The return of the BOM

securelist.com/the-return-of-the-bom/90065/

Research

Research

28 Mar 2019

minute read

Authors

Expert GReAT

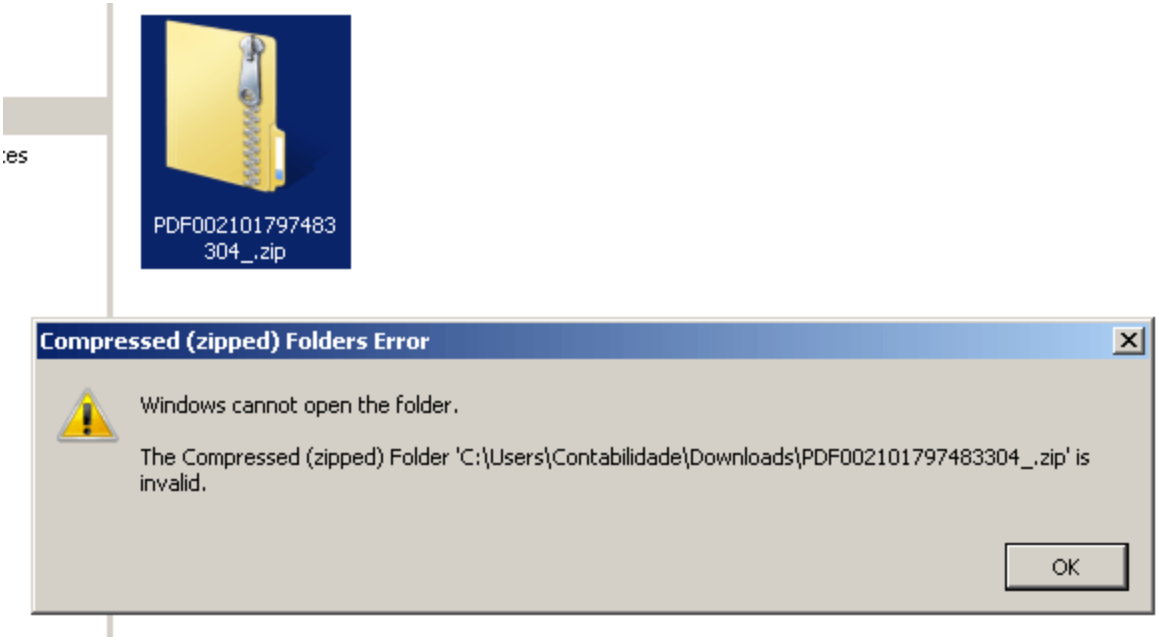
Because sometimes you can't teach an old malware developer new tricks

There’s nothing new in Brazilian cybercriminals trying out new ways to stay under the radar. It’s just that this time around the bad guys have started using a method that was reported in the wild years ago.

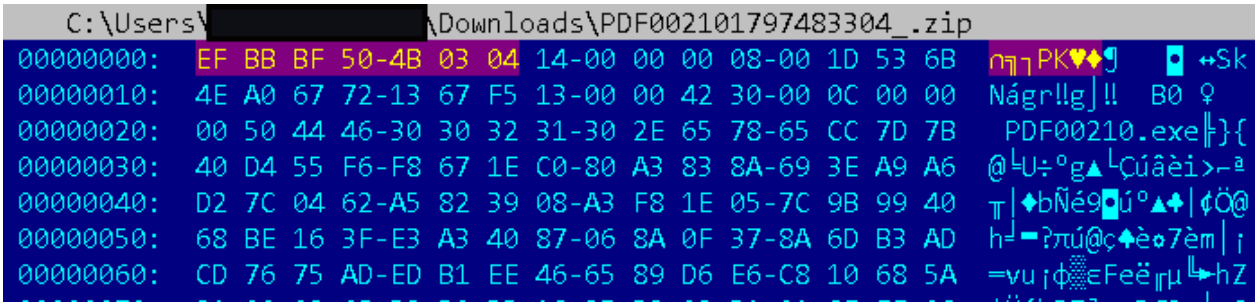
Russian gangs used this technique to distribute malware capable of modifying the hosts file on Windows systems. Published by McAfee in 2013, the UTF-8 BOM (Byte Order Mark) additional bytes helped these malicious crews avoid detection.

