Putting data in Alternate data streams and how to execute it – part 2

Published: 2018-04-11 · Archived: 2026-04-05 13:39:09 UTC

I wrote a blogpost a while back about Alternate data streams that you can find here: https://oddvar.moe/2018/01/14/putting-data-in-alternate-data-streams-and-how-to-execute-it/

After I wrote that post I have made some new discoveries that I wanted to share around Alternate data streams. As you probably already know if you read some of my stuff is that I am a big fan of Living off the land techniques.

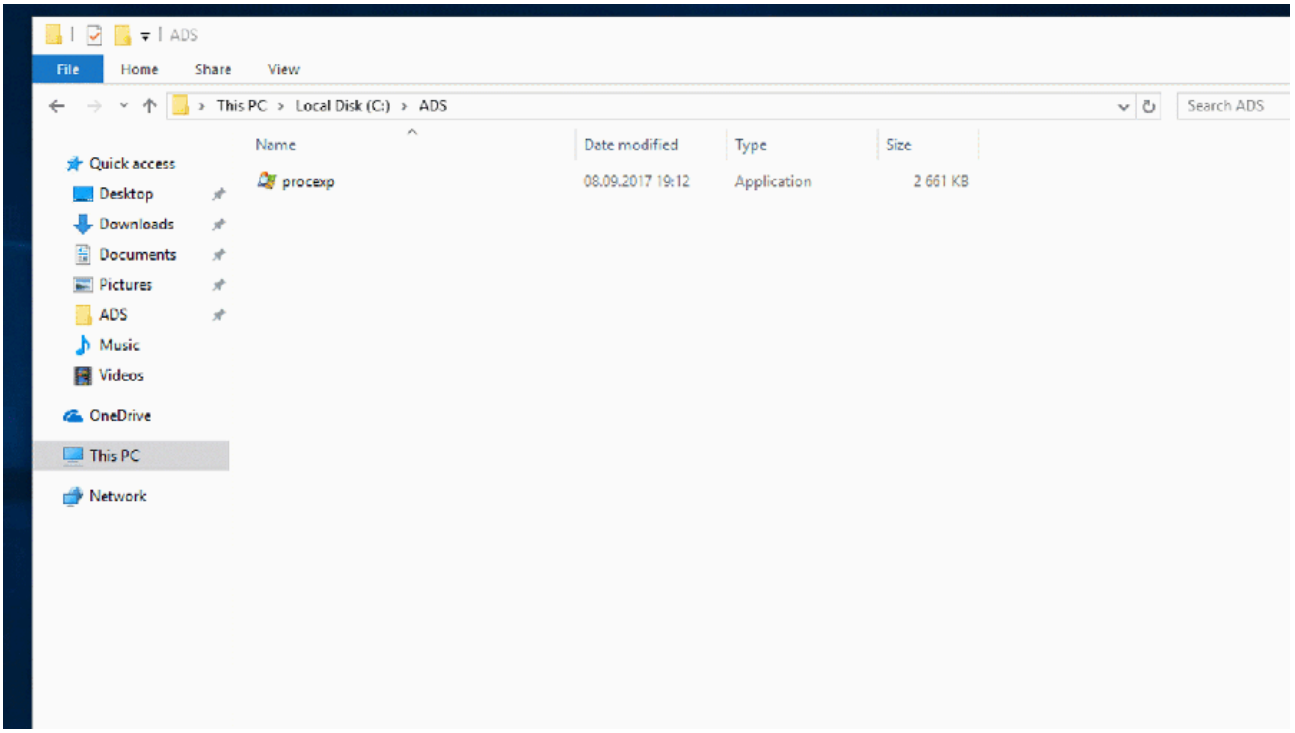
The only method I knew about to inject data into a alternate data stream when I wrote the first post was the “type” command.
I have since my last blogpost discovered some other techniques as well. These techniques I have discovered can of course have been discovered by others and already been blogged about, if so please let me know and I will link to your blogpost.

