OUT-OF-BAND FILE TRANSFER ON CLOSED NETWORKS An Insider's Options

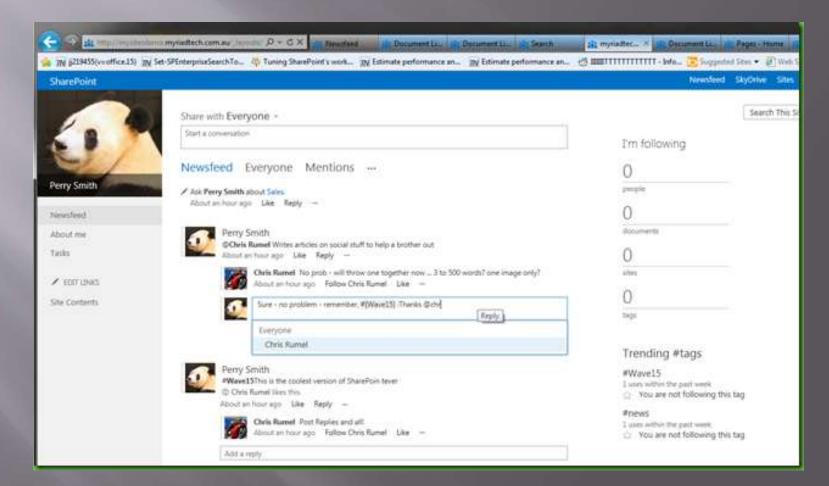
Michael Rich mike@tofet.net



- The Challenge
- The Tools Available
- Phase 0 Set up
- Phase 1 Hex Attack
- Phase 2 Attack of the Big Barcode
- Bringing it all Together
- Future/Branch Research Paths
- Conclusion

The Challenge

There I was, hacking the collaboration portal..



The Challenge Tumble

- How could I intercept the POST call to modify the inputs?
 - Tamper Data, Burp Suite, etc..
- How could I forge the POST call?
 - Curl, wget, etc..

Eventually: "How could I load one of these tools on to my closed, secure network without getting caught?"

The Conditions

- Closed, secured network (sort of)
- USB ports secured & monitored
- CD use secured & monitored
- Host-based security system
- Data transfer entry points do exist (DOTS)
 - Not in control of attacker
 - Unknown scanning rules
 - Leaves logs
- Windows / MS-Office environment

Tools Available

MS Office = Visual Basic for Applications
 Professional level printers & scanners

Adobe Acrobat OCR



Phase 0 - Set up

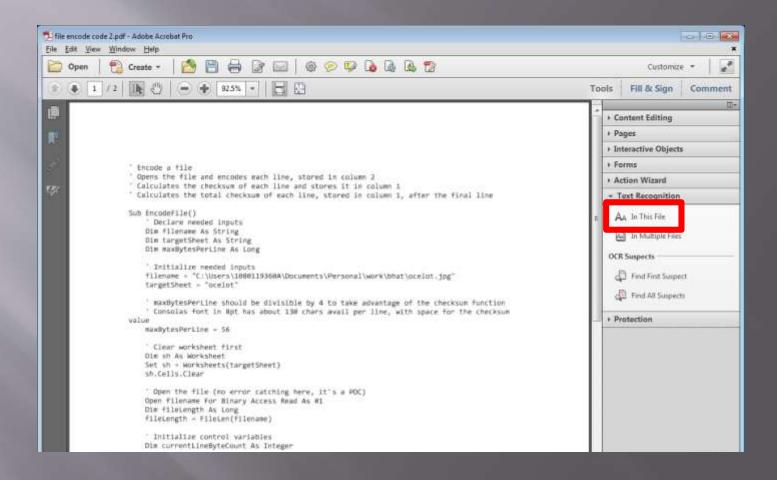
Put Excel into attack mode

File Home Sen	d	
🛃 Save As	Excel Options	
Save Attachments	General	Customize the Ribbon: 🕕
Open	Formulas Proofing	Main Tabs
Print Help	Save Language	
Options	Advanced	References Mailings
Exit	Customize Ribbon Quick Access Toolt	Review View Oeveloper
	Add-Ins	
	Trust Center	 ✓ Insert (Blog Post) ✓ Outlining ✓ Background Removal

