THE TEMPEST METHOD OF COMPUTER DATA INTERCEPTION!

----------by Al Muick for P-80 Systems, OCT 86----------

Let me begin by a brief history of myself. I spent the better part of six years in Uncle Sam's Country Club (better known as the US Army) working in the Intelligence and Security Command (better known as the ASA--Army Security Agency). During that time, my primary duties were Cryptology, Cryptologic Intercept, Counterintelligence, and Field First Sergeant (whatta drag!).

What I'm about to tell you comes under the heading of Cryptologic Intercept. Incidently, for those of you in the know, I was stationed at Field Station Augsburg in West Germany (if you're not in the know, read the book, THE PUZZLE PALACE).

The interception of radiated data from computers and computer terminals is known in the world of the ASA as "TEMPEST." TEMPEST intercept may be accomplished in several ways. One, is via a mobile van with the commo equipment on board, two is via strategicly stationed intercept sites (Field Station Augsburg) and the third, rarely used, is relay from one site to another.

To run a TEMPEST operation, you will need a good communications receiver, both high frequency and very high frequency with adjustable bandwidths and a VFO. If you plan to just intercept and leave the exploitation of the collected intelligence for later, you will need a HIGH-QUALITY tape deck; not one of those cheap-assed portables, but a high quality deck. If you plan to do the exploitation now or later, you will still need to convert the IF of your communications receiver to a recordable frequency. To do this, simply patch the output of your 1 MHz or below IF to the input plug on your tape deck. If your IF is something above 1 MHz you're S.O.L unless you have an IF downconverter around or have the ability to construct one. You will, in effect, be recording an RF frequency on your tape deck, vice an audio frequency.

Your tape deck MUST run at either 7 1/2 or 15 i.p.s in order for it to record this signal. You will later play that signal back into your IF for exploitation.

As soon as you have your intercept station (it is best to use a van) set up with receiver, antenna, and recorder, you are ready to engage your intercept target. Most computers are RF shielded these days, so your receiver had better be damn sensitive and have a very selective bandwidth. If you are planning to intercept such a computer, you will need to be outside its building location (if possible). Since we know, most microprocessors operate at frequencies between 2-12 MHz, we will look for the radiated data here in that frequency range. It is here that a spectrum analyzer, connected to your IF output will aid in discerning the signals and binary emissions of your target computer. If you know how to use a spectrum analyzer, it will prove invaluable, but since they are so complicated, I will not attempt to explain their proper use here.

You will simply scan the bands between 2-12 MHz until you find the radiated signal (if you must, go for the 2nd, 3rd, 4th, etc. harmonics if local interference on the primary frequency is too high) and then tune to the spot where it comes in best. Next adjust your bandwidth until you can just hear the signal as pure as day, with very little to no outside interference.

Once you have your target tuned in, you may want to drive around the block or further away, to avoid detection. Remember, not to go too far or you will lose the signal. Mainframe computers (when unprotected) sometimes radiate a signal for 3 to

four miles! A typical PC computer will radiate a signal for at least 1/2 mile if unprotected!

You should, by now, have picked your intercept site, have parked the van, and have made sure that you still have your signal coming in at good strength. The next step is easy! Simply connect the output of your low frequency IF to the input of your deck and let 'er rip! I find that 10" reels suit this purpose just fine, and you should be able to get at least one or two UIDs or PWs in the amount of time you will have at 7 1/2 or 15 i.p.s. After the tape is done (you may want to record both sides) pack up your gear and head for home!

Once home, you will need another piece of equipment, possibly two. In various surplus magazines, you will see a machine called a "visi-corder" advertised. This is a machine that burns a copy of binary code onto light-sensitive paper. They cost some money, but are basically invaluable. You are now ready for signal exploitation.

You now need to play your recorded tape into the IF input of your communications receiver. The output of your IF will be connected to the IF input on the visi-corder. This will give your the truest binary representation on the paper. If you so desire, you may connect the audio out of your communications receiver to the audio input of the visi-corder. The audio is rectified into DC and then you get a crisp, clear presentation on the paper. But remember this....DC LIES!!! While the representation may be clear, the binary spacing will be off slightly, increasing in error as you continue, until you finally wind up with continuous error.

Assuming you have made the proper connections, get some beer for your relaxation (or them funny l'il pills, or whatever makes you relax....here comes the hair-pulling part). Begin playback of the deck into your receiver and initiate the visi-corder's print mode. I recommend a medium-fast speed, because if you use slow speed to conserve paper (you cheap fucker!), the bauds will be so close together as to render the paper useless and wou wind up wasting the paper anyway!

