# String Obfuscation: Character Pair Reversal

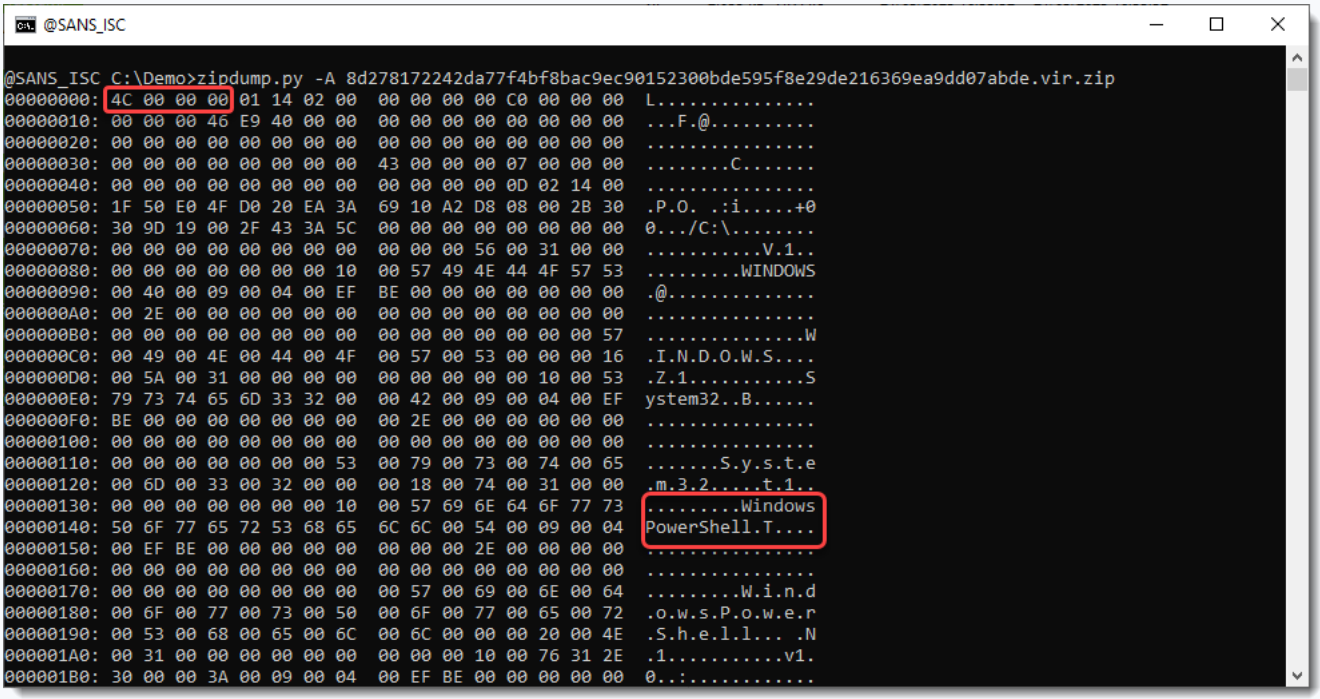
**Published**: 2023-03-21
**Last Updated**: 2023-03-21 16:48:19 UTC
**by** Didier Stevens (Version: 1)

0 comment(s)

I found a malicious .LNK file on MalwareBazaar that contains an obfuscated URL. The obfuscation is a classic one, with a twist: string reversal, not per character, but per character pair.