Phase 0 – Set up

者 Microsoft Visual Basic for App	lications - text compare.xlsm - [Module5 (Code)]	Į	- • •
🦂 <u>F</u> ile <u>E</u> dit <u>V</u> iew <u>I</u> nsert	F <u>o</u> rmat <u>D</u> ebug <u>R</u> un <u>T</u> ools <u>A</u> dd-Ins <u>W</u> indow <u>H</u> elp	Type a question for help	- 8 ×
	🖌 🖅 🖓 🕨 🖬 🔛 🤮 🚰 😼 🔅 🛛 🚱 🛛 Ln 19, Col 9		
Project - VBAProject	(General) EncodeFile		•
Sheet2 (Exp_hex Sheet3 (Results Sheet3 (Results Sheet4 (Data Sou Sheet5 (ocelot) Sheet5 (ocelot-or Sheet5 (ocelot-or Sheet3 (Sheet1) Sheet3 (Sheet1) Module2 Module1 Module2 Module3 Module4 Module4 Module5 Module	<pre>' Encode a file ' Opens the file and encodes each line, stored in column 2 ' Calculates the checksum of each line and stores it in column 1 ' Calculates the total checksum of each line, stored in column 1, after the final line Sub EncodeFile() ' Declare needed inputs Dim filename As String Dim targetSheet As String Dim maxBytesPerLine As Long ' Initialize needed inputs filename = "C:\Users\1080119360A\Documents\Personal\work\bhat\ocelot.jpg" targetSheet = "ocelot" ' maxBytesPerLine should be divisible by 4 to take advantage of the checksum function ' Consolas font in 8pt has about 130 chars avail per line, with space for the checksum value maxBytesPerLine = 56</pre>		
Alphabetic Categorized	<pre>' Clear worksheet first Dim sh As Worksheet Set sh = Worksheets(targetSheet) sh.Cells.Clear ' Open the file (no error catching here, it's a FOC) Open filename For Binary Access Read As #1 Dim fileLength As Long fileLength = FileLen(filename) ' Initialize control variables Dim currentLineByteCount As Integer currentLineByteCount = 0 Dim currentRow As Long</pre>		× ×

