{"payload":{"allShortcutsEnabled":false,"fileTree":{"FormBook":{"items":[{"name":"img","path":"FormBook/img","contentType":"directory"},
{"name":"FormBook.md","path":"FormBook/FormBook.md","contentType":"file"}],"totalCount":2},"":{"items":
[{"name":"Brbbot","path":"Brbbot","contentType":"directory"},{"name":"Conficker","path":"Conficker","contentType":"directory"},
{"name":"FormBook","path":"FormBook","contentType":"directory"},{"name":"HawkEye","path":"HawkEye","contentType":"directory"},
{"name":"Kovter","path":"Kovter","contentType":"directory"},{"name":"Remcos","path":"Remcos","contentType":"directory"},
{"name":"RevengeRAT","path":"RevengeRAT","contentType":"directory"},
{"name":"README.md","path":"README.md","contentType":"file"}],"totalCount":8}},"fileTreeProcessingTime":8.489944,"foldersToFetch":
[],"reducedMotionEnabled":null,"repo":{"id":453063504,"defaultBranch":"main","name":"malware-analysis-
writeups","ownerLogin":"itaymigdal","currentUserCanPush":false,"isFork":false,"isEmpty":false,"createdAt":"2022-01-
28T12:42:46.000Z","ownerAvatar":"https://avatars.githubusercontent.com/u/32266629?
v=4","public":true,"private":false,"isOrgOwned":false},"refInfo":
{"name":"main","listCacheKey":"v0:1688151375.0","canEdit":false,"refType":"branch","currentOid":"767ea5a62567ea056e890319bc85c5227a679c
{"rawLines":null,"stylingDirectives":null,"csv":null,"csvError":null,"dependabotInfo":
{"showConfigurationBanner":false,"configFilePath":null,"networkDependabotPath":"/itaymigdal/malware-analysis-
writeups/network/updates","dismissConfigurationNoticePath":"/settings/dismiss-
notice/dependabot_configuration_notice","configurationNoticeDismissed":null,"repoAlertsPath":"/itaymigdal/malware-analysis-
writeups/security/dependabot","repoSecurityAndAnalysisPath":"/itaymigdal/malware-analysis-
writeups/settings/security_analysis","repoOwnerIsOrg":false,"currentUserCanAdminRepo":false},"displayName":"FormBook.md","displayUrl":"https
analysis-writeups/blob/main/FormBook/FormBook.md?raw=true","headerInfo":{"blobSize":"5.37 KB","deleteInfo":{"deleteTooltip":"You must be
signed in to make or propose changes"},"editInfo":{"editTooltip":"You must be signed in to make or propose
changes"},"ghDesktopPath":"https://desktop.github.com","gitLfsPath":null,"onBranch":true,"shortPath":"86693c4","siteNavLoginPath":"/login?
return_to=https%3A%2F%2Fgithub.com%2Fitaymigdal%2Fmalware-analysis-
writeups%2Fblob%2Fmain%2FFormBook%2FFormBook.md","isCSV":false,"isRichtext":true,"toc":[{"level":1,"text":"Unpacking
FormBook","anchor":"unpacking-formbook","htmlText":"Unpacking FormBook"},{"level":2,"text":"Analysis process","anchor":"analysis-
process","htmlText":"Analysis process"}],"lineInfo":
{"truncatedLoc":"292","truncatedSloc":"243"},"mode":"file"},"image":false,"isCodeownersFile":null,"isValidLegacyIssueTemplate":false,"issueTempla
issue-and-pull-request-
templates","issueTemplate":null,"discussionTemplate":null,"language":"Markdown","large":false,"loggedIn":false,"newDiscussionPath":"/itaymigdal/r
analysis-writeups/discussions/new","newIssuePath":"/itaymigdal/malware-analysis-writeups/issues/new","planSupportInfo":
{"repoIsFork":null,"repoOwnedByCurrentUser":null,"requestFullPath":"/itaymigdal/malware-analysis-
writeups/blob/main/FormBook/FormBook.md","showFreeOrgGatedFeatureMessage":null,"showPlanSupportBanner":null,"upgradeDataAttributes":r
{"dismissActionNoticePath":"/settings/dismiss-notice/publish_action_from_dockerfile","dismissStackNoticePath":"/settings/dismiss-
notice/publish_stack_from_file","releasePath":"/itaymigdal/malware-analysis-writeups/releases/new?
marketplace=true","showPublishActionBanner":false,"showPublishStackBanner":false},"renderImageOrRaw":false,"richText":"