EXTRAC32.EXE

First up is extrac32. If do not know this command you can read more about it here: https://ss64.com/nt/extract.html

Basically what you use it for is to extract cab files. What I figured out was that you also can use this command to add alternate data streams. The PoC for doing this (including creating a CAB) looks like this:

```
echo "empty file" > c:\ADS\file.txt
makecab c:\ADS\procexp.exe c:\ADS\procexp.cab
extrac32 C:\ADS\procexp.cab c:\ADS\file.txt:procexp.exe
wmic process call create '"c:\ADS\file.txt:procexp.exe"'
```



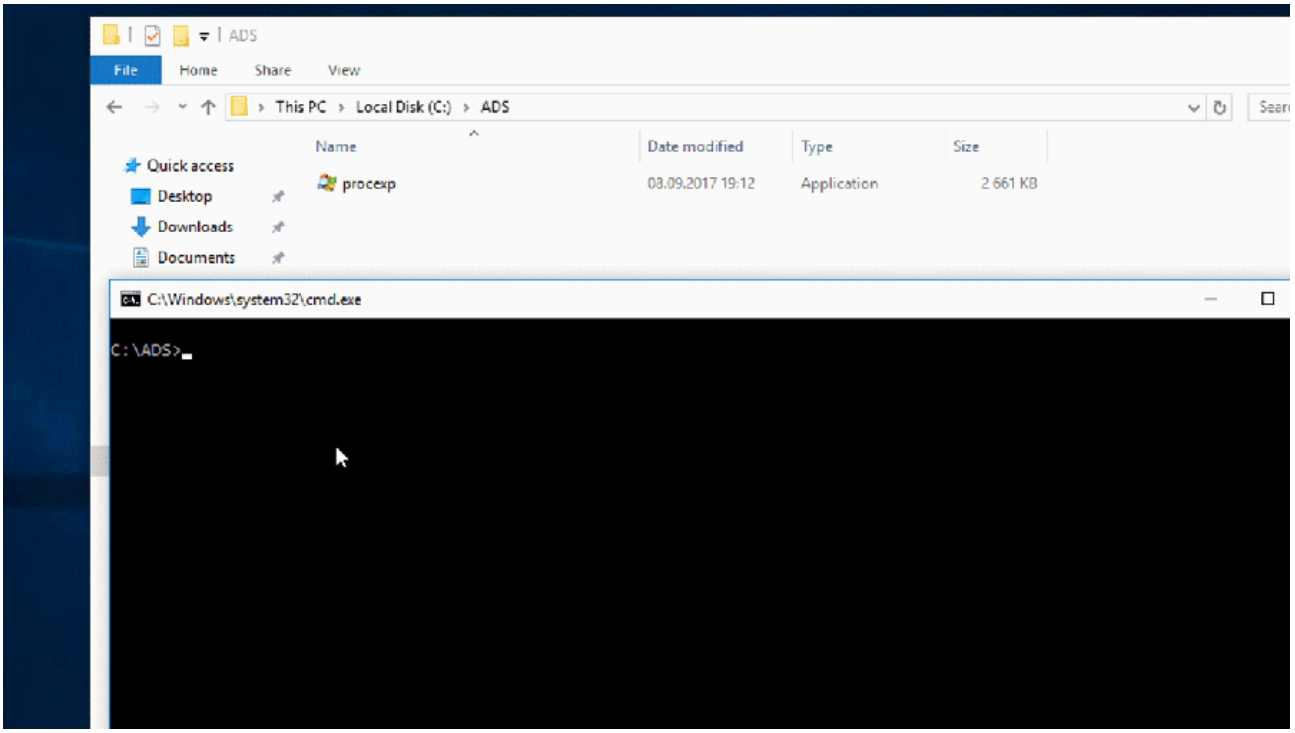
FINDSTR.EXE

Also in my research I found that Findstr can also be used to inject a payload into another file as an ADS stream. Findstr.exe is basically a command you use to find strings within files.
More about the binary here: https://ss64.com/nt/findstr.html

The cool thing I figured out was that you can search for a string that does not exist in a file and pipe that into a new file. And the cool thing is that it does allow it to be piped into a ADS stream of a file. It looks like this:

```
echo "empty file" > c:\ADS\file.txt
findstr /V /L W3AllLov3DonaldTrump c:\ADS\procexp.exe > c:\ADS\file.txt:procexp.exe
wmic process call create '"c:\ADS\file.txt:procexp.exe"'
```