At this point, print out about 2 minutes worth of paper. Once the paper is printed, expose it to light so it develops and have several 3x5" cards handy. As soon as it develops, scan the paper and the binary stream on it for a section that has three or four of the smallest (closest together) bits. This is ASCII. Once you have found the section, place one 3x5" card at the base of the section and mark off tick marks where each bit stops and ends (on the smallest bits only!!). You are now ready to do what we in the ASA call "bustin' bauds."

As you know, one ASCII byte consists of 8 bits. simply start at a reasonable point at the beginning of your interception and begin to mark off tick marks along the binary stream. Even if you come across 1s and 0s that are very wide, mark as many thin ticks from your 3x5" card on them. This is necessary to break the ASCII code.

The complete 8 bit ASCII code is at the end of this tutorial for your convenience.

Once you have marked off the paper, count off the first eight bits, e.g. 10011101 and refer to the ASCII chart to find a character that fits it. If you can't find one immediately, don't despair! Try using the complement of the 8-bit code in front of you (i.e. the reverse of what you've decoded. Instead of 10011101, try 01100010.). If you still have not found anything, slide your card over one bit and try to get another byte of ASCII. This time you may come up with 00111010

(complement 11000101). Check it with the table. Remeber, you may have to do this eight times (that is, shift a bit over eight times) before you make any sense out of it. It is long and tedious, but it will pay off in the end.

Note: this is illegal and is punishable under federal law. I assume no responsibility for your actions, and neither does the operator of P-80. This is presented for your information only. If you have any questions, please leave me mail!.....happy hacking!....Al Muick.

ASA LIVES FOREVER!!

The 8 bit ASCII code:

(for 7 bit ASCII, simply delete the last bit...it's not always there...something to keep in mind....al)

BINARY	MEANING
00000000	Null
10000000	Start of message
01000000	End of address
11000000	End of message
00100000	End of transmission
10100000	WRU (Who are you?)
01100000	RU (Are you...?)
11100000	Bell (audible signal)
00010000	Format effector
10010000	Horizontal tabulation or skip (for card puncher)
01010000	Line feed
11010000	Vertical tabulation
00110000	Form feed
10110000	Carriage return
01110000	Shift out
11110000	Shift in
00001000	Device control reserved for data link escape
10001000	Device control
01001000	Device Control
11001000	Device Control
00101000	Device control (stop)
10101000	Error
01101000	Synchronous idle
11101000	Logical end of media
10001000	Information separator
10011000	Information separator
01011000	Information separator
11011000	Information separator
11001000	Information separator
11011000	Information separator
11101000	Information separator

11111000	Information separator
00000100	Word separator (space, normally non-printing)
10000100	!
01000100	"
11000100	#
00100100	$
10100100	%
01100100	&
01110100	'
00010100	(
10010100	)
01010100	*
11010100	+
00110100	,
10110100	-
01110100	.
11110100	/
00001100	0
10001100	1
01001100	2
11001100	3
00101100	4
10101100	5
01101100	6
11101100	7
00011100	8
10011100	9
01011100	:
11011100	;
00111100	<
10111100	=
01111100	>
11111100	?
00000010	@
10000010	A
01000010	B
11000010	C
00100010	D
10100010	E
01100010	F
11100010	G
00010010	H
10010010	I
01010010	J
11010010	K
00110010	L
10110010	M
01110010	N

11110010	O
00001010	P
10001010	Q
01001010	R
11001010	S
00101010	T
10101010	U
01101010	V
11101010	W
00011010	X
10011010	Y
01011010	Z
11011010	Left bracket
00111010	Reverse slash bar
10111010	Right bracket
01111010	Up arrow
11111010	Left arrow
00000110	Unassigned
10000110	Unassigned
01000110	Unassigned
11000110	Unassigned
00100110	Unassigned
10100110	Unassigned
01100110	Unassigned
11100110	Unassigned
00010110	Unassigned
10010110	Unassigned
01010110	Unassigned
11010110	Unassigned
00110110	Unassigned
10110110	Unassigned
01110110	Unassigned
11110110	Unassigned
00001110	Unassigned
10001110	Unassigned
01001110	Unassigned
11001110	Unassigned
00101110	Unassigned
10101110	Unassigned
01101110	Unassigned
11101110	Unassigned
00011110	Unassigned
10011110	Unassigned
01011110	Unassigned
11011110	Unassigned
00111110	Acknowledge
10111110	Unassigned control
01111110	Escape

11111110 Delete/Idle