Powershell Info

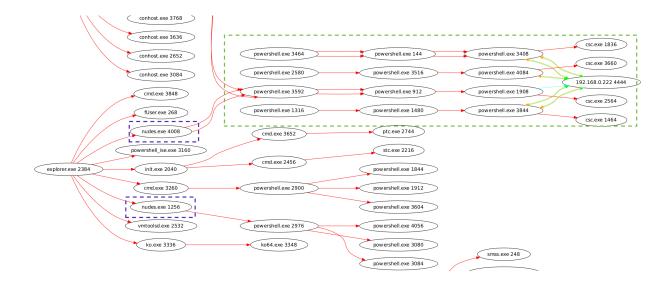
First stage binary is a 64B payload compiled on 10/09/2017

1G-Structure :	MZ(Mark Zbikowski)
leaderOffsetVal :	
StackSeg :	0000000
Stack* :	00000058
CkS :	0000000
Instr* :	0000000
leaderAdd :	00000080
***************	***************************************
## FILE TYPE => F	
# FILE_IIFE -/ F	
+	AMD
+	EXE_GT_2GB
+	_EXE_,GT_2GB Mon Oct 09 11:44:59 2017
+	18
+	0 <- Base*
+	GUI
+	(64B)
+	8704 <- CS
+	0x1000 <- CoseBase*
******	× <mark>*×≈×≈×≈×≈×≈×≈×≈×≈×≈×≈×≈×≈×≈×≈×≈×≈×≈×</mark> ××××××
*	.text:
*	.text: (X), I, (R),
×	data:
*	$data: I, \langle R \rangle, \langle W \rangle,$
*	.uata: 1, th/, tw/, .rdata:
*	\cdot rdata: I, $\langle R \rangle$,
*	.bss:
<u> </u>	.nss: .bss: U. U. (R). (W).
· · · · · · · · · · · · · · · · · · ·	$.088 \cdot 0, 0, \Lambda M, \Lambda W,$

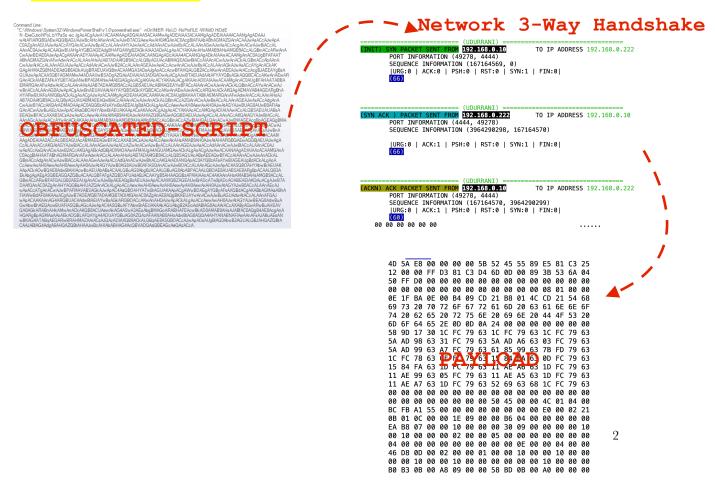
lst stage binary spawns powershell payload. Payload is buffered form one powershell to another powershell. This is to make sure that the payload remains fileLess and obfuscation is removed in the memory. Eventually a .tmp file is utilized by csc.exe



Here is the flow:



1st stage Payload -> Powershell -Powershell -> Powershell -> Communicate to C2 and spawn CSC.exe



Let's look atthe traffic flow

```
memset(rbp + 0xe0f0, 0x0, 0x68);
FUNCTION_1(rbp + 0xdfd0, 0xff, "%s\rlGnqU4iDgy3", *(rbp + 0xe160), array[2047], array[2048], array[2049], array[2050]);
fwrite(rbp + 0xe0580, 0x1, 0x4950, *(rbp + 0xe158));
fclose(*(rbp + 0xe158));
CreateProcessA(0x0, rbp + 0x2680, 0x0, 0x0, array[2047], array[2048], array[2049], array[2050], 0x0, 0x8000000);
```

File %s\rlGnqU4iDgy3 is created:

% represents the path, where to drop this file

rlGnqU4iDgy3 is the name of the file.

Eventually this file is dropped in the <mark>%TMP%</mark> location. Powershell will call csc.exe to use cvtres.exe. This is used to convert a resource file to an object file on the fly.