Phase 0 - Set up



Consolas Font – Down to 8 pt font

Phase 0 – Set up

Suble (Seet) Suble (Seet) Woods Wo	T Microsoft Visual Basic for Applications - Book1 - [Module2 (Code)]									
Signal Wathrought Image: Signal S										
Image: Second a file Image: Second a file Opens the file and Description food (byock) Image: Second a file Image: Second a file Opens the file and Description food (byock) Image: Second a file Image: Second a file Image: Second a file Image: Second a file </th <th></th>										
Provemove (sect) Image: Sect Dect) Image: Sect Dect) <tr< th=""><th></th></tr<>										
Woodeltandteud (Sect) Caleilates the end Monord Your Bair (Applications - Book (Journa)) The fail (Journa)										
Departed Sherid) Decide Sherid Decid										
Swelt (Swelt) Sub EncodeFile() Sub EncodeFile() Sub EncodeFile() Decise Voide Dis EncodeFile() Sub EncodeFile() Sub EncodeFile() Dis maxBytesFetLing Encode Sub EncodeFile() Sub EncodeFile() Sub EncodeFile() State Classeviel Sub EncodeFile() Sub EncodeFile() Sub EncodeFile() State Classeviel Sub EncodeFile() Sub EncodeFile() Sub EncodeFile() Sub EncodeFile() Sub EncodeFile() Sub EncodeFile() Sub E	14 A 12-									
Traviosion In FileFame A.s In FileFame FileFileFileFileFileFileFileFileFileFile										
Produce Image: Specific Amodel Image: Specific Amodel <										
<pre> " Module" " Initialize meet " Module" " Initialize meet " Module" " Initialize meet " Open the file and encodes each line, stored in column 3 " Open the file and encodes each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Calculates the coal declawam of each line, stored in column 1 " Initialize content column " Initialize content column " Calculates the coal declawam of each line, stored in column 1 " Initialize content column " Calculates the coal declawam " Initialize content column " Initialize content column " Calculates the coal declawam " Calculates the coal declawam " Initialize content column " Calculates the coal declawam " Calculates " Calculates " Calculates " Calculates " Calculates " Calculates " Calc</pre>	_									
Properties - Module2 Filescame = * 0:10m Value Calculates the coal checksam of each line, stored in column 1 Sub EncodeTie() Sett(Sect) Value Display the filescame maxBytesFerLine at converting the convere converting the converting the converting the converti										
Properties - Module? Image: Sheet = *ock Value Image: Shee										
Properties - Module Markey respression and respression anderespression and respression and respression an										
Properties - Module2 Consolase froms in file Properties - Module2 Image: Module - Module2 Produce2 Module - Module2 Dim sh = Worksheet - Module2 Athebetk Categoroud Consolase from sin file Produce2 Module - Module2 Image: Module2 Athebetk Categoroud Consolase from sin file Consolase from sin file Image: Module2 Athebetk Categoroud Consolase from sin file Consolase from sin file Image: Module2 Nume Module2 File Nume Module2 Image: Module2 Nume Module2 Image										
Properties : Module2 Xi Properties : Module2 Mode Dim sh Am Worksheet sh. Cells. Clear Akhdekt Categorized Dim sh Am Worksheet sh. Cells. Clear 'Open the file (n Open filename For Dim fileLength As fileLength = File 'Initialize control 'Value2' Dim maxBytesFerline As Long 'Initialize needed inputs filename = "Ci\Users\UseSt\UseStL\Documents\Fersonal\work\bhat\ocelot.jpg" targetSheet = "ocelot" 'Value2' NumeS Wook/2 ModeX Mode ModeX Mode * Initialize needed inputs filename = "Ci\Users\UseStLDOCUT" 'Value2' NumeS Wook/2 ModeX Mode ModeX Mode * Initialize control 'Initialize control 'Initialize control 'Initialize control 'Initialize control 'Initialize control variables for eurrentRow = 0 'Initialize control variables for eurrentFilePos 'Initialize control variables Morecol Moud Brit Magint magint for Applications Sub of function not defined Dim surrentKow As Long currentLineByteGout = 0 Dim surrentKow As Long										
<pre>Properties : Module2 x; Foodule2 Module</pre>										
Bet sh = Worksheet sh.Cells.Clear Set sh = Worksheet sh.Cells.Clear Workey Modulez Copen the file (in Cpen fileLength As fileLength = File) Versel Modulez FileLength As fileLength = File) Versel Modulez Produlez Module Versel Modulez Produlez Versel										
Advalue: Voide Advalue: Categorized V Cpen filename For Din fileLength As V Cpen the file (in point) Module: V Cpen filename For Din fileLength As Din currentRow As OurrentRow As Charter (Categorized) V Cpen the file (in ourrentRow As Charter (Categorized) V Charter Voide V Consolar font in Spt has about 130 charts avail per line, vith space for the checksum Name V Consolar font in Spt has about 130 charts avail per line, vith space for the checksum Name V Consolar font in Spt has about 130 charts avail per line, vith space for the checksum Name V Consolar font in Spt has about 130 charts avail per line, vith space for the checksum Name V Consolar font in Spt has about 130 charts avail per line, vith space for the checksum Name V Consolar font in Spt has about 130 charts avail per line, vith space for the checksum Name V Categorized) V Categorized V Cate										
