Spying On WINLINK

Gordon L. Gibby KX4Z

July 2019

This text, and many others, are available on https://qsl.net/nf4rc/ for appropriate usages as constrained by various copyrights.

ISBN: 9781080563197

Version 2 July 14, 2019
(Modest improvements from Version 1)

Cover Design
Three winlink stations set up in a hallway
with coax cables heading out to various antennas,
as part of the experiment.

PREFACE

This is the story of how I apparently become the first person to document how to spy on, or crack a WINLINK mode. As you'll soon find out, WINLINK is ***not encrypted***, and so there is NO REASON why anyone couldn't do what I did, or even much better. Since John Wiseman's publicly available source code compiles to send and receive all kinds of WINLINK messages….anyone could take his work and build a monitoring system for WINLINK. The only fly in the ointment is that their compression system compresses over large swaths of text – and so you really need a perfect packet stream in order to monitor WINLINK. That can be arranged, but it does take some effort.

It is embarrassing that there was even a need for a proof like this!

The WINLINK people actually make it **easy** to read any WINLINK message to/from a United State Amateur Radio Operator that passes through their central message system – they provide a free VIEWER where you can literally READ any message. See: https://winlink.org/content/us_amateur_radio_message_viewer
You'll need an account but that shouldn't be difficult. There's no need for any of the software or work that I did in this text – you just read the messages right there!

Now, not everyone follows the rules, and the WINLINK system is so popular that tens of thousands of messages are sent and received every month….so they encourage you to click on a button to report any messages that appear to violate regulations. Those messages are then pulled from the viewer and a process is undertaken to deal with the issue properly.

But that still didn't satisfy some people….and thus this book came to be.

Contact me at: docvacuumtubes@gmail.com

Gordon L. Gibby MD KX4Z
Newberry, Florida
July 14, 2019

DEDICATION

This short recounting is devoted to my long-suffering and wonderful wife, Nancy Gibby, who put up with the kitchen table being taken over many, many times during the development of various systems....my own workbench already covered with other projects. The upstairs hallway has to be cleared out within a week for kids and grand kids arriving!

She's a great example of a Lady who serves her Lord.

Amateur radio operators should aspire to the highest
the radio craft allows:

Do you see someone skilled in their work?
They will serve before kings;
they will not serve before officials of low rank

Proverb 22:29 (NIV)

CONTENTS

ACKNOWLEDGMENTS

A large number of people have mentored me so many many times and ways in the past few years after I returned to active ham radio. It is impossible to name all of them. But the WINLINK DEVELOPMENT TEAM has put in decades of often-thankless work building not only a great working system, but also a large number of amateur operators who gained a vision, and then put in the work to become more capable, better educated and more productive radio amateurs.

1 HOW WINLINK SENDS DATA

I am just a newcomer to the WINLINK system of a few years' experience. So I'll do the best I can to explain how it sends data.

The WINLINK system is a radio version of an email system, including the ability to accept attachments. It operates on a vast range of frequencies and modes – it can be utilized on HF from 160 meters through 10 meters, and it has modes that are widely utilized on HF and VHF. Then it has telnet capabilities suitable for transmission over the Internet or across MESH microwave systems. Our local group in Alachua County has used all of them.

The WINLINK system is completely *volunteer-built* by amateur radio operators over a period of many years. Software is free. Usage is free. They "encourage" donations but only a tiny number of us in my county have sent them funds (which they justly deserve).

It operates in multiple methods, including client-server types of interactions where users connect to a server station ("Radio Message Server" RMS) which makes internet connections to more central systems that move email properly. Another method is peer-to-peer, which can be carried out between two stations using multiple modes It has a third, "radio-only" mode[1], where it is able to provide automatically relayed communications across the world, using HF stations with high performance PACTOR modems; those communications can only be retrieved when a user makes a radio connection. The purpose of that mode is to provide a backup system should the regular Internet be completely unusable.

[1] Version 1.0 of the text used the poorly-chosen adjective "obscure" – the radio-only mode uses precisely the same non-encrypted protocols, but its operation is difficult for people so used to the existence of the Internet, to grasp.

For a volunteer organization to create such a wide-ranging system, which moves approximately 50,000 emails per month, is quite amazing. The WINLINK developers are justifiably proud of the capabilities this system has achieved, and they should also be happy to see all the skills that have been developed in the larger amateur radio community as a result.