Since these campaigns depended on spear phishing to increase the victim count, the challenge was to fool email scanners and use a seemingly corrupted file that lands in the victim’s inbox.

The first indicator appears when the user tries to open the ZIP file with the default file explorer and sees the following error:



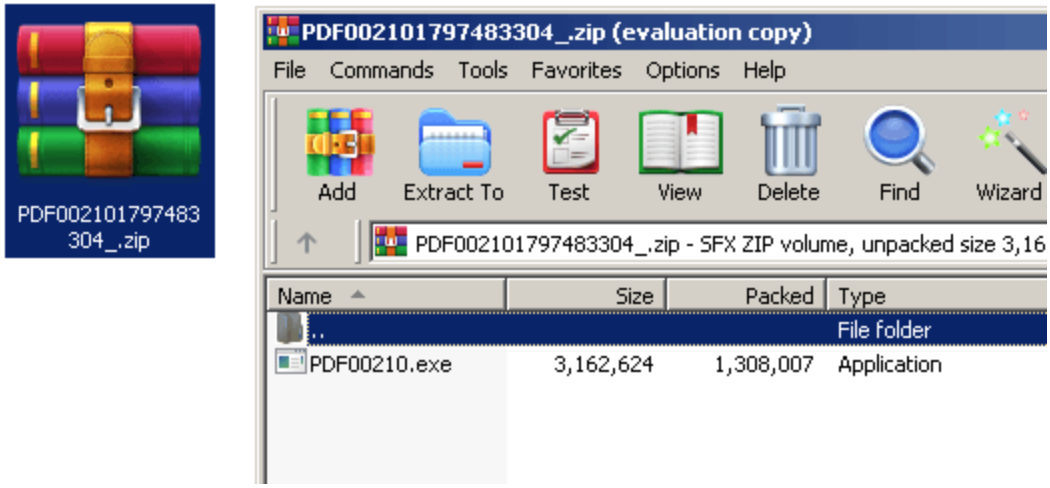
The error message suggests the file is corrupt, but when we check its contents we see something strange in there.



Zip header prefixed by UTF-8 BOM

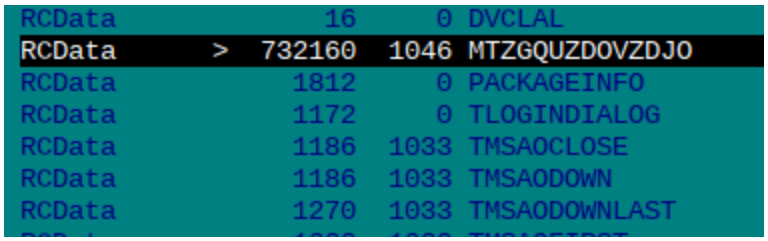
Instead of having the normal ZIP header starting with the “PK” signature (0x504B), we have three extra bytes (0xEFBBBF) that represent the Byte Order Mark (BOM) usually found within UTF-8 text files. Some tools will not recognize this file as being a ZIP archive format, but will instead recognize it as an UTF-8 text file and fail to extract the malicious payload.

However, utilities such as WinRAR and 7-Zip ignore this data and extract the content correctly. Once the user extracts the file with any of these utilities they can execute it and infect the system.