Open filename For Dim filename For Dim currentLineByteOdde * Initialize contr Module2 * Initialize contr Module2 Module * Initialize contr Module2 Module * Initialize contr Module2 Module Module2 Module * Initialize contr Module2 Module Module2 Module <th></th>										
<pre>bin fileLength As fileLength As fileLength As fileLength As fileLength As fileLength As fileLength = File ' Initialize control Properties 'Module1 ' Onnoblas font in Spt has about 130 chars avail per line, with space for the checksom ' Consolar font in Spt has about 130 chars avail per line, with space for the checksom ' Consolar font in Spt has about 130 chars avail per line, with space for the checksom ' Consolar font in Spt has about 130 chars avail per line, with space for the checksom ' Consolar font in Spt has about 130 chars avail per line, with space for the checksom ' Consolar font in Spt has about 130 chars avail per line, with space for the checksom ' Consolar font in Spt has about 130 chars avail per line, with space for the checksom ' Consolar font in Spt has about 130 chars avail per line, with space for the checksom ' Consolar font in Spt has about 130 chars avail per line, with space for the checksom ' Consolar font in Spt has about 130 chars avail per line, with space for the checksom ' Consolar font in Spt has about 130 chars avail per line, with space for the checksom ' Consolar font in Spt has about 130 chars avail per line, with space for the checksom ' Clar worksheet first Dim sharentow are ' Clar worksheet first Dim sharentow are ' Open filename for Binary Access Read As #1 Dim fileLength As Long ' Initialism control variables Bin currentLineByteCount = 0 Dim currentLineByteCount = 0 Dim currentLineByteCount = 0 Dim currentNew As Long ' Get a Byte</pre>										
<pre>* Initialize control Hopethis - Mnodel Xi Dim currentLineByte Control Hopethis - Mnodel Xi Produke2 Mode * Clear Vockaneet first Dim currentRow As currentRow As currentRow As currentRow As currentRow As Dim sharkCoded = *** Dim chainChecksum chainChecksum = 0 * Dim chainChecksum chainChecksum = 0 * Else file mane sht.Colls(1, 1) = 1 * Encode the data For currentLineByteCount As Integer currentRow As For currentFilePool * Get a Byte * Set a Byte * Set</pre>										
Dim currentLineByteCot Module2 Noble ImaxBytesPerLine 56 CurrentLineByteCot Module2 Noble ImaxBytesPerLine 56 Dim currentRow # 2 Dim hexEncoded As Module2 hexEncoded # "" Dim chainChecksum ImaxBytesPerLine 56 chainthecksum = 0 ImaxBytesPerLine 56 Completence sh.Cells.Clear Open filename For Binary Access Read As #1 Dim fileLength As Long fileLength # fileLength # fileLen(filename) ImaxByteScount #0 Sub or Function not defined 0K Help										
durrentLineByteCot Adnabetz Gategorzed in chareGotzed Dim currentRow # 2 Dim hexEncoded As hexEncoded As Now Module2 Dim chainChecksum Open filename For Binary Access Read As #1 Dim chainChecksum Open filename For Binary Access Read As #1 Dim chainChecksum Titalize montrol variables St.Cells(1, 1) = 1 Titalize montrol variables Process Byte Dim currentRow # 2										
currentRow = 2 Dim hexEncoded As hexEncodes Open filename For Binary Access Read As #1 Dim currentRow = 0 Dim currentRow = 2 OK Help										
Dim hexEncoded As sh.Cells.Clear hexEncoded = "" ' Open the file (in error catching here, it's a POC) Dim chainChecksum = 0 ' Flace file name ' Flace file name ' Initialize montrol variables bin currentFilePea Din currentExex & Long ' Get a Byte OK										
Dim chainChecksum chainChecksum = 0 ⁴ Fince file mame sh.Cells(1, 1) = 1 ⁵ Initialize control variables For durrentFilePon ⁶ Get a Byte ⁶ Get a Byte										
ChainChecksum = 0 Dim fileLength As Long ' Flace file mame fileLength = FileLen(filemame) ' Initalize montrol variables Dim currentLineByteCount As Integer ' Set a Byte OK										
' Flace file name sh.Cells(!, 1) = 1 ' Initialise control variables OK Help ' Encode the data For durrentFilePos ' Get a Byte Dim currentEneByteCount = 0 Dim currentRow As Long currentRow = 2 Dim currentRow = 2										
sh.Cells((, 1) = 1 Dim currentlineByteCount As Integer 'Encode the data currentLineByteCount = 0 For currentFilePos Dim currentRow As Long 'Get a Byte currentRow As Long										
For durrentFilePos 'Get a Byte CurrentRev As Long CurrentRev As Long										
'Get a Byte Currenze w a cong										
Din hexEncoded As String										
binnediate Din chainChecksum As Long										
chainChecksum = 0										
* Place film name in the sheet										
sh.Cells(1, 1) = filename	_									
Immediate										
	لي									