To keep such a large system working, it is likely important that unifying concepts are utilized over and over, no matter what the frequency or what the radio protocol happens to be. Why reinvent the wheel for each different situation?

The WINLINK system is able to deal with multiple different radio protocols;

PACTOR I, II, III and IV (the latter only outside the USA)
WINMOR
ARDOP
VARA
PACKET (AX.25)

WINMOR was developed to give users an inexpensive alternative to the expensive hardware-based PACTOR mode. It operates on ordinary sound-card systems. ARDOP and VARA are more recent improvements, VARA being an external protocol written by a completely separate entity. PACKET (AX.25) is quite old, but in recent years it no longer requires dedicated hardware but instead can be operated on simple soundcard hardware.

For all of these modes, the WINLINK system follows a systematic method to pass messages across the radio, which is optimized for efficiency and accuracy, as follows.

A simple check-in looks like this, and seems to have a lot of "tracking numbers" employed:

```
*** Connecting to a CMS...
*** Connected to CMS-SSL at 2019/07/13 14:02:53
[WL2K-5.0-B2FWIHJM$]
;PQ: 77805051
CMS>
   ;FW: KX4Z FLBDR-OPER|30854781 KXFOURZ-EM|77949673
   [RMS Express-1.5.21.0-B2FHM$]
   ;PR: 42373124
```

```
    ; WL2K DE KX4Z (EL89RQ)
    FF
  FQ
  *** --- End of session at 2019/07/13 14:02:54 ---
  *** Messages sent: 0.  Total bytes sent: 0,  Time: 00:00,
bytes/minute: 0
  *** Messages Received: 0.  Total bytes received: 0,
Total session time: 00:00,  bytes/minute: 0
  *** Disconnected at 2019/07/13 14:02:59
```

But that is a case where no messages were transacted in either direction (and the connection was by way of telnet protocol over the Internet.)

Messages, including attachments, when present are packaged up, and assigned identification numbers so that the system can track actions.

One or more messages, with their attachments, are then compressed using a publicly available compression algorithm, developed around 1988.[2] [3] [4] [5]

Compression techniques have been developed in highly theoretical work extending back nearly a century, and exploit statistical redundancies in language data. The history of these developments – so incredibly important in the development of the Internet, operating systems, hard drive storage media, and modern communications – is fascinating. Shannon-Fano coding was invented in 1949 by Claude Shannon and Robert Fano, assigning shorter codes to more-often occurring symbols in a data block, allowing lossless compression. Morse Code (and PSK31) similarly use shorter symbols for English characters that are more likely to occur.[6]

Jump forward two years to 1951, and David Huffman was a student of Robert Fano, and was given the opportunity to do a paper rather than take a final exam – and ended up creating a "bottom-up" probability-

[2] LZHUF, part of the B2F forwarding protocol. See for discussion (and criticism): http://www.danplanet.com/blog/2009/11/09/winlink-1988/

[3] Source code for one implementation publicly available here; https://www.pcorner.com/list/C/LH_UNIX.ZIP/LZHUF.C/

[4] See more discussion of LZHUF's background here: https://en.wikipedia.org/wiki/LHA_(file_format)

[5] The actual source code for the WINLINK compression is freely available on Github here: https://github.com/ARSFI/Winlink-Compression

[6] https://ethw.org/History_of_Lossless_Data_Compression_Algorithms

based encoding that was superior to the compression system which his Professor had developed! The tables in these types of loss-compression systems are often considered as "Huffman tables." In 1977, the LZ77 algorithm to perform compression using a dictionary created by software was created by Abraham Lempel and Jacob Ziv. Huge patent and legal issues resulted from the later widespread usage of compression algorithms that sprang up from all these ground-breaking developments, as they were used in MS-DOS and many, many other situations[7]….. The compression algorithm ("LZHUF") used within WINLINK is a public domain available offspring of these discoveries, made available years and years ago. That obviously avoided potential costs and legal issues for the volunteer developers of WINLINK.

By taking a LARGE chunk of data into the compression algorithm, it is fairly obvious that a better table of probabilities can be developed that results in a more compact message, requiring less usage of the radio time-spectrum for transmission. It would be much more wasteful to take smaller chunks for compression… (However, just as with RAM and Hard Drives….people seem to find a way to use every available resource to the limit they find most advantageous…..a human characteristic unlikely to change.)