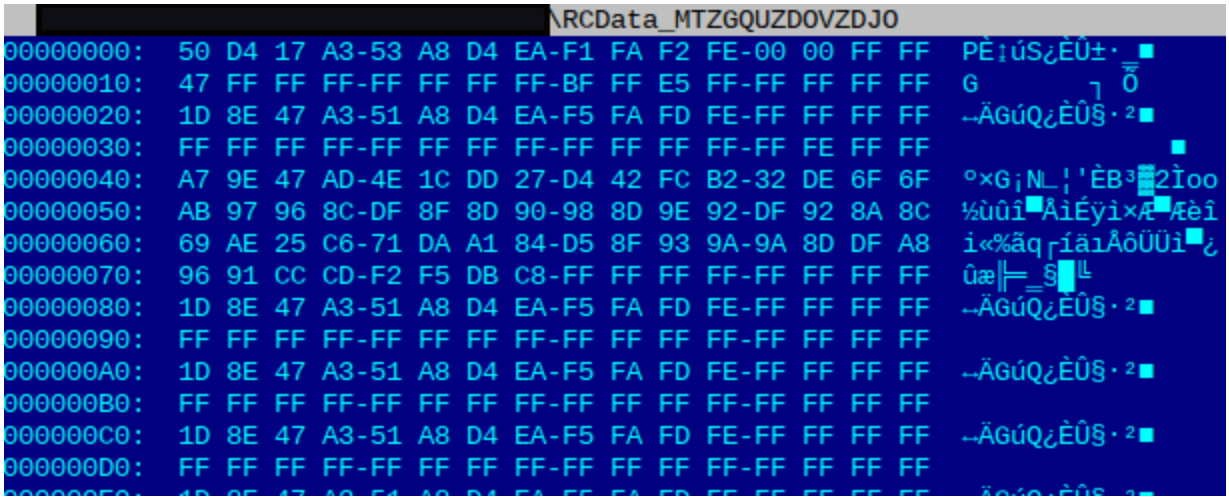


The file is successfully extracted by WinRAR

The malicious executable acts as a loader for the main payload which is embedded in the resource section.



Resource table showing the resource containing the encrypted data



Encrypted DLL stored in resource section

The content stored inside the resource, encrypted with a XOR-based algorithm, is commonly seen in different malware samples from Brazil. The decrypted resource is a DLL that will load and execute the exported function “BICDAT”.

