <sup>a</sup> -3(4±0±A)         A7 AA E0 C2 B3 30 AC B2 BA 9C 9F B0 BA CA A7 D9 B1 61 96 F2 38 C4 F8 F1 71 63 21 52       S*åÅ'0-t*°∞2**±5Å'>pte <sup>a</sup> -3(±±6±A)         A7 AA E0 C2 B3 40 C B2 A9 C 9F B0 BA CA A7 D9 B1 61 96 F2 38 C4 F8 F1 71 63 21 52       S*åÅ'0-t*°∞2*±5Å'>ta*bicaðt         CE 61 50 CD C6 24 4C FD 73 32 CF B2 AF AC E8 E2 E2 C9 GD 30 85 01 AD F5 F6 F1 B9 72       IalEt1y21***********************************   | 4C 34 A4 35 B3 25 CD BA A0 21 A5 B9 FE 01 59 63 5B CF 95 B0 BC AA F0 12 4F 73 4D A0 | 0 L4#5*%1° !¥'b.Yc[I•°4*8.OsM                             |
| 38       71       E7       16       4C       F1       A1       A2       7F       83       D9       F4       DA       30       21       E4       B0       DD       D9       63       36       F1       48       25       14       Sqc_Lñ§.K;c.fÜd0Ú0!ä*.Úc6ňH*.         4B       E7       E5       SA       E3       24       32       11       61       95       D2       62       CA       E3       66       67       20       C1       E7       63       D5       C4       E3       67       01       63       D5       C1       E7       C1       E7       16       30       D2       C1       E3       C1       E3       E4       E5       10       E5       E1       E3       E4       E5       17       16       E5       E1       E5       E1       E5       E3       E2       E4       E5       E5       E1 <td>F5 23 AB BB F6 21 55 B1 CE D1 A5 47 A7 4B 87 33 33 38 2E BA 07 21 8F CD 62 31 96 F4</td> <td>2 õ#«»ö!U±ÍÑ¥G§K‡338.°.!.Íb1-ò</td>   | F5 23 AB BB F6 21 55 B1 CE D1 A5 47 A7 4B 87 33 33 38 2E BA 07 21 8F CD 62 31 96 F4 | 2 õ#«»ö!U±ÍÑ¥G§K‡338.°.!.Íb1-ò                            |
| 4B       E7       E5       5A       B2       32       24       3E       31       16       17       95       F0       26       E0       CA       E3       66       67       20       C1       85       67       31       D9       11       6F       20       Kçå2*25>163àåång Å_g10.o         47       60       65       31       FE       D5       F4       DA       84       E7       C6       A6       180       B5       16       19       B3       72       C1       F1       13       D5       C7       6       G       e1båðún_a*2:a@*xa@*råç.sðíx, aðín-a@*xa@*råç.sðíx, aðín-a@*xa@*råç.sðíx, aðín-a@*xa@*råç.sðíx, aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*råç.sðíx-aðín-a@*xa@*xa@*xa@*xa@*xa@*xa@*xa@*xa@*xa@*x   | 5A 8F A5 36 38 74 55 32 38 63 BD B1 53 23 D1 3E FE A3 BA AC 33 28 25 28 D9 51 25 12 |   |