The compressed result is then sent in small chunks ("packets") through whatever radio protocol is in use. The technical details of all that are documented by the WINLINK group.[8] Protocols can be generally expected to employ forward error correction (FEC) within the packet but that alone only increases the chance of accurate transfer at the cost of added data and time; *it cannot guarantee accuracy*. Just as amateur operations on any mode from CW to AM to SSB and beyond, from the beginning of radio, have asked for acknowledgments or offered "fills," these digital modes then utilize ARQ practices to find out if the packet arrived safely and to resend it, if it didn't. On a widely-popular internet amateur radio discussion forum, I recently posed a topic asking for the applicability of Forward Error Correction and in dozens of posts, I was unable to find a single commenter who was willing to propose FEC alone (without ARQ) for the transmission of important (non-trivial) communications.[9] It seems that *almost every amateur* understands that "fills" are a normal part of ham radio communications, whether it be

[7] https://ethw.org/History_of_Lossless_Data_Compression_Algorithms

[8] See here: https://winlink.org/B2F

[9] https://forums.qrz.com/index.php?threads/forward-error-correction.665519/

contesting, regular rag-chews, or messages of importance. [10]

The published data on the Pactor III protocol indicates that it uses a forward error correction technique involving an optimum rate 1/2 convolutional code with a constraint length (CL) of 7 or 9.[11] [12] This is obviously pretty technical stuff!

At the other end of a connection, the packets are now guaranteed to be correct (thanks to ARQ) and using the inverse public-domain decompression, the message (s) and attachment(s) are reconstituted and appropriately handled.

To get any more precise details of the exact process beyond that above, would likely require perusing the publicly available operating code from John Wiseman, which I haven't yet done, but the above discussion suffices to hit the high points.

Notice that there is NO ENCRYPTION anywhere in the system, no matter whether the radio protocol is PACTOR or PACKET or WINMOR or any other protocol. That's because the FCC forbids intentional obscuration of messages on the amateur bands.[13] The WINLINK system actually turns off a compression system utilized within Pactor modems in favor of its likely-superior LZHUF compression, used on every one of its modes.

Since there is no encryption, the main obstacles to snooping on these messages include:

a) For PACTOR, since the manufacturer cannot obtain a software

[10] Phil Karn was moved to rebut an earlier commenter who claimed that commercial wireless systems used FEC without ARQ. https://ecfsapi.fcc.gov/file/10513525129724/rm11831-rebuttal-to-rappaport.pdf Mr. Karn developed the link layer of one of the early systems and thus had personal knowledge.

[11] The Pactor-III Protocol, https://www1.scs-ptc.com/download/PACTOR-III-Protocol.pdf

[12] See http://suraj93.github.io/files/ecc-convcodes.pdf for one discussion of convolutional codes as FEC.

[13] FCC Regulation97.113(a)(4), which prohibits: **(4)** Music using a phone emission except as specifically provided elsewhere in this section; communications intended to facilitate a criminal act; messages encoded for the purpose of obscuring their meaning, except as otherwise provided herein; obscene or indecent words or language; or false or deceptive messages, signals or identification.

patent in Europe and thus keeps the internal software as a trade secret and only releases the details of the modulation systems employed – you probably need to purchase a PACTOR modem. They are cheaper on Ebay (used) than buying new….and in either case are FAR less expensive than MIL-STD protocol devices….or some cell phones!

b) For the other modes, you will need either a soundcard system or a commercial TNC system

c) Because the compression is over large swaths of characters….you need to capture a **perfect copy of all the packets** in order to properly decompress them. Every receiving station in the WINLINK station obviously accomplishes this (or the communication fails).

The latter point is important – if you happen to be listening in on a WINLINK transmission, and you do not have a strong signal from them – you are likely to miss some packets….and you may not miss the same packets that the intended receiver misses (and gets retries for) --- so you're going to be out of luck because with anything missed….you won't be able to decompress the message.

How to overcome that? If you are serious about it, and you REALLY want to spy on WINLINK, for whatever reason, here is one method to achieve your goal – it requires "diversity receivers" or "diversity antennas".

You need additional reception sites, from whom you may acquire signals. An example would be Internet-available SDR receivers[14], but you could accomplish the same thing with a network of volunteer monitoring receivers, making their output available by way of the Internet.

Then you need to write software to capture the stations you want to surveil, and to find the BEST packet at each time from the group of volunteer stations you've managed to be able to use. That is a diversity receiving system…..