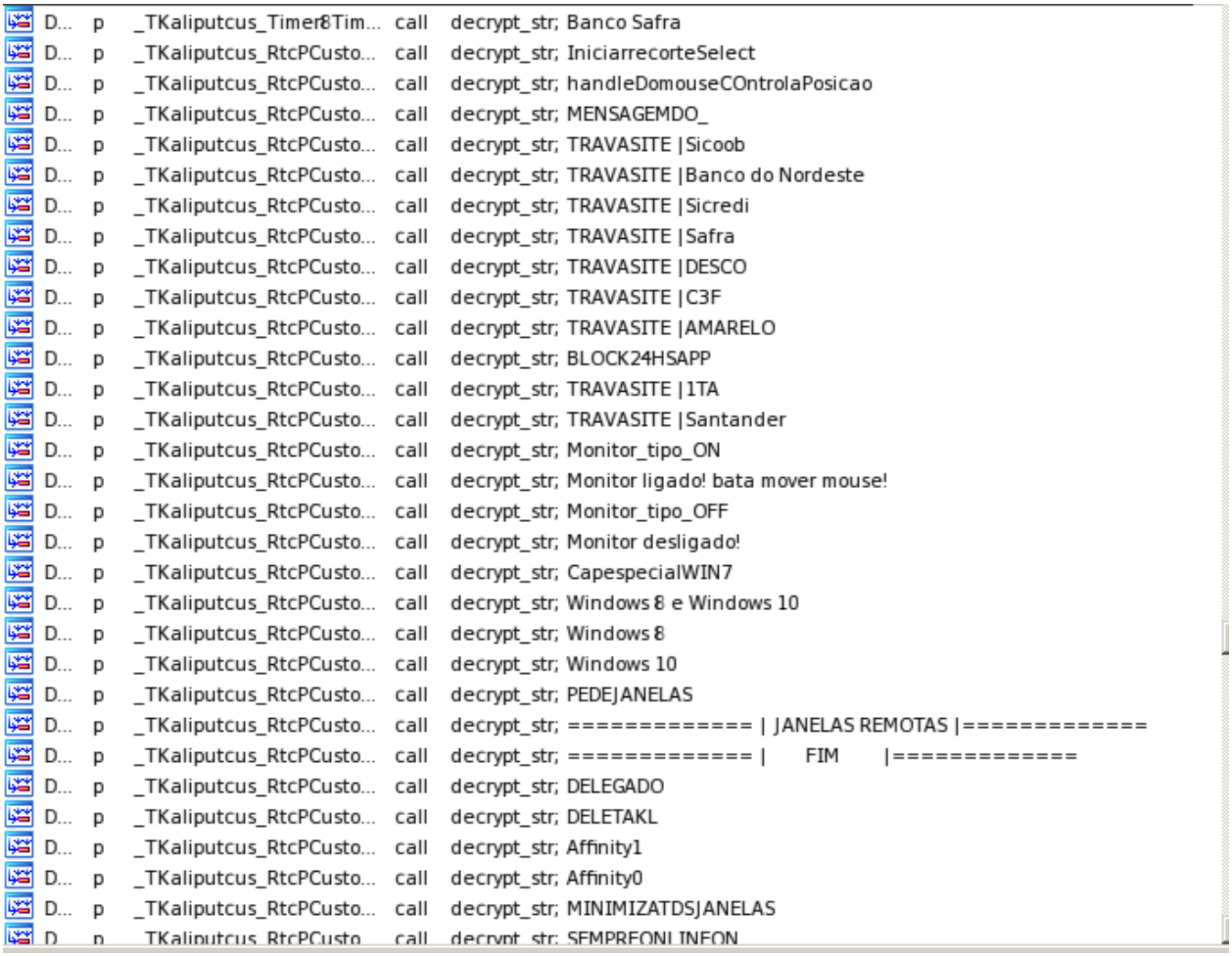
```
  dll_image = (int)load_code(&dword_516F84, dword_516F88, dword_516F88 >> 31);
  HIDWORD(v19) = &savedregs;
  LODWORD(v19) = &loc_476801;
  v18 = __readfsdword(0);
  __writefsdword(0, (unsigned int)&v18);
  if ( !dll_image )
    Sysutils::Abort();
  p_BICDAT = (int (__cdecl *)(_DWORD))get_func_addr((int)&dll_image, "BICDAT");
  if ( !p_BICDAT )
    Sysutils::Abort();
  v17 = a6;
  v16 = System::__linkproc__ LStrToPChar(a5);
  v15 = System::__linkproc__ LStrToPChar(a4);
  System::__linkproc__ LStrToPChar(v25);
  System::__linkproc__ LStrToPChar(v26);
  args = System::__linkproc__ LStrToPChar(v27);
  p_BICDAT(args);
  __writefsdword(0, v14);
  if ( dll_image )
    unload_image(&dll_image);
}
```

