



One of the most common action you would take during debug of a failing SCCM Task Sequence is to dump the TS Variables to a file. This could be done in more than one way, including running a script as part of the TS. “**TSVarsSafeDump.ps1**” is such a script.

There are dozens of “dump scripts” available on the internet, so this is, in many ways, just another one – what makes it unique is that it excludes a few variables known to contain sensitive information, such as username and password for domain join and network access accounts.

Since we all tend to forget to remove the debug script from the TS we risk, as time goes, to have this information, that potentially can compromise the environment, laying around on hundreds, or more, computers – all in clear text.

To avoid this, we use this script, that filters out the information making the dump file harmless.

Requirement:

You must include Powershell Optional Component in you boot image if run in PE.

Usage:

Of course, it must be put in a package and distributed, then you can run it as a “Run Powershell Script” step where appropriate in you Task Sequence.

Properties Options

Type: Run PowerShell Script

Name: Dump TS-Variables

Description:

Package: eKlient 3.0 - OSD Scripts Browse...

Script name: TSVarsSafeDump.ps1

Parameters:

PowerShell execution policy: Bypass

 PowerShell v1.0 does not support Undefined and Bypass execution policies.

Update 2020:

Since first release back in 2016 it has become possible to run the script inline:

Enter a PowerShell script:

Edit Script... Script status: Script is entered.

Parameters:

PowerShell execution policy: Bypass Output to task sequence variable:

Feel free to add or remove variables as you wish, this is done on line #15

```

13 # Config Start
14
15 $HideVariables = @('_OSDOAF', '_SMSTSReserved', '_SMSTSTaskSequence', '_TSSub')
16
17 # Config End

```