| <pre>47 60 65 31 FE D5 F4 DA 84 E4 B7 C6 3A 61 80 BB B1 61 A9 B3 72 C1 E7 16 33 D5 2C 78 6 \$\circle{1}\$ bio.A: \$\mathbf{z}\$.a\$ \$\mathbf{x}\$.a\$ \$\mathbf{x}\$.a</pre> | 38 71 E7 16 4C F1 A7 12 4B A1 A2 7F 83 D9 F4 DA 30 21 E4 B0 9D D9 63 36 F1 48 25 14 | 4 8qç.Lñ§.K;c.fùôú0!ä°.ùc6ñH%.                            |
| A7       AA       E0       C2       B3       30       AC       B2       BA       CA       A7       D9       B1       61       96       F2       38       C4       F8       F1       71       62       95       12       \$*åų0-**œ¥**2\$  | 4B E7 E5 5A B2 32 24 3E 31 16 17 95 F0 26 E0 CA E3 68 67 20 C1 85 67 31 D9 11 6F 20 |   |
| SB 10 A9 32 B3 62 E4 B9 FB 21 85 63 D9 25 2E 67 BB AA 81 B0 17 E1 B1 CD 63 E1 8E B1       [.@2*ba*d]c0*a,p*.*.å±fcåž±         CE 61 5D CD C6 24 4C FD 73 32 CF B2 AF AC E8 E2 E2 C9 6D 30 85 01 AD F5 F6 F1 B9 72       falfEzLýs21*-dååÉmočôň±         B1 AC F0 66 33 73 66 1B BB AA 34 9A F2 C2 9D E0 36 E1 A5 3E 32 1E 24 2F EF D8 05       falfEzLýs21*-dååÉmočôň±         B1 AC F0 66 33 73 66 1B BB AA 34 9A F2 C2 9D E0 36 E1 A5 3E 32 1E 24 2F EF D8 05       falfEzLýs21*-dååÉmočôň±         FB 01 D0 5B B2 0F E0 EA B3 1A E7 16 C7 7F 62 77 47 20 E4 51 D9 29 28 7F 5F 70 4D 34       0.b[*.dě*], c.c.,we*k.*děAcQÜ)(,W4         C9 62 86 30 85 01 2C 77 5F AA A5 7F 6B A8 E8 C2 3E 74 51 22 E1 AC E0 DE F2 34 F4 DA       fbg0,w.**k.*děAcQÜ         C9 28 53 5B 2D AA 84 63 A4 77 92 C7 3E 2E 77 47 2A 2A 7F 33 E9 66 B1 8E 21 83 A -#f2m-="gowr(C.wGG,def2]:       salp 5D 70 D0 EB 86 30 1F E5 66 3D 36 50 5A CD 4C EA 23 37 F1 2B 77 B6 F3 70 F4 72         B3 A9 5D 70 D0 EB 86 30 1F E5 96 19 89 AB 35 7A AF 29 CE 7F 70 37 F3 E1 A2 36       Haã*_jiz*cw#d; jiz*i,wa*ä*W*dœqç, ó.ác6         S6 FF 65 38 7A E2 CF 73 E9 21 AA 85 72 47 9E E3 C4 2D A5 54 30 D8 FF 45 B5 AC E4 32       Yerzäfset:c&Aä+"c&Aä+"c&Aä+"c&Aä+"c&Aä+"c&Aä+"c&Aä+"c&Aä+"c&Aä+"c&Aä+"c&Aä+"c&Aa*"c         S9 2E 12 F2 70 99 E8 68 B3 21 A5 54 6A 20 D0 37 S3 21 E4 0E B0 E0 24 0A E3 64 BD 32       "op#eh*! ¥TŠ 92!A.*.d&aa*  | 47 60 65 31 FE D5 F4 DA 84 E4 B7 C6 3A 61 80 BB B1 61 A9 B3 72 C1 E7 16 33 D5 2C 76 | 8 G`elþŐôÚ"ä Æ:a€»±a©"rÁç.3Ő,x                            |
| CE 61 5D CD C6 24 4C FD 73 32 CF B2 AF AC E8 E2 E2 C9 6D 30 85 01 AD F5 F6 F1 B9 72<br>B1 AC F0 B6 63 73 66 1B BB A3 34 9A F2 2C 9D E0 36 E1 A5 3E 32 1E 24 26 FE FD 85 92<br>±-d€dsf.*4åð,.c,C,LwG àdů)(DM4<br>C9 62 B6 30 85 01 2C 77 5F AA A5 7F 6B A8 E8 C2 3E 74 51 22 E1 AC E0 DE F2 34 F4 DA<br>E5 83 B8 2D A3 44 63 A1 77 92 C7 3E 2E 77 47 23 0E 47 51 22 E1 AC E0 DE F2 34 F4 DA<br>E5 83 B8 2D A3 44 63 A1 77 92 C7 3E 2E 77 47 72 A2 A7 F3 3E 9 66 B1 8E 21 85 3A<br>=-425,,ex,,  | A7 AA E0 C2 B3 30 AC B2 BA 9C 9F B0 BA CA A7 D9 B1 61 96 F2 38 C4 F8 F1 71 62 95 12 | 2 §*àÂ'0⊣'°œŸ°°Ê§Ù±a-ò8Ăøñqb∙.                            |