Use Phase 0 methods to make Excel a binary file hex encoder/decoder

- Why hex?
 - Printable text
 - Tests showed excellent OCR results

	Hex Encoding	Base64 Encoding
Encoding	Consolas, 8 pt font	Consolas, 12 pt font
Word Length Errors (80 char words)	0 in 73 words	9 in 73 words
Transcription Errors	0 in 5840 symbols	216 in 5840 symbols
Error Types	1 to 1	Many to Many

🔣 🛃 🖉 = (°' =) =			text compare.xlsm	- Microsoft Excel		_ 0	22
File Home Ins	ert Page Layout For	mulas Data	Review View	Developer Acrobat		V 🕜 🗆 (
File Home Ins A2 A B 2 FC7B FFD8FFE0 3 2128 18141418 4 6BC4 2C2C2C2 5 3A07 11010313 6 4E6C 01000203 7 7570 00000000 8 1EC3 021103113 9 B4E6 8F35AE60 10 2CD2 EFB361C0 11 C19D 8F013800 12 7A49 494942EA 13 B9E5 946AB11 14 1DD4 FE21F8C0	ert Page Layout Form	E F 000100010000FDI 312125292822228 12125292822222 10100000000000 261718191A11432 400221101010101 28A59412A511E46 688390DAEDFD13 6038E711E70DEC0 66F27C791E9A027 92385604E5CCE49 9082882B9633F54 911ABE67B4DBE84 9840B884	Review View G 3008400 (171F33 4252B1 0002030 00200 00200 002000000 00200000 00200000 00200000 00200000000			♥ ⑦ □ □ 141314351514151619 288304A04A046000E1 202202202202202202020 20202202020202020 3382E1F0172453637 000000011162122100 8382E1F017245363010444 A39D1028A06A188EE1 9658C024908E768508 9658C024908E768508 9658C024908E768508 9658C024908E768508 9658C024908E768408 9658C024908E768408 9658C024908E768408 9658C024908E768408 9658C024908E768408 9658C024908E768408 9658C024908E768408 9658C024908E7640788 9658C024908E7640788 9658C024908E7640788 9658C024908E7640788 9658C024908E7640678 9658C024908E7640788 9658C02498E725911 9648C74259426281028 9658C02108771 8472021547550288128 8472921547550288128 847202108711 847202108711 847202108711 847202108711 847202108711 847202108711 847202108711 847202108711 847202108711 847202108711 847202404846095	5181718141514141417141415171614 5181718141514141417141415171614 5181718141514141417141415171614 5181718141514141417141415171614 5181718141514141417141415171614 5181718141514141417141415171614 5181718141514141417141415171614 518181718145141417141415171614 518181718145141417141415171614 5181817181420000000000 51818171821200000000000000000000000000000
16 4658 7EDACC1 17 D192 C63D916 18 3228 C82DFF0 19 93EE C7F44394 20 6185 23F0C0A 21 2F1A 6D6789A 22 8703 98A95C3 23 3175 AA21A62 24 ED95 B8347EF0 25 1F36 C436BC5	FE7F457D73EE557C540F1 71D370C4C1231DA806B9 1C32317235AD68A15B803 00B71ABF0CF6C18A75B10 81D0B7632347BB82261A5 013C41DD27883BA5FB28/ 2FCDD9589E6E546ECB478 CAC2F712F3C86E46DDF92 05C40A2E35EDF055F9DE2 05C40A2E35E0F055F9DE2 05C40A2E35E0F055F9DE2 05C40A2E35E0F055F9DE2 05C40A2E35F0F05F9E2 05C40A2E35F0F05F9E2 05C40A2E35F0F05F05F9E2 05C40A2E35F0F05F05F05F05F05F05F05F05F05F05F05F05F	95D5AE6D9F6A36 356E9F959232F09E DA743C8A2DD5F8 8DE2DAE0EF620A AE686FC18EE80D0 8E26EBC964BAB90 2F4BE2BCD4B4784 24C5A1A29D2BF9 2F32E91B8786590 559F6970A176D6E	A4E07F DEAF91 5DDC59 528C70 888496 0B82E70 50537F 3FC478 886968E 77 509D9A	995,88347EF8F829E7E38885507C9E8 736,C4368C594968295565E99E5A303 850,34618558080588860F7EA59362 997,58684478863587F598260C18076 998,5868486601A80EC08F92785051C 338,954879504FC49555F1A25C96007 988,310291A367A8753449503803748 578,167F19386821644555F1230EE449 958,556114F3040F402CF4CA55FA521 1455,25C57A1080078666633800C80041 105,337208843508832AF7805964A70 978,28871240CE8719880521F290506041 105,337208843508832AF7805964A70 978,28871240CE871988052129CD8108 144,886867590CA8902390AC1143775 885,14731A8C448C2078C59694E1643 951,3123150F6630031897A22468 146,3364124821242816678034674 973,148C448C2078C59694283673 973,148C448C2078C5969428367 973,148C448C2078C5969428367 973,148C448C2078C5969428367 973,148C448C2078C5969428367 974,14867144827343248166F780528363 171,1680F14482733436779CC595883 17,1586F146C8144814471202080555 172,4750281291C57248616675651202 174,4554028146214413477202860555 174,1680F14482743210280575876106 135,85784462144814471202806055 145,9523416290E18548420008C784	22F32E91887865983FC478283C1D8EBD861F1 5559F6970A176D6886968D86D86D89A739285778 6122C8582BD86468909A893C88FA8593428C 18523CF9F4D388D468F20E8577D215239883C	2DF9A88F739CD68DCT 3CBCC7317538596956 3CBCC7317538596956 639384Ca39AE97469 30806F85948D2855 439384Ca39AE97469 30806F85948D2855 44812081C010b284 409326C5F387753584545 409326C5F387753584545 409326C5F387753584545 409326C5F387802 6044545607856078015 804045412085420985730 968454912085420985730 968454912085420985730 968454912085420985730 96845491208542096580 4722512885412665865 472454925425958545533 4822598250965836475 43772298354845533 4822598250965836475 483772298354845533 4822598250965836475 536462971229835488575854 9530762488745887565 53646297122981578534 953076248875854597855 53646279122643974 053076248875854597855 5364627912548875854 9530762418693524974 9530762418693524974 9530762418557854 9530762418557854 95307624418578545 95307624418578545 9530762440127	7591A86DB36F58228FE2E482E7C9A84 34F988AAEE473FB9512798ED180600 EF40AD46F0DBF02F3EA6858658A31 3566A8F119C12E2010BA6058272ECF42 1765C3FE512ABEF8A49AF5FE2504007 46D6CB306813C20C8BEF288686841081 3520328202C2099010915928256866 3550482CD3C8060DB41208C747407A40 948969C5C3346C2358735F1EA307704 4883573083398E53243218A61A1CP66 708908E0CF66A1C8079600423046687 7556647477523286732F083381068F1 3778CBAC8A44178120479495326568508 3675180FB100687886501866308764E 8738A6246F5C08E83070567F408C38E 32202760834417850A625086758F08C38E 3420C760838022809A82208849134601 267386AC97766AA2208849134601 267386AC97766AA2208849134601 267386AC97766AA2208849134601 267386AC97766AA2208849134601 267386AC97766AA2208849134601 267386AC97766AA22088473467E06 320C6406807347785208F1815820F1 3266686A0C8163766FA2508F1815820F1 3266686A0C8163766FA2508F1815820F1 3266686A0C8163766FA25082885EFECF 714456A306B8FA25748518585EFECF 714456833400597861209712732470653 76623703080AC7422714885885EFECF 714456833400597861209712732470653 76623703080AC742271485885EFECF 714456833400597861209712732470653 3265683664683340059880738219963 3281293CC8807966585484845F416