And that basic sort of thing has been done for a very, very long time. The NAVY was doing it (in a limited sense) back when vacuum tube

[14] Websdr currently lists 166 internet-available SDR's: http://websdr.org/

receivers were all the rage!

"Diversity Receiving Equipment" AN/FRR 3A[15] photo from: http://navy-radio.com/rcvr-div.htm

So there is nothing technologically overwhelming about it, but it will take you some work.

Once you assemble all the packets in order, then you apply the publicly available decompression and bingo! You can read that email.

The reason this book was written was because for some reason, it seemed to me that quite a few people thought that accomplishing this was **completely impossible.** To an engineer skilled in the art (not me, but many others who have discussed this with me), it looks VERY possible, and far more difficult things have been routinely done in communications over the years…. So I set about thinking of how I could make some proof-of-concept demonstration that would illustrate the fact that is is only an engineering project, not a Nobel prize target.

One way would be to take John Wiseman's code, rework it so that it takes advantage of the monitoring mode that all SCS PACTOR modems have[16], add software to correctly line up the received packets, and then

[15] The manual for this diversity receiving gear, dated 26 June, 1944, can be studied here: http://www.tmchistory.org/PressWireless/manuals/prewi_frr-3a_manual.pdf

[16] "PACTOR monitoring mode is available in all our modems." contained in https://ecfsapi.fcc.gov/file/10417301289214/SCS_FCC_Comment_RM11831.pdf

send them through the various subroutines that John Wiseman already wrote to get back to the original message.

That would be a nice engineering project, but it has been a decade or more since I did any significant C-coding, so I wasn't that enamored with the project.

The ready availability of *source code* that already handles thousands of WINLINK emails per month is important to emphasize. Anyone who wished to develop the software to make radio-spying on WINLINK a simple matter would be wise to take advantage of code that literally passes these messages one right after another.

Source Code at github:	https://github.com/g8bpq/LinBPQ
The portion that deals with compression and decompression	https://github.com/g8bpq/LinBPQ/blob/master/lzhuf32.c
John Wiseman's home page	http://www.cantab.net/users/john.wiseman/Documents/
Map of stations utilizing bpq code	http://www.ve9sc.com:81/NodeMap.html

But I have other responsibilities in life, and building a radio-spying system to view messages *that are already available on a web-viewer*….is not one of my highest priorities.

Then it hit me – the WINLINK system itself receives these messages all the time. Wouldn't there be a way to make a demonstration by simply using WINLINK software itself to snoop on an ongoing WINLINK exchange? That would be an engineer's "quick and dirty" way to demonstrate the fact that it COULD be done. Without writing a single line of code, one might be able to prove the ultimate success of the coding project described above.

There are two issues there:

a) You have to spoof your identify and instruct the winlink software that you are the same person as the person who is supposed to be receiving the message. That part is easy if you have control of three stations in front of you.

b) You have to keep your system perfectly synchronized with the actual intended receiver, so that you receive packets when they receive packets (right down to a millisecond or so) since you can't transmit and get the sending station to sync with you. **That part is largely luck!!!!**

None of that is necessary if you take the time to write the code upgrades to John Wiseman's code to make it into a snooping system – but if you want a quick and dirty proof that there is nothing stopping that code-writing project from succeeding – then accomplishing (a) and (b) above even **ONCE** provides all the proof that is needed.

Succeeding at (b) involves no small amount of LUCK….but already I accomplished it months ago and filed an FCC comment to document it. As you'll read in Chapter Two, I re-created that experiment for further proof. So the proof is done. It is possible to spy on WINLINK. There is nothing encrypted. The next chapter will detail HOW I did it. Put some good engineering to work on this, and success is in your future. If you really think it is important. Most of us will just use the free message viewer on the web….