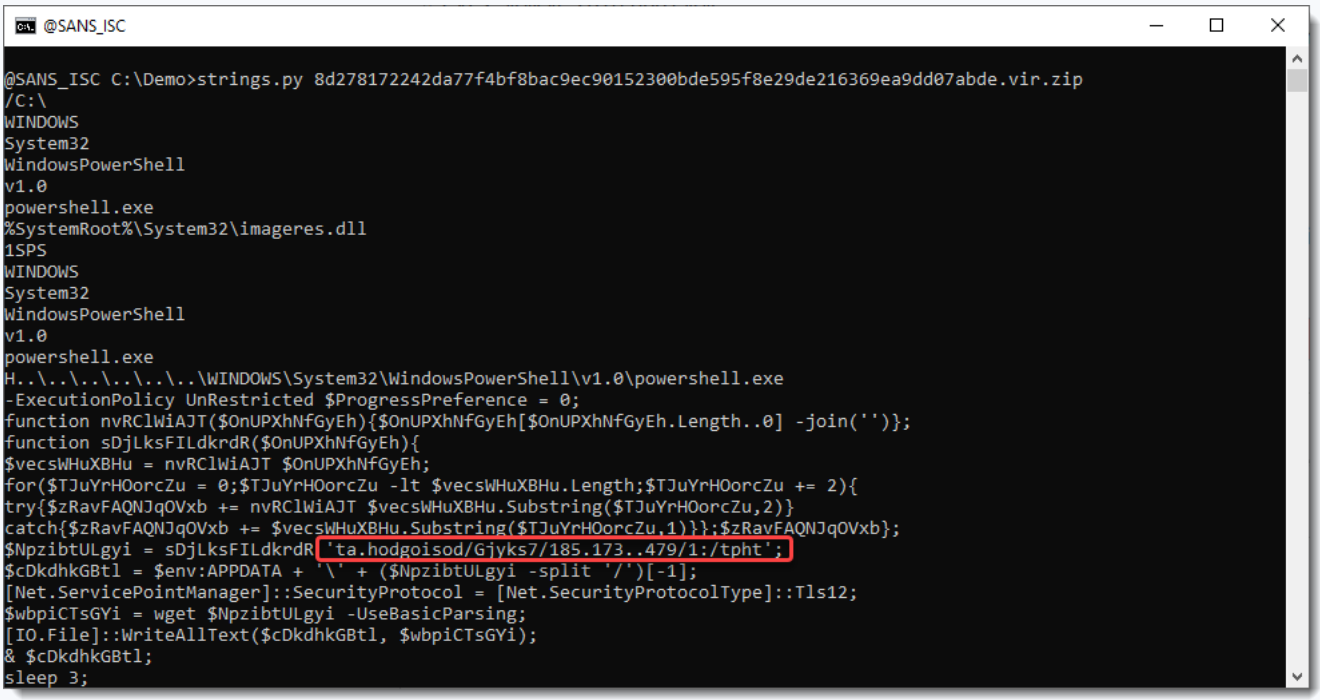
Let's take a look.

The file is indeed a .LNK file:




And it seems to contain powershell code.

In stead of using a .LNK parser, I'm just going to extract the strings of this .lnk file:




The powershell script contains a string, that looks like a mangled URL. It looks like a reversed string, but is a bit different.