Hex Encoding is good, but probably not perfect
Need compact error detection

2-byte XOR checksum

Immediate

This data is corrupt Cannot decode data, it is corrupt

ĺ	< 🛃	19	- (21 - 1	Ŧ				text comp	pare.xlsm	- Microsoft Exc	el								
	File		Home	Insert	Page Layou	it Formulas	Data	Review	View	Developer	Acrobat								
Γ		1	43	-	f_x	6BC4						•							
		А					В					С	D	E		F	0	6	
	1 FC	7B	FFD8FFE	E000104A4	649460001	010000010001	0000FFDB	0084000906	50714121	214141314151	151415161	5181718	1415141414	1714141	517161	14			
	2 21	28	1814141	81C28201	81A251C141	142131212529	2B2E2E2E1	71F333833	2C37282	D2E2B010A0A	0A0E0D0E	1A10101	A2C241F24	42C2C2C	2C2C20	C2C2C			
	3 <mark>6</mark> 8	C4	3C2C2C	2C2C2C2C	2C2C2C2C2C2	C2C2C2C2C2C	2C2C2C2C2	2C2C2C2C2	C2C2C2C	2C2C2C2C2C2	2C2C2C2C2	C2C2C3	7FFC000110	0800BC01	10D030	12200	02		
	4 3A	07	1101031	101FFC40	01C0000010	50101010000	000000000	000000030	01020405	06000708FFC	4003E1000	0104000	404040306	04060007	700000	0			
	5 4E	6C	0100020	31104051	2210631415	113226171819	1A114324	252B1C123	6292D10	7153382E1F01	724536372	A2B2FF	C40018010	00301010	000000	00			
	6 75	70	0000000	00000000	0000102030	4FFC40022110)10101010	0020300030	00030000	00000000011:	102122103	3141325	161132242	FDA000	C03010	00			
	7 1E	C3	0211031	1003F00F	677A88E75A	A97228A59412	A511E46D	8A2A3CA3	AA92E40	956668CE720	3CA2BD01	C54D388	18E779826	3DDBD7/	A21E6F	F7989	4		
	8 B4		8F35AE6	5E7F9D6D	7F8C3C04E4	89C1688390D	AEDFD13A	4342D3603	B5F74AF	DE1CFA39D1	028AD6A1	B8EE11C	AAE26818	DC03666	163B70)55666	98		
	9 2C	D2	EFB3610	CO6FDE3E5	6FA01D568	19C96038E71	LE70DECD	BF993FD82	3AFB87C	4DF4A965BC9	249BBE76	B5BC1B9	A924C4E3	53A7A1E	D6B01	0637D	56		
1	L0 C1					32FB6BF27C7													
	L1 7A					483C92385604													
	L2 B9					FDFF00828B2													
1	12 1D		FE21E80	F7693A6P	7DD/E169/	452911ABE67	'RADRESA	145646068	ROSESUS/	A91A39C61AF	ISERB027E	44D7C17	ADB765E0	RAUBAUI	IRDE20	6FE71	767		

Phase I - Oops.. Real World

Assumptions, assumptions, assumptions

1551 errors in 135,420 symbols (1.1 % error)

- B to 8: 261; 1 to 1: 359; 5 to S: 864
- D to 0, O: 57; 6 to G,q,b: 3
- Alternative characters:
 - # for B
 - ? for D

Auto-replace other major errors

- "1", "S["], ".", " "
- Add strong visual indicators
- 1 manual correction in 1210 lines of text

Demo Time!

• Pros:

Extremely reliable

Can be entered by hand if no scanner

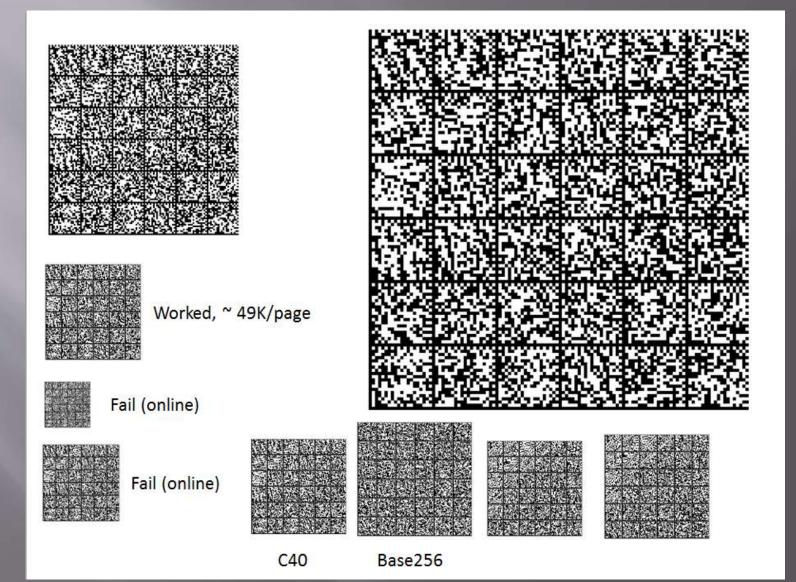
• Cons:

- Low data density: ~3.6K per page best case
- Common Tools:
 - PowerSploit: 835 kB = 232 pages
 - Mimikatz: 538 kB = 150 pages
- No exfiltration "compression" advantage



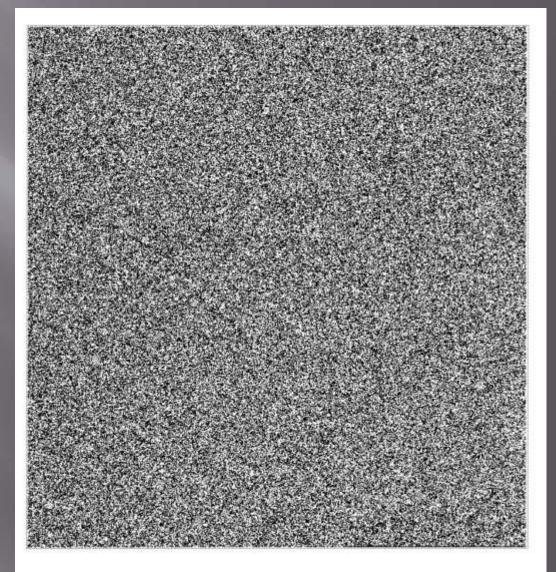






Phase 2

• ~ 25 K data per page 60% error correction Good features Timing lines Reed-Solomon FEC Different design problem ■ I can make it better!



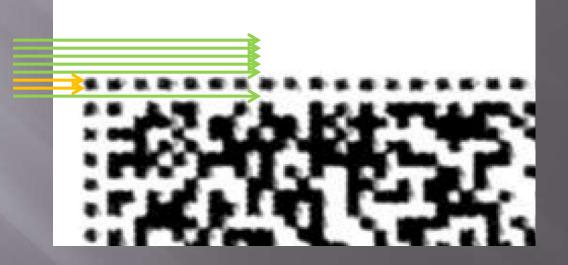
Phase 2 - Big Barcode

Data grid where each pixel represents one bit state (white = 1, black = 0)

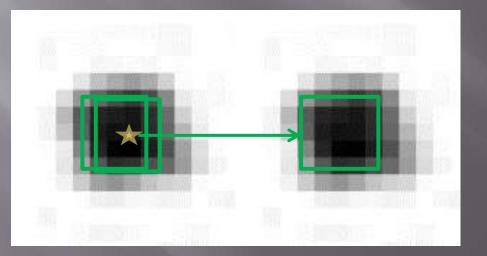


Printed at 72 dpi, get about 88 bytes across
 ~ 85 kB data per page

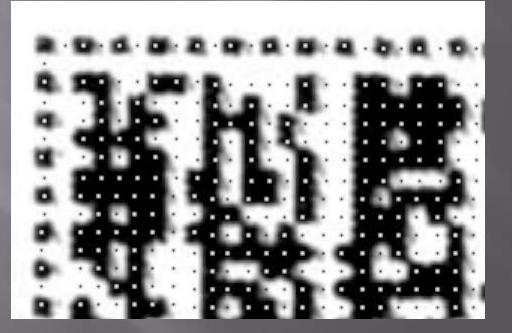
Finding the timing marks <u>Start with raster scan across the image</u>



Finding centers of timing marks
 "Wiggle Fit" from "root" pixel
 Best mask fit

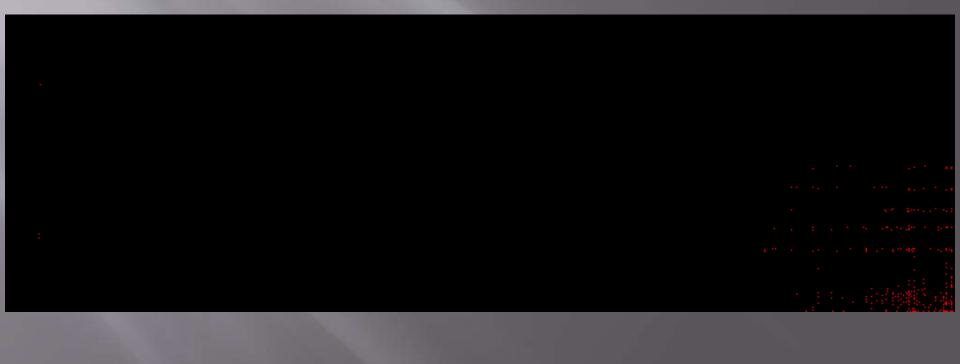


Timing mark location is very successful:
 Once all timing marks found, simply compute a grid of intersections to locate data:



Results:

20K of binary data: 189 bytes missed (0.953% error)
65K of binary data: 491 bytes missed (0.76% error)
72K of ASCII data: 972 bytes missed (1.35% error)