| B1 AC F0 B6 63 73 66 1B B8 AA 34 9A F2 2C 9D E0 36 E1 A5 3E 32 1E 24 26 FE FD 85 09 ±→\$\delta delta d  | 5B 10 A9 32 B3 62 E4 B9 FB 21 85 63 D9 25 2E 67 BB AA 81 B0 17 E1 B1 CD 63 E1 8E B1 | l [.©2"bä`û!…cÙ%.g»*.°.á±ÍcᎱ                              |
| FB 01 D0 5B B2 0F E0 EA B3 1A E7 16 C7 7F 62 77 47 20 E4 51 D9 29 28 7F 5F 70 4D 34<br>C9 62 B6 30 85 01 2C 77 5F AA A5 7F 6B A8 E8 C2 3E 74 51 22 E1 AC E0 DE F2 34 F4 DA<br>P7 23 8E B3 BB 2D AA 84 63 A4 77 92 C7 3E 2E 77 47 A2 A0 7F 33 E9 66 B1 8E 21 85 A - + € <sup>2</sup> x→ -,∞x×',×'∈Å×C0*A+Ab∂4ôŬ<br>97 23 8E 73 B8 B2 D0 EB 86 30 1F E5 86 3D 36 50 5A CD 4C EA 23 37 F1 2B 77 B6 F3 70 F4 72<br>BC E3 9A B4 EF A1 9E 72 B2 69 2C 77 BF 61 98 9A B3 57 AA F2 9C E7 E7 03 F3 E1 A2 36<br>F45 7A F4 57 AF 42 9C E7 73 E9 21 AA 85 72 47 9E E3 C4 2D A5 54 30 D8 FF 45 B5 AC E4 32<br>93 2E 12 F2 70 99 E8 68 B3 21 A5 54 6A 20 D0 3F 38 21 E4 0E B0 E0 24 0A E3 64 BD 32<br>opmeh*!¥TŠ D?8!A.°Aô.ādv2   | CE 61 5D CD C6 24 4C FD 73 32 CF B2 AF AC E8 E2 E2 C9 6D 30 85 01 AD F5 F6 F1 B9 74 | 2 Îa]ÎEŞLýs2Ï <sup>-</sup> ¬èââÉm0Õöñ <sup>+</sup> r      |
| C9       62       B6       30       85       01       2C       77       5F       AA       A5       7F       6B       A8       E8       C2       3E       74       51       22       E1       AC       E0       DE       F2       34       F4       DA       Ébg0, w       *¥.k."èà>tQ"á→à>Q4ôÚ         97       23       8E       B3       DB       2D       AA       84       63       A4       77       92       C7       7F       A2       AO       7F       32       B5       BB       2D       AA       44       63       A4       77       92       C7       7F       A2       AO       7F       32       B5       68       B1       6E       21       65       3A       -#ź²>=#ź?=www'/c>.wce .3ź±źż!Li::         36       A9       5D       D       IF       E5       86       3D       36       5A       CD       CE       A2       37       F1       2B       7D       6F       73       7D       F4       7D       9D       F4       A4       7D       F4       A4       7D       F4       A4       7D       F4       A4       F4       F4   | B1 AC F0 B6 63 73 66 1B BB AA 34 9A F2 2C 9D E0 36 E1 A5 3E 32 1E 24 26 FE FD 85 09 | 9 ±¬ð¶csf.»⁴4šò,.à6á¥>2.\$&þý                             |
| 97 23 8E B3 BB 2D AA 84 63 A4 77 92 C7 3E 2E 77 47 A2 A0 7F 33 E9 66 B1 8E 21 85 3A<br>38 A9 5D 70 D0 EB 86 30 1F E5 86 3D 36 50 5A CD 4C EA 23 37 F1 2B 77 B6 F3 70 F4 72 60]pBĕt0.åt=627L±47ħ+w€ópór<br>BC E3 5A B4 BF A1 9E 72 B2 69 2C 77 F6 198 9A B3 57 AA F2 9C E7 E7 03 F3 E1 A2 6 44ã'; Lifri, wcără¥m²dœç, dáo6<br>58 FF 65 3B 7A E2 CF 73 E9 21 AA 85 72 47 9E E3 C4 2D A5 54 30 D8 FF 45 B5 AC E4 32<br>93 2E 12 F2 70 99 E8 68 B3 21 A5 54 6A 20 D0 3F 38 21 E4 0E B0 E0 24 0A E3 64 BD 32<br>opméh !¥TŠ D?8!a.°åč.äd+2  | FB 01 D0 5B B2 0F E0 EA B3 1A E7 16 C7 7F 62 77 47 20 E4 51 D9 29 28 7F 5F 70 4D 34 | 4 û.Đ[ʻ.àê'.ç.Ç.bwG äQÙ)(pM4                              |