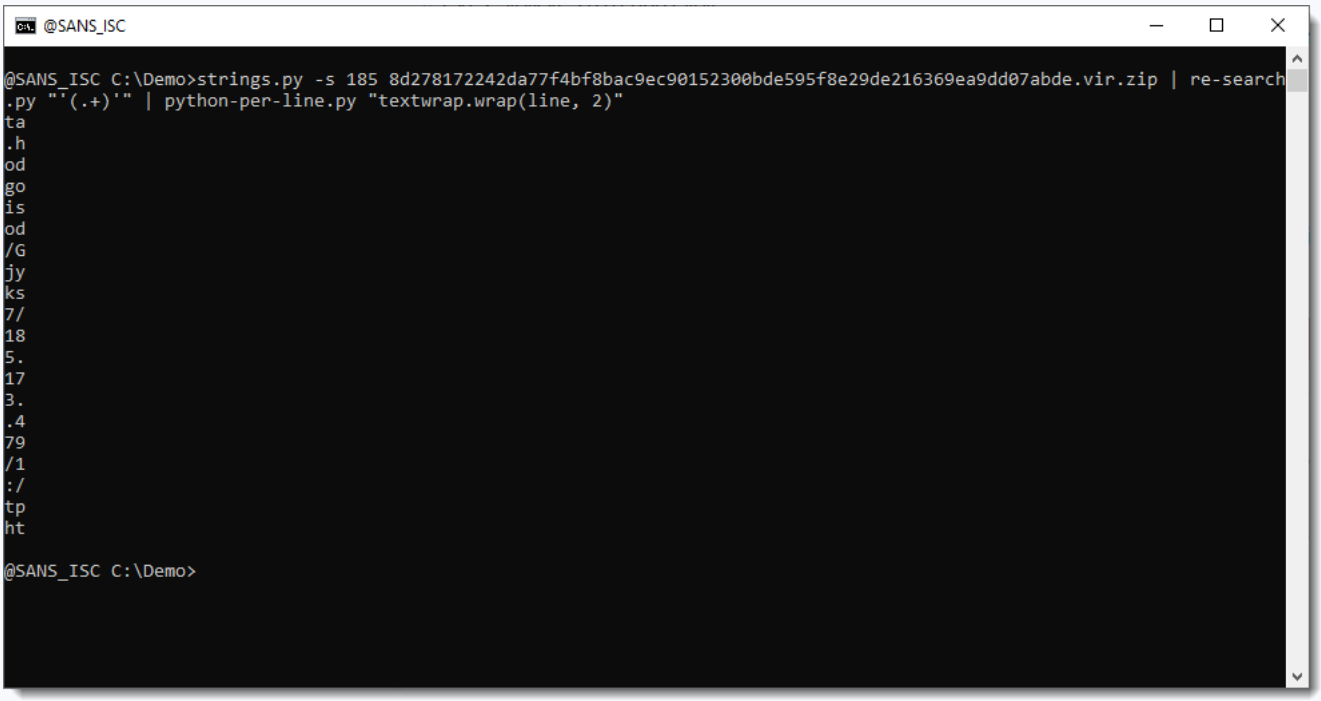
A hint on how to decode this, can be found 2 lines above the mangled URL: .Substring(..., 2)

The decoding algorithm works with 2 characters.

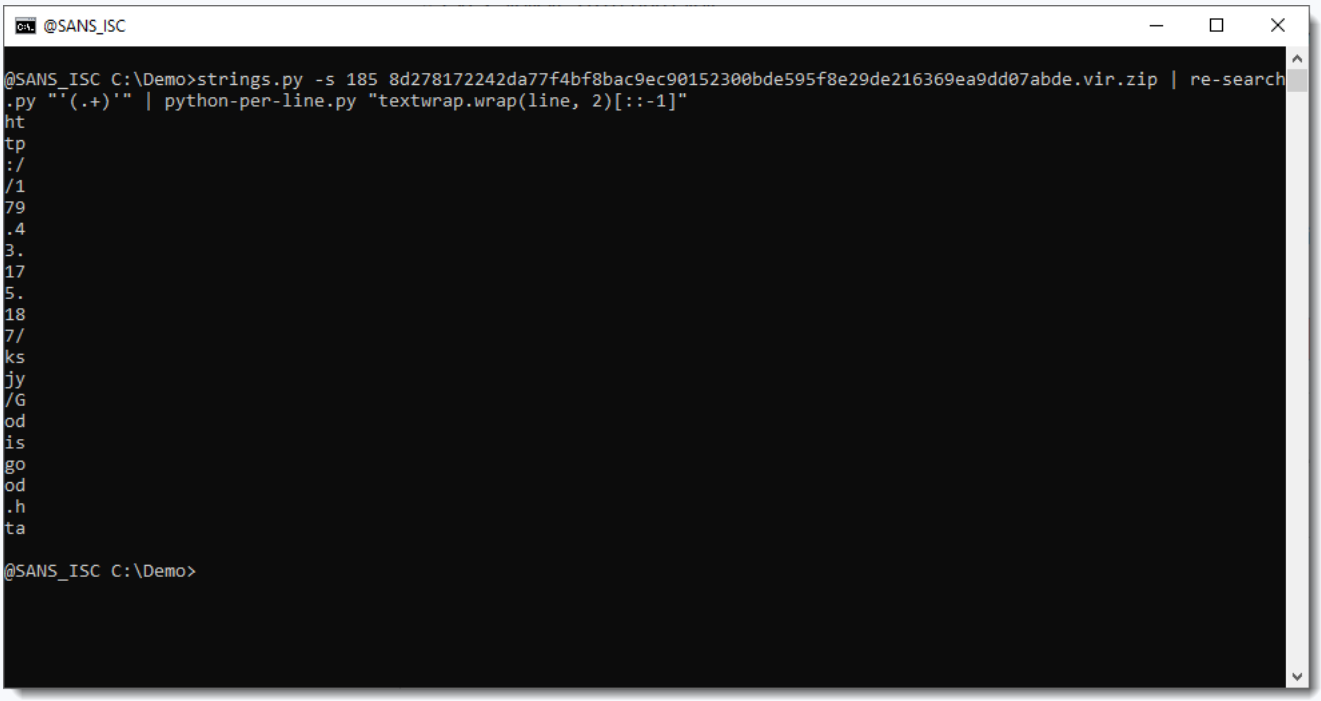
So this is a reversed string, but in stead of reversing character per character, reversing is done per character pair.