The /V in the findstr command makes sure that everything that does not match the string I am searching for is showed. 🙂

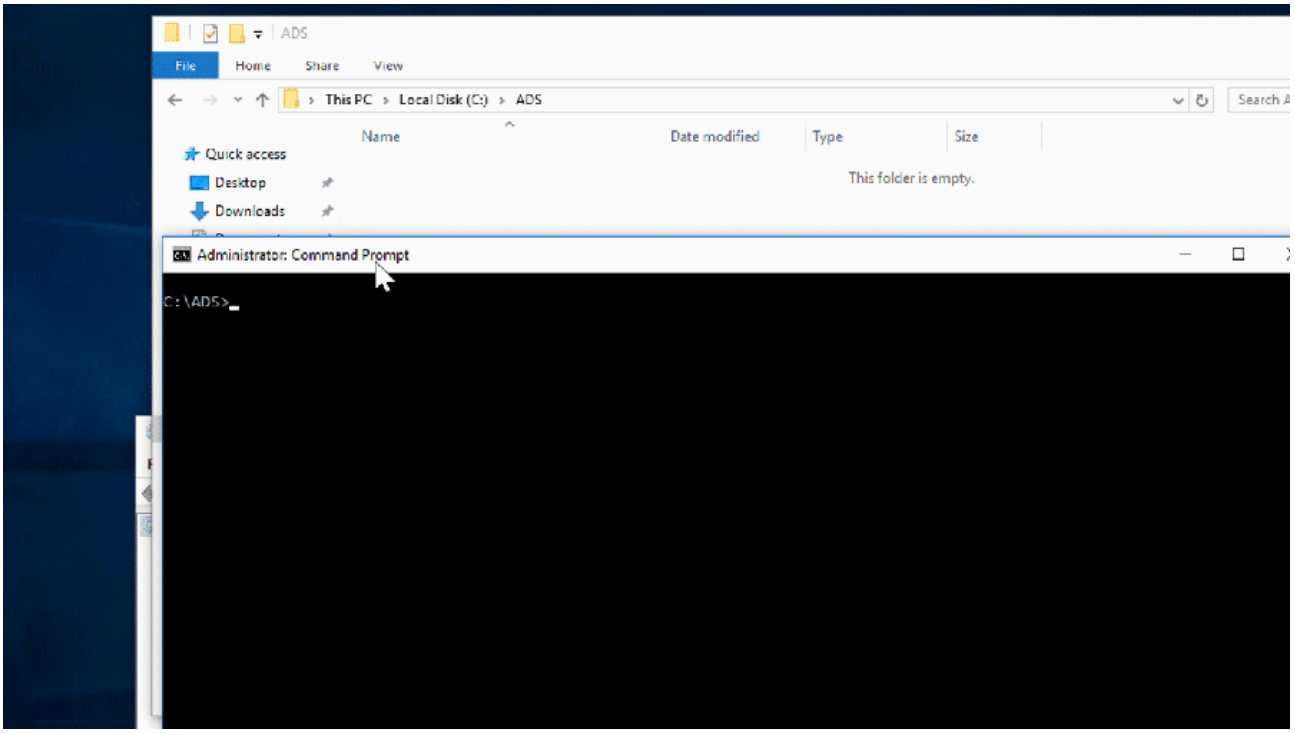


Executing ADS binary

I found another way to execute a binary from a alternate data stream when I was digging into this. It is possible to create a service in Windows (this requires local admin rights) that executes content from an Alternate Data Stream. I use the SC command to execute the necessary commands to create the service as want using these commands:

```
echo "empty file" > c:\ADS\file.txt
type c:\windows\system32\cmd.exe > c:\ADS\file.txt:cmd.exe
sc create evilservice binPath= "\"c:\ADS\file.txt:cmd.exe\" /c echo works > \"c:\ADS\works.txt\"" Di
sc start evilservice
```

And it looks like this:



That’s all for this time. I have also updated my ADS gist here for other methods: https://gist.github.com/api0cradle/cdd2d0d0ec9abb686f0e89306e277b8f

Hope you liked the post and as always I appreciate feedback. 🙂