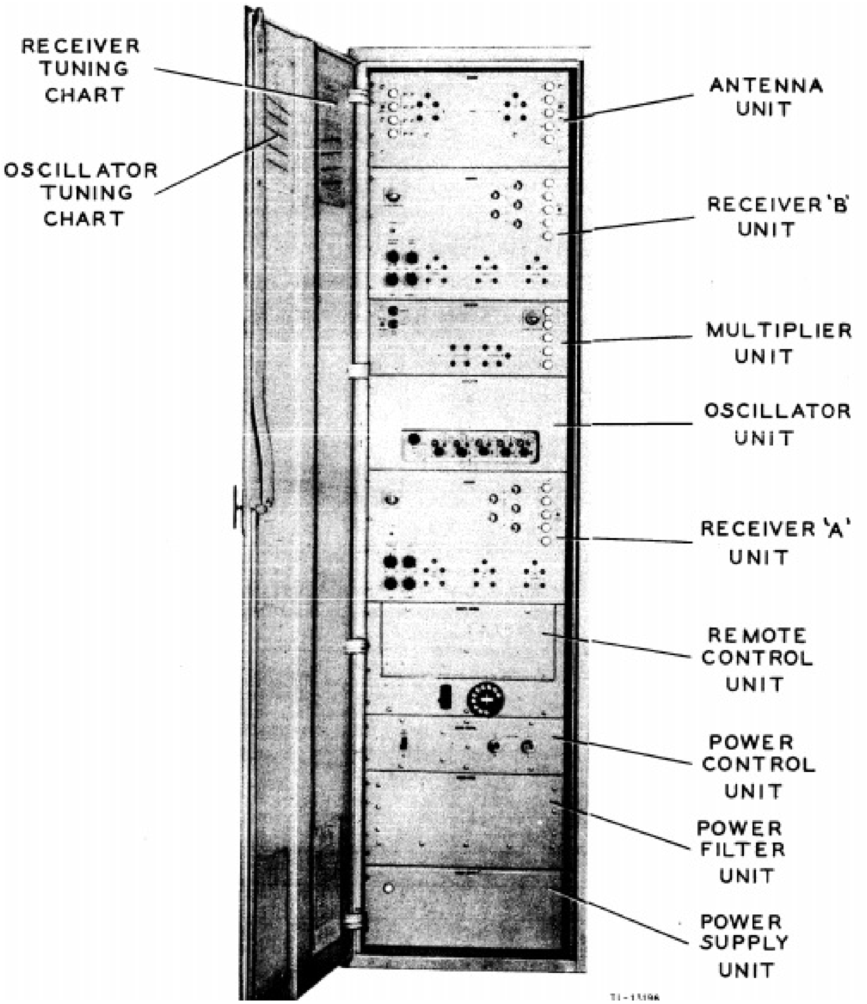


Photo of the makeup of a 1944 Diversity Receiver. From: http://www.tmchistory.org/PressWireless/manuals/prewi_frr-3a_manual.pdf

2 SUCCEEDING AT SPYING

My first effort at this was quite lucky and succeeded almost on the first try. Since it was only a proof of concept trial, I didn't proceed any further, but documented it in a filing with the FCC, amazed that no one had documented this before

(From a submission dated April 9, 2019)

> *A very simple proof of concept experiment was carried out. To prove the concept that using free available or simple software to decode WINLINK messages without actually participating in the back-and forth packet acknowledgments requires, for the easiest case, a stream [of] uncorrupted packets. To create that situation, I utilized two amateur radio stations, using two different call signs, in peer-to-peer PACTOR communications on a legal frequency using minimal power and outside antennas on the same property. I then took freely available WINLINK EXPRESS software on another complete WINLINK computer/radio/pactor modem system, and instructed it to attempt to monitor a peer-to-peer message passage as if it were the intended recipient – but it was refused transmit capability by having connection only to a shielded Heathkit dummy load and minimal power, rendering its signal capture and transmission 60 dB below the incredible signals the other two stations were experiencing. The radios involved were simple used ICOM 718's and an older 725. Lacking any ability to make any changes to the software (since I was not using the freely available BPQ code in this experiment) I set transmit delay times to longer than normal so that I could have a reasonable chance of capturing the full packet stream as a diversity receiver system would. The two intended participants easily established communications with the screen indicating very high speed transfer. The third (monitor) station however was able to capture a workable signal and went*

> *about its business, fooled into thinking it was part of the conversation using the freely available WINLINK EXPRESS software. Hilariously, the largest problem I faced was running between two receiving stations to answer a dialog box asking whether I wished to accept the proffered message – I enabled the secret monitor station and the intended recipient and the message sped past at blinding speed. This entire experiment took about 30 minutes to set up and complete at my home.*
>
> *I was rewarded by having* ***both the participating, intended recipient station AND the secret monitor station capture exactly the same message ("#4 in my subject line") as shown in the photos below*** *and confirmed by their winlink software lengthy assigned message numbers*[17]

Notice that for that first successful test, I utilized a dummy load for the snooping station to render its transmitter ineffective compared to the behemoth signal from the intended recipient.