Code used to load the extracted DLL and execute the exported function BICDAT

This library will then download a second stage payload which is a password-protected ZIP file and encrypted with the same function as the embedded payload. After extracting all the files, the loader will then launch the main executable.

```
download_file_0(v21, v63);
unknown_libname_907(*(Forms::TApplication **)off_4A2754[0]);
Sysutils::GetEnvironmentVariable((const int)&str_TEMP[1], &v58);
System::__linkproc__ LStrCatN(&v59, 3, v12, &str___28[1], dword_4A4488);
v13 = (_BYTE *)System::__linkproc__ LStrToPChar(v59);
unknown_libname_70((int)&v60, v13);
v22 = v60;
Sysutils::GetEnvironmentVariable((const int)&str_TEMP[1], &v55);
System::__linkproc__ LStrCatN(&v56, 3, v14, &str___28[1], dword_4A4488);
v15 = (_BYTE *)System::__linkproc__ LStrToPChar(v56);
unknown_libname_70((int)&v57, v15);
decrypt_file(v57, v22);
unknown_libname_907(*(Forms::TApplication **)off_4A2754[0]);
unknown_libname_907(*(Forms::TApplication **)off_4A2754[0]);
Sleep(0x280Du);
Sysutils::GetEnvironmentVariable((const int)&str_TEMP[1], &v52);
System::__linkproc__ LStrCatN(&v53, 3, v16, &str___28[1], dword_4A4488);
v17 = (_BYTE *)System::__linkproc__ LStrToPChar(v53);
unknown_libname_70((int)&v54, v17);
v23 = v54;
get_programfiles_dir(&v51);
System::__linkproc__ LStrCat(&v51, dword_4A4494);
extract_zip(v23, v51, &str____[1]);
Sleep(0x2F79u);
unknown_libname_907(*(Forms::TApplication **)off_4A2754[0]);
get_programfiles_dir(&v49);
System::__linkproc__ LStrCat(&v49, dword_4A4494);
read_ini(v49, &v50);
Sleep(0x1039u);
get_programfiles_dir(&v47);
System::__linkproc__ LStrCat(&v47, dword_4A4494);
sub_498188(v47, &v48);
Sleep(0x870u);
get_programfiles_dir(&v46);
System::__linkproc__ LStrCat(&v46, dword_4A4494);
run_executable(v46);
unknown_libname_907(*(Forms::TApplication **)off_4A2754[0]);
Sleep(0x1528u);
Sysutils::GetEnvironmentVariable((const int)&str_TEMP[1], &v44);
System::__linkproc__ LStrCatN(&v45, 3, v18, &str___28[1], dword_4A4488);
lpFileName = (const CHAR *)System::__linkproc__ LStrToPChar(v45);
DeleteFileA(lpFileName);
Sleep(0x1528u);
```

Code executed by BICDAT function



Strings related to Banking RAT malware

The final payload that's delivered is a variant of a Banking RAT malware, which is currently widespread in Brazil and Chile.

Kaspersky Lab products can extract and analyze compressed ZIP files containing the Byte Order Mark without any problem.

Indicators of compromise

087b2d745bc21cb1ab7feb6d3284637d
3f910715141a5bb01e082d7b940b3552
60ce805287c359d58e9afd90c308fcc8
c029b69a370e1f7b3145669f6e9399e5

- Malware Technologies
- RAT Trojan
- Spear phishing
- Trojan Banker

Authors

Expert GReAT

The return of the BOM

Your email address will not be published. Required fields are marked *

GReAT webinars

13 May 2021, 1:00pm

GReAT Ideas. Balalaika Edition

26 Feb 2021, 12:00pm
17 Jun 2020, 1:00pm
26 Aug 2020, 2:00pm
22 Jul 2020, 2:00pm
From the same authors

APT trends report Q2 2021

Arrests of members of Tetrade seed groups Grandoreiro and Melcoz

Ferocious Kitten: 6 years of covert surveillance in Iran

Bizarro banking Trojan expands its attacks to Europe

APT trends report Q1 2021

Subscribe to our weekly e-mails

The hottest research right in your inbox

-

Reports

APT trends report Q1 2022

This is our latest summary of advanced persistent threat (APT) activity, focusing on events that we observed during Q1 2022.

Lazarus Trojanized DeFi app for delivering malware

We recently discovered a Trojanized DeFi application that was compiled in November 2021. This application contains a legitimate program called DeFi Wallet that saves and manages a cryptocurrency wallet, but also implants a full-featured backdoor.

MoonBounce: the dark side of UEFI firmware

At the end of 2021, we inspected UEFI firmware that was tampered with to embed a malicious code we dub MoonBounce. In this report we describe how the MoonBounce implant works and how it is connected to APT41.

The BlueNoroff cryptocurrency hunt is still on

It appears that BlueNoroff shifted focus from hitting banks and SWIFT-connected servers to solely cryptocurrency businesses as the main source of the group's illegal income.

-
-
-