Hiding data inside the padding area in files and packets

Fady Mohammed Osman Pen. Tester fady.mohamed.osman@gmail.com September 2010

Hiding data inside the padding area of files

Today we will talk about steganography which is the science of hiding information inside other data instead of just encrypting it you can think of it as the cousin of cryptography and you can mix steganography with cryptography to have something stronger than both.

In most cases this can be done by changing small number of bytes in a file to contain your data. This can be an image file or sound file.

To make this clear lets see an example.

One of the following images contains the message "My secret message" can you tell which one of them contains the message???





Actually yes, by zooming in the second image you will notice that the image have some colored pixels <u>at the bottom left corner</u> those pixels are the pixels that contains our message.



The following picture shows the image opened with a hex editor (you may use hex work shop in windows or ghex or bless hex editor in linux):

				USA 📭 🖏 🐹 Sat Sep 11, 5:56 PM 🚷 Tady
Factives Home				
File Edit View	n message.omp - GHex v Windows Help			
000000000 000000000 000000022 000000033 000000034 000000034 000000035 000000066 00000005 00000006 00000008 00000004 00000008 00000008 00000000	12 4D DE 8A 09 00 10 1C 02 00 00 82 18 8A 09 00 13 08 10 00 00 4D 79 20 11 67 65 FF FF FF F FF FF FF FF FF	00 00 00 00 36 00 01 00 01 00 18 00 03 00 01 3 08 00 03 73 65 63 72 65 74 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF	00 00 28 00 00 00 00 00 00 00 00 00 00 00 20 6D 65 73 FF FF FF FF FF FF FF	00 BM6(00 BM6(73My secret mess FF age FF
Signed 8 bit:	77	Signed 32 bit: 1931508	045 Hexa	decimal: 4D
Unsigned 8 bit:	77	Unsigned 32 bit: 1931508	045 Octa	115
Signed 16 bit:	31053	32 bit float: 1.271405	6e+31 Binar	y: 01001101
Unsigned 16 bi	t: 31053	64 bit float: 4.768900	De+180 Strea	m Length: 8
	Show little endian decoding	g	Show unsigned a	nd float as hexadecimal
Offset: 36				
		/.	1	KANALAL

Now you can see our message.

Cool right. But what if some one else zooms in the image and he notices these colored pixels and by further examination he determined the message we hide??

One way to solve this is by encrypting the message but still it will be obvious that there's something strange in the image and by some cryptanalysis the cipher can be broken .and i have seen some applications using this technique.

Any way i came up with new idea i don't know if anybody uses this before or not.

What we gonna do is that we will hide the information inside what is called the padding area so first lets talk about what is the padding area (you can skip this part if you are already familiar with the padding area).

What is the padding area??

It's some bytes added to a file or a network packet for 4 byte alignment. Those bytes are added because computer can handles data which are multiples of four bytes faster since the registers and the buses are 32 bits(assuming a 32 bits machine), also one of the obvious examples is your graphics card which needs the frames sent to it to be aligned for four bytes.

Let's examine a simple bmp file and see the padding area (this example was taken from bmp file specifications in wikipedia):

Here's the bmp:



and this table shows how the image stored in the file:

36h	3	00 00 FF	0 0 255	Red, Pixel (0,1)
39h	3	FF FF FF	255 255 255	White, Pixel (1,1)
3Ch	2	00 00	0 0	Padding for 4 byte alignment (Could be a value other than zero)
3Eh	3	FF 00 00	255 0 0	Blue, Pixel (0,0)
41h	3	00 FF 00	0 255 0	Green, Pixel (1,0)
44h	2	00 00	0 0	Padding for 4 byte alignment (Could be a value other than zero)

as you can see in an image of 4 pixels we have wasted four bytes and these bytes will not be rendered by any graphics application and no application cares about their value so we can hide our message on them.

Lets see an example in this 4 pixel image and store the word test in those four bytes and see if the colors changed. One of the following two images contains the message and the other one doesn't contain any thing:



as you can see the colors haven't changed and in the following image you can see the second image containing the word "test" opened in ghex:

子 Арр	licat	ions	Pla	ces	Sys	tem	6		3	4				2/	U	SA 📢) (PÛ		Sat S	ep 11, 1	L0:42 AM	Fad	ly Moh	amme	ed Os	man
						F																						
		-										e	fter	bm) - (Hex								ſ		ล		
(Con	File	Edi	it V	/iew	Wi	ndow	/s ⊦	lelp	_	_								_	_								
		42	4D	46	00	00	00	00	00	00	00	36	00	00	00	28	00	00	00	02	00 BI	ЧF	6	.(
£-,		00	00	02	00	00	00	01	00	18	00	00	00	00	00	10	00	00	00	13	0B.							
Ta	ay.	00 74	00	13 FF	0B	00	00	00 EE	00	00	00	00	00	00	00	00	00	FF	FF	FF	FF.	••••						
	5	/4	05		00	00	00		00	15	14																	
	Ti																											
																									_			
1	true																								-		/	
	F																											
cod	levi																											
		Signe	ed 8 l	bit:	[116					Sig	ned 3	32 bi	t:	116					He	exadec	imal:	74					
		Unsi	gned	8 bi	t: [116					Uns	igne	d 32	bit:	116	i -				00	ctal:		164					1
	1	Signe	ed 16	5 bit:	[116					32	bit fl	oat:		1.62	2550	6e-4	3		Bir	nary:		0111010	0				
ea	al-b	Unsi	gned	16 k	oit:	116					64	bit fl	oat:		5.73	3116	1e-3	322		Str	ream l	ength:	8		÷			
in	ijec				Sh	ow li	ttle e	endia	an de	codi	ka					1	S	now	unsig	gned	and flo	oat as h	exadecima	al				
		Off	set: 4	45																						-		
		actio										15																
extr	acteo	d info	o.txt																									
	/ [G	oogl	e - M	ozill	a Fir	ef		[Do	wnlo	bads	- File	Bro]		after	r.bmj	0 - G	Hex										

How to calculate the available padding space??

you can calculate the available padding space using the following formula(assuming a 32-bit machine):

(number of pixels in a row * number of bytes for each pixel)%4

Where can i use this??

It can be used for hiding information but also it can be used for uploading shell codes in websites that allows image uploading and this can be used by worms instead of uploading the shell code to a server.

If i used it to store a shell code will the antivirus be able to detect that??

I think that antivirus can't detect that since the signature will be changed by changing the image or it's dimensions.