| 38 A9 5D 70 D0 EB 86 30 1F ES 86 3D 36 50 5A CD 4C EA 23 37 F1 2B 77 B6 F3 70 F4 72 80]pDeT0.åt=6221L47ñ+w⊈ópór<br>BC E3 9A B4 BF A1 9E 72 B2 69 2C 77 BF 61 98 9A B3 57 AA F2 9C E7 E7 03 F3 E1 A2 36<br>4Åå ¿įžr <sup>4</sup> 1,µ¿a <sup>+</sup> šW <sup>+</sup> óœçç.óåc6<br>58 FF 65 3B 7A E2 CF 73 E9 21 AA 85 72 47 9E E3 C4 2D A5 54 30 D8 FF 45 B5 AC E4 2X Xye:zäľst <sup>4</sup> c5&ÅA=¥T00yEt,⊷A2<br>93 2E 12 F2 70 99 E8 68 B3 21 A5 54 6A 20 D0 3F 38 21 E4 0E B0 E0 24 0A E3 64 BD 32 "óp™eh*!¥TŠ Đ?8!A.°å&.åd∿2  | C9 62 B6 30 85 01 2C 77 5F AA A5 7F 6B A8 E8 C2 3E 74 51 22 E1 AC E0 DE F2 34 F4 D2 | A Éb¶0,w_*¥.k"èÂ>tQ"á¬à⊉ò4ôÚ                              |
| BC E3 9A B4 BF A1 9E 72 B2 69 2C 77 BF 61 98 9A B3 57 AA F2 9C E7 E7 03 F3 E1 A2 36 4ÅÅ <sup>*</sup> ;;žr <sup>4</sup> 1,w;a <sup>*</sup> 3 <sup>*</sup> W <sup>2</sup> dœçç.Óác6<br>58 FF 65 3B 7A E2 CF 73 E9 21 AA 85 72 47 9E E3 C4 2D A5 54 30 D8 FF 45 B5 AC E4 32 Xyerzáľsé!*_rGžãA <sup>+</sup> ¥TOØyEµ-A2<br>93 2E 12 F2 70 99 E8 68 B3 21 A5 54 8A 20 D0 3F 38 21 E4 0E B0 E0 24 0A E3 64 BD 32 "ôp <sup>m</sup> èh <sup>*</sup> !¥TŠ Đ?8!ä.*å§.ãd <sup>4</sup> s2  | 97 23 8E B3 BB 2D AA 84 63 A4 77 92 C7 3E 2E 77 47 A2 A0 7F 33 E9 66 B1 8E 21 85 33 | A -#޳»-*"C¤w'Ç>.wG¢ .3éf±Ž!:                              |
| 58 FF 65 3B 7A E2 CF 73 E9 21 AA 85 72 47 9E E3 C4 2D A5 54 30 D8 FF 45 B5 AC E4 32 Xye;záľsé!*_rGžāÄ-¥TOØyEu-a2<br>93 2E 12 F2 70 99 E8 68 B3 21 A5 54 8A 20 D0 3F 38 21 E4 0E B0 E0 24 0A E3 64 BD 32 "op=eh!¥TŠ B?8!a.*ač.ād+2   | 38 A9 5D 70 D0 EB 86 30 1F E5 86 3D 36 50 5A CD 4C EA 23 37 F1 2B 77 B6 F3 70 F4 72 | 2 8©]pĐë†0.åt=6PZÍLê#7ñ+w¶ópôr                            |
| 93 2E 12 F2 70 99 E8 68 B3 21 A5 54 8A 20 D0 3F 38 21 E4 0E B0 E0 24 0A E3 64 BD 32 "op™eh*!¥TŠ Đ?8!ä.*å\$.ād%2   | BC E3 9A B4 BF A1 9E 72 B2 69 2C 77 BF 61 98 9A B3 57 AA F2 9C E7 E7 03 F3 E1 A2 30 | 6 ¾ãš´¿¡žr°i,w¿a″š³W≇òœçç.óá¢6                            |
| RA 22 RE 20 RA 00 40 RA 10 AC 10 RA 10 AC RA 01 AO RA 00 AO RE 01 AO RA 21 AO 71 AO 11 A 11 A 11 BATTRAUTT TA ATLANĂT A   | 58 FF 65 3B 7A E2 CF 73 E9 21 AA 85 72 47 9E E3 C4 2D A5 54 30 D8 FF 45 B5 AC E4 32 | 2 Xýe;záïsé!*rGžãÄ-¥TOØýEµ-a2                             |
|   | 93 2E 12 F2 70 99 E8 68 B3 21 A5 54 8A 20 D0 3F 38 21 E4 0E B0 E0 24 0A E3 64 BD 32 |   |
|   |   |   |