Unpacking FormBook

\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n

Malware Name	File Type	SHA256
FormBook	x32 exe	9B11FA3CFA0ACDD01BE3595FBA22F7B38C333E7EC8DA88228C971735913BB6F7

\n

Analysis process

\n
The initial file is a 32 bit Nullsoft installer (NSIS). Right-clicking and unzipping will successfully extract the installer files.

\n


\n
I have read in some blog posts before that usually NSIS installers contain also an installer script, that was missing in that case.\nSince the other two files apart from the DLL have unknown format and extension, I started to analyze the DLL.

\n


\n

This is a quite simple 32 bit DLL file that contains 2 exports. the first one was empty, I then decompiled the second one and performed some renaming and cleaning:

\n


\n
This function locates one of the other extracted files `hicclc.io` (that supposes to be dropped to `%TEMP%` by the NSIS installer by now), allocates some memory, copy the file content inside, and doing a simple xor loop with a hardcoded key to decode the next stage. then it simply calls that, which means that this is a shellcode.

\n
I wrote a little python script to decode this stage:

\n

```
i = 0\nkey = \"584058684148\"\ndecoded = b\"\"\n\nwith open(\"hicclc.io\", \"rb\") as f:\n    encoded = f.read()\n\nwhile i < 0x15e6:\n    decoded_byte = encoded[i] ^ ord(key[i % 0xc])\n    decoded += bytes([decoded_byte])\n    i += 1\n    \nwith open(\"shellcode.bin\", \"wb\") as f:\n    f.write(decoded)    \n
```

\n
Here I used BlobRunner to debug the shellcode:

\n


\n
The shellcode mission is to load the 3'rd extracted file which is the 3'rd unpacking stage:

\n


\n
Following this flow carefully in the debugger:

\n


\n
Capturing `VirtualAlloc` return address to keep tracking of the decoded stage:

\n


\n
Then we can see how `ReadFile` is filling this memory with the file content:

\n


\n
Vwalla!

\n


\n
The following graph snippet contains a loop in the left block for decoding this stage. I located a BP on the completion of the loop in the right block and let it run:

\n


\n
And now we love what we see!

\n
Let's dump this PE to disk:

\n

\n
When keeping debug the shellcode we observe that it spawns another instance of itself:

\n


\n


\n
Then it maps a fresh copy of NTDLL to evade EDR hooks in the original NTDLL:

\n


\n
Then it performs process hollowing by replacing the child process image with the PE we dumped before, some incriminating process hollowing calls are:

\n


\n


\n


\n
Observing the final PE payload, reveals that it's highly obfuscated and difficult to analyze. It's written in pure assembly using MASM:

\n


\n
It contains only `.text` section, and Cutter really struggled to create the navigation bar above due to obfuscation and non-conventional code:

\n


\n
The final payload is really tough, and contains lots of anti-vm and anti-debugging tricks, that I won't cover today (or any other time 😜)

\n
By using FLOSS, I managed to extract low-hanging fruits such as interesting stack strings, without delving too much into:

\n

```
FLOSS extracted 180
stackstrings\nDEST\nversion.dll\no%%Jr..\\$\nInternetCloseHandle\n[System]\nNT\\CurrentVersion\nencrypted_key\nc!!B0\nAAIA\n:.6$\n!
H88p\nPOST\n.exe\nwindir\n.exe\nProgram Files\nB>>|\n[Enter]\nimage/jpeg\n\\DB1\nInternet
Explorer\\IntelliForms\\Storage2\nHost:\nURL:\nHttpOpenRequestA\ncl.ini\nPassword\nri.ini\nPATH\nClipboard\npass\nhttpRealm\nPATH\n
ProductName\nUser :\n__Vault\nXhHp\nFBIMG\n\\INetCookies\nwindir\nbrowser\nOpera\nAut:\n[v)C\n\" /V\n\\explorer.exe\nG==z\n11#?
*0\ncB@\"\n\\Main\nf\"\"D~**T\n[<-Del]\nPass:\nlogin\nP00`\nWA      t\nWindows
Explorer\nimage/png\nx86\n7CbI\n7A@@(\nQSeA~\nl$$H\nim.jpeg\nguid\nId:\nY77n\nurlmon.dll\n,4$8_@\nrv.ini\nU33f\nInternetReadFile\n[
Esc]\nInternetConnectA\nauth\n_2016\\\nAA&\nProgramFiles\n2N H\nFBNG:\ni''N\nhostname\nprofiles.ini\n+Q0$\nI<(A\n=j&&LZ66lA??
~\nwww.\ng0+]C\n)w--Z\nWindows
Explorer\n_jbF~T\nrg.ini\nChrome\nim.jpeg\n2008\nABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/\nrc.ini\n.dll\n}+
+V\n!tX)i\n}{))R>\n2012\nProgramFiles\nx((Pz\nx64\nACAA\nUSERNAME\nt,,X.\n`  @\ngK99r\nPass:\n='9-6d\n_55j\nAaAA\nMS-WAPI-
\nCurrentVersion\nt\\lHBW\nAPI-\nLocal State\n2016\nIexplor\n.sqlite\nServer:\nUnknown\nD<<x\nwininet.dll\naAAWindows
Explorer\nName:\nOutlook Recovery\nV22dN::t\n00.ini\nThunderbird\\\n\\explorer.exe\nAaAAt\n\\Cookies\nSniff
from:\nHttpSendRequestA\n.zip\ne##F^\n\\44h\nM;;va\nS11b?\n\\Opera Software\\Opera Stable\\Login
Data\nlpHP\naiKwZ\nlog.ini\nuser\n\\Microsoft\\Windows\nchrome_child.dll\nopen\n\\Low\n200
OK\nAaAA\nFirefox\\\n.exe\nUser:\nInternetOpenA\nFirefox\nInstall Directory\nRecovery\n\\Current Session\nProgramFiles\nURL:\nUser-
Agent:\nUrl:\nhc 82\nAEFA\n@J7<\n;fD4~\n[Tab]\n\\Firefox\nPort:\nHost:\nUnknown\n.dll\nUser-Agent:\nq//^\nFirefox\\\n\"
/V\nUnknown\nNrZl\n
```

\n
The final payload is waiting for reversers better than me :)

\n","renderedFileInfo":null,"tabSize":8,"topBannersInfo":
{"overridingGlobalFundingFile":false,"globalPreferredFundingPath":null,"repoOwner":"itaymigdal","repoName":"malware-analysis-writeups","showInvalidCitationWarning":false,"citationHelpUrl":"https://docs.github.com/en/github/creating-cloning-and-archiving-repositories/creating-a-repository-on-github/about-citation-files","showDependabotConfigurationBanner":false,"actionsOnboardingTip":null},"truncated":false,"viewable":true,"workflowRedirectUrl":null,"symb
{"timedOut":false,"notAnalyzed":true,"symbols":[]}},"copilotAccessInfo":null,"csrf_tokens":{"/itaymigdal/malware-analysis-writeups/branches":
{"post":"IaOE8p6wujaijW5f1ZE3YW1iNDzmPJClza1b1YPuV75gz1hL1Rp54ywv0_lsRF8Z3Rq-8_htunj02R1Rwlj7RQ"}}},"title":"malware-analysis-writeups/FormBook/FormBook.md at main · itaymigdal/malware-analysis-writeups","locale":"en"}