Let's isolated the mangled string and decode it:

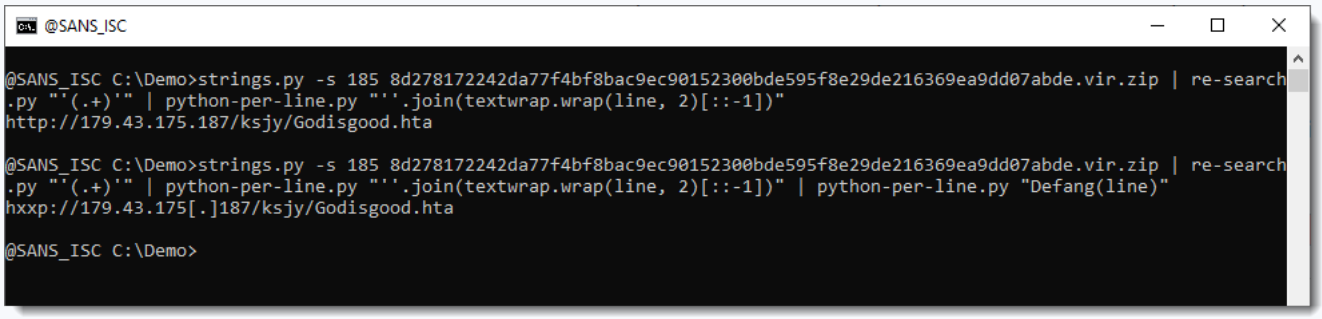

With Python's textwrap.wrap function, we can split up the string in substrings of 2 characters, like this:




Then we reverse this list of substrings ([::-1]):



And we join the substrings together:




Giving us the deobfuscated URL: hxxp://179.43.175[.]187/ksjy/Godisgood.hta

At time of writing, the payload was 6c1be182c5ae4b5cc44d1aedd202327c71253000d29d28e87686ad71bff41804.

This payload will ultimately download zgRAT
malware: f87246f639ed528fe01ee1fea953470a2997ea586779bf085cb051164586cd76 and 592f1c8ff241da2e693160175c6fc4aa460388aabe1553b4

Tools used in this analysis: zipdump.py, strings.py, re-search.py, python-per-line.py. All can be found on GitHub.

Didier Stevens
Senior handler
Microsoft MVP
blog.DidierStevens.com