Figure 1. HijackLoader key retrieving and decrypting (click to enlarge)

Following XOR decryption, the configuration undergoes decompression using the RtlDecompressBuffer API with COMPRESSION\_FORMAT\_LZNT1. After decompressing the configuration, the malware loads a legitimate Windows DLL specified in the configuration blob (in this sample, C:\Windows\SysW0W64\mshtml.dll).

The second-stage, position-independent shellcode retrieved from the configuration blob is written to the .text section of the newly loaded DLL before being executed. The HijackLoader second-stage, position-independent shellcode then performs some evasion activities (further detailed below) to bypass user mode hooks using Heaven's Gate and injects subsequent shellcode into cmd.exe.The injection of the third-stage shellcode is accomplished via a variation of process hollowing that results in an injected hollowed mshtml.dll into the newly spawned cmd.exe child process.

The third-stage shellcode implements a user mode hook bypass before injecting the final payload (a Cobalt Strike beacon for this sample) into the child process logagent.exe. The injection mechanism used by the third-stage shellcode leverages the following techniques:

- Process Doppelgänging Primitives: This technique is used to hollow a Transacted Section (mshtml.dll) in the remote process to contain the final payload.
- Process/DLL Hollowing: This technique is used to inject the fourth-stage shellcode that is responsible for performing evasion prior to passing execution to the final payload within the transacted section from the previous step.

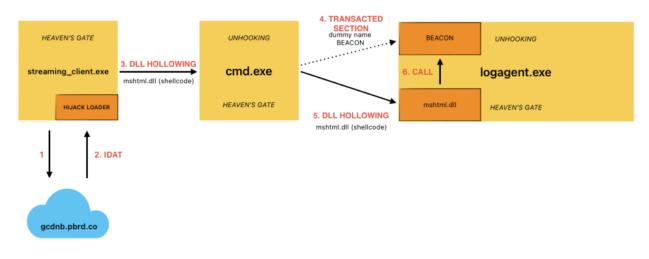


Figure 2 details the attack path exhibited by this HijackLoader variant.



## Main Evasion Techniques Used by HijackLoader and Shellcode

The primary evasion techniques employed by HijackLoader include hook bypass methods such as Heaven's Gate and unhooking by remapping system DLLs monitored by security products. Additionally, the malware implements variations of process hollowing and an injection technique that leverages transacted hollowing, which combines the transacted section and process doppelgänging techniques with DLL hollowing.

#### Hook Bypass: Heaven's Gate and Unhooking

Like other variants of HijackLoader, this sample implements a user mode hook bypass using Heaven's Gate (when run in SysW0W64) — this is similar to <u>existing</u> (x64\_Syscall function) implementations.

This implementation of Heaven's Gate is a powerful technique that leads to evading user mode hooks placed in SysW0W64 ntdll.dll by directly calling the syscall instruction in the x64 version of ntdll.

Each call to Heaven's Gate uses the following as arguments:

- The syscall number
- The number of parameters of the syscall
- The parameters (according to the syscall)

This variation of the shellcode incorporates an additional hook bypass mechanism to elude any user mode hooks that security products may have placed in the x64 ntdll. These hooks are typically used for monitoring both the x32 and x64 ntdll.

During this stage, the malware remaps the .text section of x64 ntdll by using Heaven's Gate to call NtWriteVirtualMemory and NtProtectVirtualMemory to replace the in-memory mapped ntdll with the .text from a fresh ntdll read from the file C:\windows\system32\ntdll.dll. This unhooking technique is also used on the process hosting the final Cobalt Strike payload (logagent.exe) in a final attempt to evade detection.