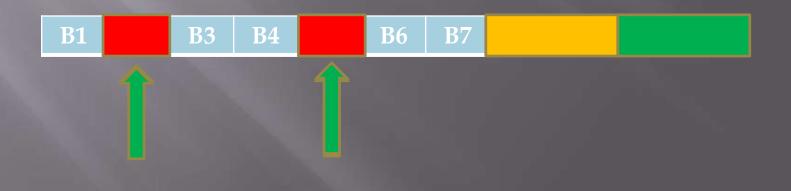
and and a second se

Reed Solomon FEC- Two Types

Forward ERASURE Correction

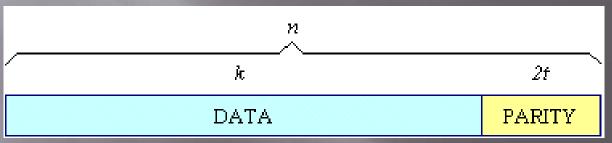
Block 1	P1	Block 1	C1	P1	Block 1
Block 2	P2	Block 2	C2	P2	Block 2
Block 3	P3	Block 3	C3	P3	Block 3
Block 4	P4			P4	Block 4

Forward ERROR Correction



Error Correction Details

Reed Solomon encoding

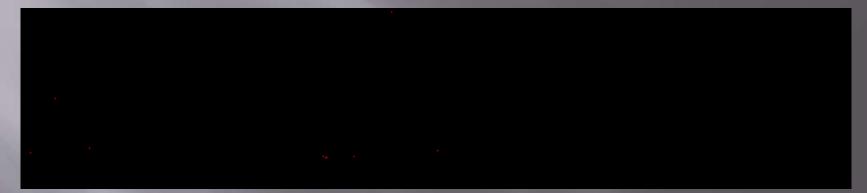


From http://www.cs.cmu.edu/~guyb/realworld/reedsolomon/reed_solomon_codes.html

- Codewords can be 2^s symbols long, each symbol s-bits wide
 - S = 8, codeword is 255 symbols; each symbol 8 bits wide
 - S = 16, codeword is 65535 symbols; each symbol 16 bits wide
 - Codewords can be less than n symbols long
- Can correct up to "t" symbol errors (2 parity symbols required for find and correct each error)

Real-World R-S FEC

Few open-source error correction libraries Those that do are 2⁸ only

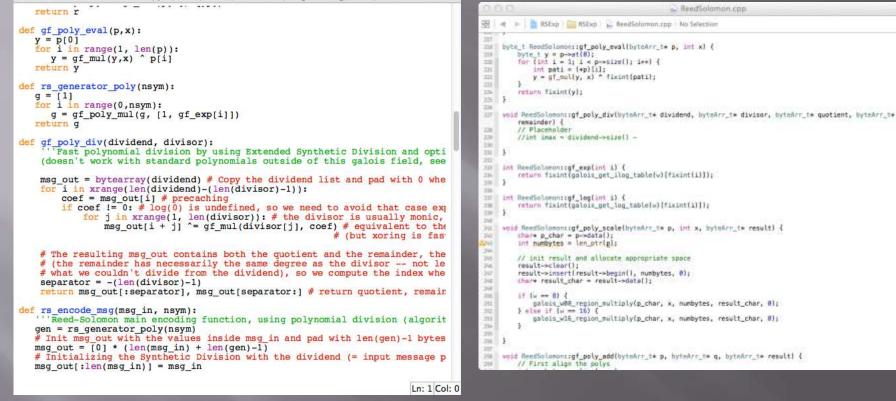




Coding an R-S FEC

44.

O O ReedSolomonTutorial.py - /Users/mrrich/Documents/Programming/RSTut/ReedSolomonTu....



https://en.wikiversity.org/wiki/Reed%E2%80%93Solomon_codes_for_coders

Big Bar Code Results

With s=8, k=140 to work reliably
~47 kB per page of data (~38 kB of parity)
PowerSpoit: 18 pages (vs. 232 pages in hex)
Mimikatz: 12 pages (vs. 150 pages in hex)

Demo Time!

Proof Of Concept

 Goal: Using techniques described here, install PowerSploit on a machine

Step	Resu!t
Interpret a page-sized bar code	
Reed-Solomon Encoder/Decoder	\checkmark
Build Sideload Library	\checkmark
Encode, Print, Scan, Decode payload with library	
Print, Scan, and load hex encoder/decoder into Excel	
Emplace library using hex OCR method	\checkmark
Encode/decode using DLL called from Excel	\checkmark

Future/Branch Research

Big Bar Code

- Reduce size of BBC DLL
- Improve error rates
- Get 2^16 Reed Solomon FEC working
- Add color to BBC
- Excel-a-sploit
 - Hex Editor
 - Steganographic encoder/decoder
 - Restore command prompt
 - Direct DLL injection?

Conclusion

- Big Bar Code POC was a success
- Standard office tools provide a lot of power
- If a user can code, a system is not secure
- Innocuous input/output systems can be used for creative purposes

QUESTIONS?