Why Peer to Peer?

Testing this proof of concept in WINLINK peer to peer mode (rather than a connection to an RMS) utilizes precisely the same data compression algorithms and the same PACTOR modems or other systems. But it has a huge advantage for my proof of concept: The human starts off ONE station (the transmitting one), not TWO stations, as you have to do if trying to spy on a station connecting to a Gateway. I'm not certain I fully understood what an advantage that is...but for the proof of concept test, this makes it much more likely to succeed based on limited testing I've tried.

Not Convinced

Somewhat surprisingly, some remained unconvinced that on the air monitoring of WINLINK was possible. That is odd – it only takes ONE experiment to prove something is possible; it takes an eternity for experimental evidence alone to prove it is not. However, having been asked to further document this feat, I set up the system again.

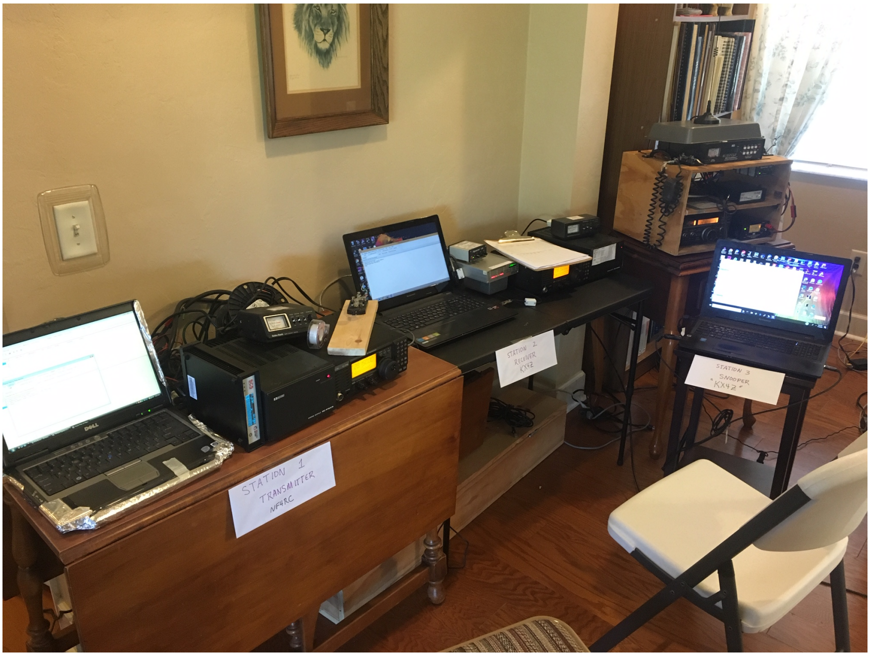
But this time it dawned on me that, rather than using the dummy

[17] https://ecfsapi.fcc.gov/file/10410170249078/FCCRM11831-4.pdf

load on the Snooping Station, I could simply snip the Push-To-Talk wire from the pactor modem to the Icom snooping transceiver, and the computer and Pactor modem would continue to think they were transmitting, when in fact the transmitter would never come on. So I tried that for the more recent tests. I verified that when instructed to transmit the red transmit light on the snooping transceiver did not turn on at all, and no transmissions occurred. This meant I no longer needed the dummy load. At first I used a 2-meter mag mount antenna….but I still had what seemed to be excessive signal, so I left the transceiver only connected to a MFJ Balun.

Three HF radio systems were set up, each with a pactor modem, as detailed in the following table:

Station 1 (“Transmitter”)	Icom-718, Icom PS-15 power supply, SCS DR 7800 pactor modem; Dell D630 computer, VISTA operating system, Winlink Express 1.5.19.0 Coax going to LDG600 tuner, to balanced feedline to non resonant dipole east side of house; NF4RC
Station 2 (“intended recipient”)	Icom 718, Icom PS-15 power supply, SCS PTC-II pactor modem with P3 license upgrade, Lenovo G50 computer, Windows 10, Winlink Express 1.5.21.0, coax to LDG600 tuner, to balanced line to non resonant inverted V on South side of house
Station 3 (“snooping station”)	Icom 725, powered by MFJ 4230MVP PTC-IIIusb pactor modem; Lenovo computer, Windows 10. Express 1.5.21.0; antenna is a TRAM 2m/70cm connected through MFJ isolator Balun to radio; PTT line cut on wire to ACC #1 Socket. [Note on Day 2, was switched out for an Icom 718.]