#### **Process Hollowing Variation**

To inject the subsequent shellcode into the child process cmd.exe, the malware utilizes common process hollowing techniques. This involves mapping the legitimate Windows DLL mshtml.dll into the target process and then replacing its .text section with shellcode. An additional step necessary to trigger the execution of the remote shellcode is detailed in a later section.

To set up the hollowing, the sample creates two pipes that are used to redirect the Standard Input and the Standard Output of the child process (specified in the aforementioned configuration blob, C:\windows\syswow64\cmd.exe) by placing the pipes' handles in a STARTUPINFOW structure spawned with CreateProcessW API.

One key distinction between this implementation and the typical "standard" process hollowing can be observed here: In standard process hollowing, the child process is usually created in a suspended state. In this case, the child is **not** explicitly created in a suspended state, making it appear less suspicious. Since the child process is waiting for an input from the pipe created previously, its execution is hanging on receiving data from it. Essentially, we can call this an interactive process hollowing variation.

As a result, the newly spawned cmd.exe will read input from the STDIN pipe, effectively waiting for new commands. At this point, its EIP (Extended Instruction Pointer) is directed toward the return from the NtReadFile syscall.

The following section details the steps taken by the second-stage shellcode to set up the child process cmd.exe ultimately used to perform the subsequent injections used to execute the final payload.

The parent process streaming\_client.exe initiates an NtDelayExecution to sleep, waiting for cmd.exe to finish loading. Afterward, it reads the legitimate Windows DLL mshtml.dll from the file system and proceeds to load this library into cmd.exe as a shared section. This is accomplished using the Heaven's Gate technique for:

- Creating a shared section object using NtCreateSection
- Mapping that section in the remote cmd.exe using NtMapViewOfSection

It then replaces the .text section of the mshtml DLL with malicious shellcode by using:

- Heaven's Gate to call NtProtectVirtualMemory on cmd.exe to set RWX permissions on the .text section of the previously mapped section mshtml.dll
- Heaven's Gate to call NtWriteVirtualMemory on the DLL's .text section to stomp the module and write the third-stage shellcode

Finally, to trigger the execution of the remote injected shellcode, the malware uses:

- Heaven's Gate to suspend (NtSuspendThread) the remote main thread
- A new CONTEXT (by using NtGetContextThread and NtSetContextThread) to modify the EIP to point to the previously written shellcode
- Heaven's Gate to resume (NtResumeThread) the remote main thread of cmd.exe

However, because cmd.exe is waiting for user input from the STDINPUT pipe, the injected shellcode in the new process isn't actually executed upon the resumption of the thread. The loader must take an additional step:

The parent process streaming\_client.exe needs to write (WriteFile) \r\n string to the STDINPUT pipe created previously to send an input to cmd.exe after calling NtResumeThread. This effectively resumes execution of the primary thread at the shellcode's entry point in the child process cmd.exe.

## Interactive Process Hollowing Variation: Tradecraft Analysis

We have successfully replicated the threadless process hollowing technique to understand how the pipes trigger it. Once the shellcode has been written as described, it needs to be activated. This activation is based on the concept that when a program makes a syscall, the thread waits for the kernel to return a value.

In essence, the interactive process hollowing technique involves the following steps:

- **CreateProcess:** This step involves spawning the cmd.exe process to inject the malicious code by redirecting STDIN and STDOUT to pipes. Notably, this process isn't suspended, making it appear less suspicious. Waiting to read input from the pipe, the NtReadFile syscall sets its main thread's state to Waiting and \_KWAIT\_REASON to Executive, signifying that it's awaiting the execution of kernel code operations and their return.
- WriteProcessMemory: This is where the shellcode is written into the cmd.exe child process.
- SetThreadContext: In this phase, the parent sets the conditions to redirect the execution flow of the cmd.exe child process to the previously written shellcode's address by modifying the EIP/RIP in the remote thread CONTEXT.
- WriteFile: Here, data is written to the STDIN pipe, sending an input to the cmd.exe process. This action resumes the execution of the child process from the NtReadFile operation, thus triggering the execution of the shellcode. Before returning to user space, the kernel is reading and restoring the values saved in the \_KTRAP\_FRAME structure (containing the EIP/RIP register value) to resume from where the syscall was called. By modifying the CONTEXT in the previous step, the loader hijacks the resuming of the execution toward the shellcode address without the need to suspend and resume the thread, which this technique usually requires.

