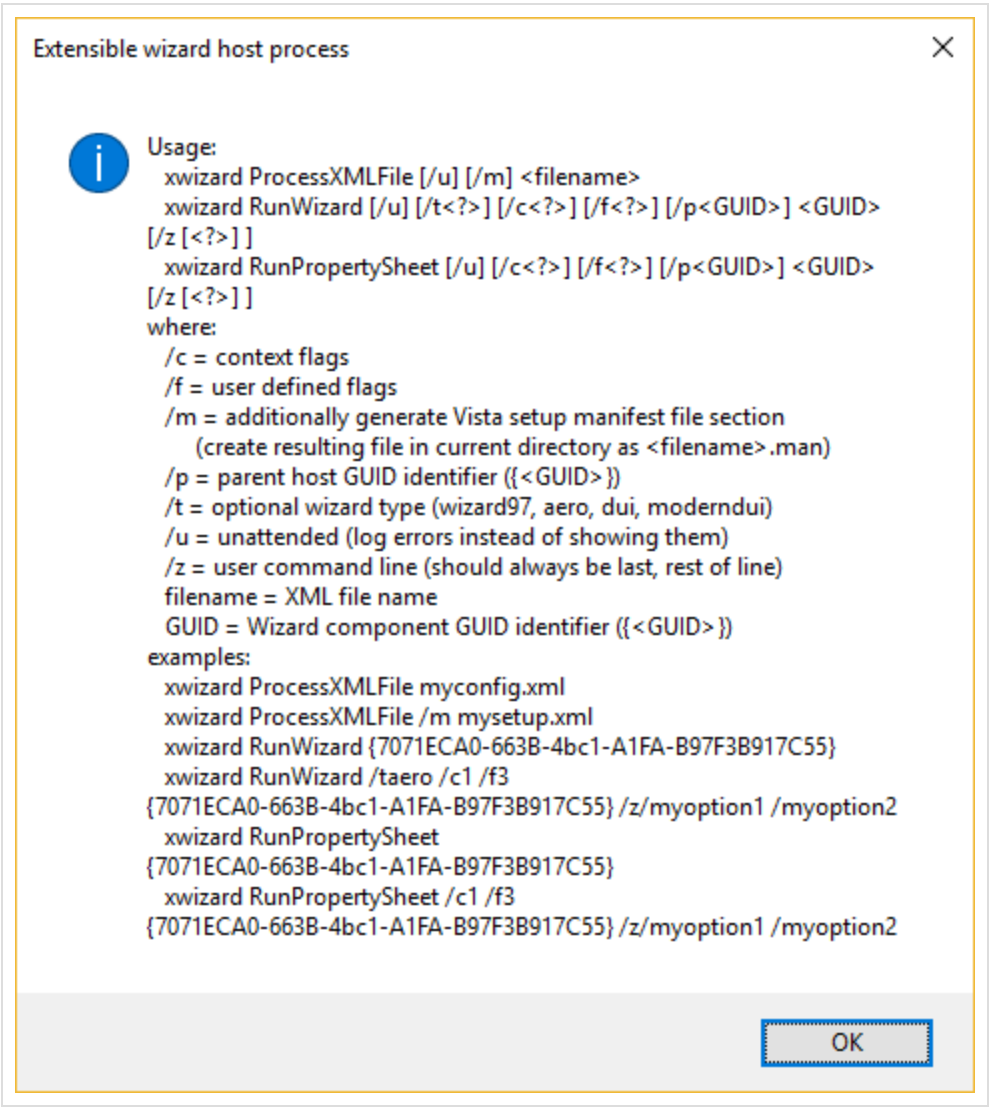


The Wizard of X – Oppa PlugX style

Xwizard is an ‘Extensible wizard host process’. While I am not 100% sure what it is doing I know for certain that – whatever it is – PlugX guys would approve.






















Why?

When you run it with a ‘h’ command line parameter, you will get this info:



Something about the unusual command line parameters described there caught my eye.

After a quick inspection I discovered why. The arguments are actually... names of functions exported from xwizards.dll!

E	Ordinal ^	Hint	Function	Entry Point
	1 (0x0001)	0 (0x0000)	DllCanUnloadNow	0x0000C390
	2 (0x0002)	1 (0x0001)	DllGetClassObject	0x0000C3D0
	3 (0x0003)	2 (0x0002)	ProcessXMLFileA	0x0000D9A0
	4 (0x0004)	3 (0x0003)	ProcessXMLFileW	0x0000DA30
	5 (0x0005)	4 (0x0004)	ResetRegistrationA	0x0000D3A0
	6 (0x0006)	5 (0x0005)	ResetRegistrationW	0x0000D430
	7 (0x0007)	6 (0x0006)	RunPropertySheetA	0x0000E7C0
	8 (0x0008)	7 (0x0007)	RunPropertySheetW	0x0000E850
	9 (0x0009)	8 (0x0008)	RunWizardA	0x0000DF00
	10 (0x000A)	9 (0x0009)	RunWizardW	0x0000DF90
	11 (0x000B)	10 (0x000A)	XWProcessXMLFile	0x0000C800
	12 (0x000C)	11 (0x000B)	XWRegisterHost	0x0000C830
	13 (0x000D)	12 (0x000C)	XWRegisterPageWithPage	0x0000CB20
	14 (0x000E)	13 (0x000D)	XWRegisterPageWithTask	0x0000CA20
	15 (0x000F)	14 (0x000E)	XWRegisterTaskWithHost	0x0000C920
	16 (0x0010)	15 (0x000F)	XWUnregisterHost	0x0000CC20
	17 (0x0011)	16 (0x0010)	XWUnregisterHostTaskLink	0x0000CEF0
	18 (0x0012)	17 (0x0011)	XWUnregisterPage	0x0000CE00
	19 (0x0013)	18 (0x0012)	XWUnregisterPagesLink	0x0000D0E0
	20 (0x0014)	19 (0x0013)	XWUnregisterTask	0x0000CD10
	21 (0x0015)	20 (0x0014)	XWUnregisterTaskPageLink	0x0000CFF0

Very nice!

And even nicer is the fact the LoadLibraryEx that loads that xwizards.dll finds its conveniently in the current path...

Ouch...

So... all you have to do is copy c:\WINDOWS\system32\xwizard.exe to your folder, drop your xwizards.dll DLL there and call xwizard.exe with at least two arguments.

And the Microsoft-signed xwizards.exe will load xwizards.dll of your choice...

This entry was posted in [Anti-*](#), [Compromise Detection](#), [Forensic Analysis](#), [Incident Response](#), [Living off the land](#), [Malware Analysis](#) by [adam](#). Bookmark the [permalink](#).