#### Transacted Hollowing<sup>2</sup> (Transacted Section/Doppelgänger + Hollowing)

The malware writes the final payload in the child process logagent.exe spawned by the third-stage shellcode in cmd.exe by creating a transacted section to be mapped in the remote process. Subsequently, the malware injects fourth-stage shellcode into logagent.exe by loading and hollowing another instance of mshtml.dll into the target process. The injected fourth-stage shellcode performs the aforementioned hook bypass technique before executing the final payload previously allocated by the transacted section.

#### **Transacted Section Hollowing**

Similarly to process doppelgänging, the goal of a transacted section is to create a stealthy malicious section inside a remote process by overwriting the memory of the legitimate process with a transaction.

In this sample, the third-stage shellcode executed inside cmd.exe places a malicious transacted section used to host the final payload in the target child process logagent.exe. The shellcode uses the following:

- NtCreateTransaction to create a transaction
- RtlSetCurrentTransaction and CreateFileW with a dummy file name to replace the documented CreateFileTransactedW
- Heaven's Gate to call NtWriteFile in a loop, writing the final shellcode to the file in 1,024-byte chunks
- Creation of a section backed by that file (Heaven's Gate call NtCreateSection)
- A rollback of the previously created section by using Heaven's Gate to call NtRollbackTransaction

Existing similar implementations have publicly been observed in <u>this project</u> that implements transaction hollowing.

Once the transacted section has been created, the shellcode generates a function stub at runtime to hide from static analysis. This stub contains a call to the CreateProcessW API to spawn a suspended child process logagent.exe

(c50bffbef786eb689358c63fc0585792d174c5e281499f12035afa1ce2ce19c8) that was previously dropped by cmd.exe under the %TEMP% folder.

After the target process has been created, the sample uses Heaven's Gate to:

- Read its PEB by calling NtReadVirtualMemory to retrieve its base address (0x400000)
- Unmap the logagent.exe image in the logagent.exe process by using NtUnMapViewofSection
- Hollow the previously created transacted section inside the remote process by remapping the section at the same base address (0x400000) with NtMapViewofSection

#### **Process Hollowing**

After the third-stage shellcode within cmd.exe injects the final Cobalt Strike payload inside the transacted section of the logagent.exe process, it continues by process hollowing the target process to write the fourth shellcode stage ultimately used to execute the final payload (loaded in the transacted section) in the remote process. The third-stage shellcode maps the legitimate Windows DLL C:\Windows\SysWOW64\mshtml.dll in the target process prior to replacing its .text with the fourth-stage shellcode and executing it via NtResumeThread.

This additional fourth-stage shellcode written to logagent.exe performs similar evasion activities to the third-stage shellcode executed in cmd.exe (as indicated in the hook bypass section) before passing execution to the final payload.

# CrowdStrike Falcon Coverage

CrowdStrike employs a layered approach for malware detection using machine learning and indicators of attack (IOAs). As shown in Figure 3, the CrowdStrike Falcon® sensor's machine learning capabilities can automatically detect and prevent HijackLoader in the initial stages of the attack chain; i.e., as soon as the malware is downloaded onto the victim's machine. Behavior-based detection capabilities (IOAs) can recognize malicious behavior at various stages of the attack chain, including when employing tactics like process injection attempts.

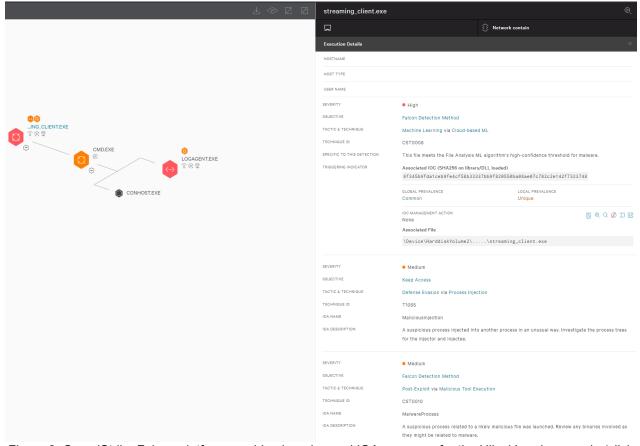


Figure 3. CrowdStrike Falcon platform machine learning and IOA coverage for the HijackLoader sample (click to enlarge)

# Indicators of Compromise (IOCs)

| File                 | SHA256  |
|----------------------|---|
| streaming_client.exe | <u>6f345b9fda1ceb9fe4cf58b33337bb9f820550ba08ae07c782c2e142f7323748</u> |

## **MITRE ATT&CK Framework**

The following table maps reported HijackLoader tactics, techniques and procedures (TTPs) to the MITRE ATT&CK® framework.

| ID        | Technique  | Description  |
|-----------|--|--|
| T1204.002 | User Execution:<br>Malicious File                                | The sample is a backdoored version of <pre>streaming_client.exe</pre> , with the Entry Point redirected to a malicious stub.   |
| T1027.007 | Obfuscated Files or<br>Information:<br>Dynamic API<br>Resolution | HijackLoader and its stages hide some of the important<br>imports from the IAT by dynamically retrieving kernel32 and<br>ntdll API addresses. It does this by parsing PEB-<br>>PEB_LDR_DATA and retrieving the function addresses. |

| T1016.001 | System Network<br>Configuration<br>Discovery: Internet<br>Connection<br>Discovery | This variant of HijackLoader connects to a remote server to check if the machine is connected to the internet by using the WinHttp API (WinHttpOpenRequest and WinHttpSendRequest).  |
|-----------|---|--|
| T1140     | Deobfuscate/Decode<br>Files or Information  | HijackLoader utilizes XOR mechanisms to decrypt the downloaded stage.  |
| T1140     | Deobfuscate/Decode<br>Files or Information  | HijackLoader utilizes RtlDecompressBuffer to LZ decompress the downloaded stage.   |
| T1027     | Obfuscated Files or<br>Information  | HijackLoader drops XOR encrypted files to the %APPDATA% subfolders to store the downloaded stages.   |
| T1620     | Reflective Code<br>Loading  | HijackLoader reflectively loads the downloaded shellcode in<br>the running process by loading and stomping the mshtml.dll<br>module using the LoadLibraryW and VirtualProtect APIs.  |
| T1106     | Native API  | HijackLoader uses direct syscalls and the following APIs to<br>perform bypasses and injections: WriteFileW, ReadFile,<br>CreateFileW, LoadLibraryW, GetProcAddress,<br>NtDelayExecution, RtlDecompressBuffer, CreateProcessW,<br>GetModuleHandleW, CopyFileW, VirtualProtect,<br>NtProtectVirtualMemory, NtWriteVirtualMemory,<br>NtResumeThread, NtSuspendThread, NtGetContextThread,<br>NtSetContextThread, NtCreateTransaction,<br>RtlSetCurrentTransaction, NtRollbackTransaction,<br>NtCreateSection, NtMapViewOfSection,<br>NtUnMapViewOfSection, NtWriteFile, NtReadFile,<br>NtCreateFile and CreatePipe. |
| T1562.001 | Impair Defenses:<br>Disable or Modify<br>Tools                                    | HijackLoader and its stages use Heaven's Gate and remap x64 ntdll to bypass user space hooks.  |
| T1055.012 | Process Injection:<br>Process Hollowing   | HijackLoader and its stages implement a process hollowing technique variation to inject in cmd.exe and logagent.exe.   |
| T1055.013 | Process Injection:<br>Process<br>Doppelgänging                                    | The HijackLoader shellcode implements a process doppelgänging technique variation (transacted section hollowing) to load the final stage in logagent.exe.  |
|           |   |  |

#### **Additional Resources**

- The CrowdStrike Falcon® platform achieved 100% protection, 100% visibility and 100% analytic detection across all steps in the MITRE Engenuity ATT&CK® Evaluations: Enterprise, Round 5. Learn more in this blog post.
- CrowdStrike was named a Leader in the 2023 Gartner® Magic Quadrant<sup>™</sup> for Endpoint Protection Platforms — furthest right in Vision and highest in Ability to Execute. <u>Read about it</u> <u>here.</u>
- Find out how the Falcon platform stops breaches, saves time and saves money in this IDC analysis: <u>The Business Value of the CrowdStrike Falcon XDR Platform</u>.
- See the Falcon platform in action <u>sign up for a free demo</u> today.