Three radio stations with computers and pactor modems. Far left is Station #1 ("Transmitter") On the black table is Station #2 ("Intended recipient") and at the far right is Station #3 ("Snooping Station")

Hoping that it would give me a better chance to keep these stations in sync, the PACTOR transmission delay was set to 80 milliseconds instead of the usual 30 milliseconds. The smallest messages possible were used – simple texts like "This is message #2" because only luck keeps these stations in sync.

Intending to avoid causing interference to other stations, I attempted to perform the experiment on 10 meters where there would be little to no ionospheric propagation and after a fruitless hour discovered that the Snooping Station's ancient Icom 725 just wasn't accurate enough on 10 meters to easily make even a normal pactor connection, and worse, it had some offset between its transmitting and receiving frequencies. That forced me back to 80 meters.

Despite using the lowest output power possible from Station #1 and Station #2, I still had S-meters that were at the very top, and communications were not very reliable. By turning on attenuators in the radios and reducing the WINLINK internal PSK and FSK transmitter levels to very small numbers (50), I was able to finally get normal pactor communications to work with Station #1 and Station #2 using outside antennas at opposite ends of my house. Station #3 did better when I completely disconnected even the 2-meter mag-mount (untuned) antenna I provided it for receiving; this allowed the S-meter readings finally to get below S9.

The sequence of a test was as follows:

1. Set both Station #2 and Station #3 (Snooping Station) to be ready to be called for a Pactor Peer to Peer WINLINK transfer, on the same frequency as Station #1.

2. Click "START" on Station #1's winlink PACTOR peer to peer session, and sit back and watch whether it worked.

After I got the signal levels DOWN enough[18], the first three attempts were all successful. For the first two, WINLINK preferences were set for me to have to request downloads of proffered files – so I had to simultaneously click a button on the computer screens of Station #2 (actually in the contact) and Station #3 (the snooping station, unable to transmit, but thinking it was part of a connection). This was unnerving, and on the 2nd attempt I was off by more than 1 second on one of the computers, yet it still worked.

I then switched the preferences so they would not require approval for accepting downloads and the next email transfer was also a success. Subsequent efforts failed several times and I tried various adjustments without success.

A successful transfer – there were now FOUR of them, counting the earlier FCC submission – meant that a snooping station without any effective means of transmitting at all, was able to receive a winlink compressed message using PACTOR, and successfully decompress it to get back the exact message sent – just like the intended receiver did at the same time.

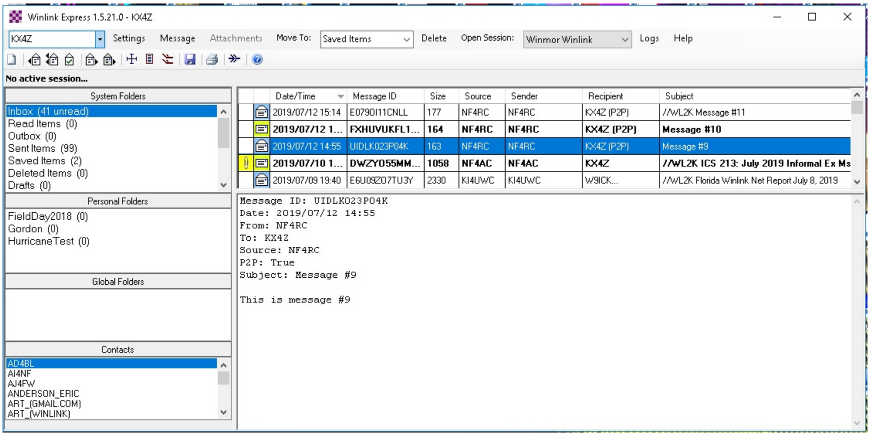
The characteristics of the 3 successful messages received simultaneously by the intended receiver and the "snooping station" are

[18] In addition to setting the power level of the IC-718 transceivers to "L" or lowest possible power, I had to decrease the PSK and FSK transmitter settings within the WINLINK software to 50 and 50 respectively.

given in the following table:

Message identifying number, received by both intended and snooping station	Size, uncompressed and compressed
U1DLK023P04K	157 compressed to 144
FXHUVUKFL14K	160 compressed to 144
E079OI11CNLL	166 compressed to 150

(Note that all of these are P2P messages and won't appear in the WINLINK internet-based viewer.)

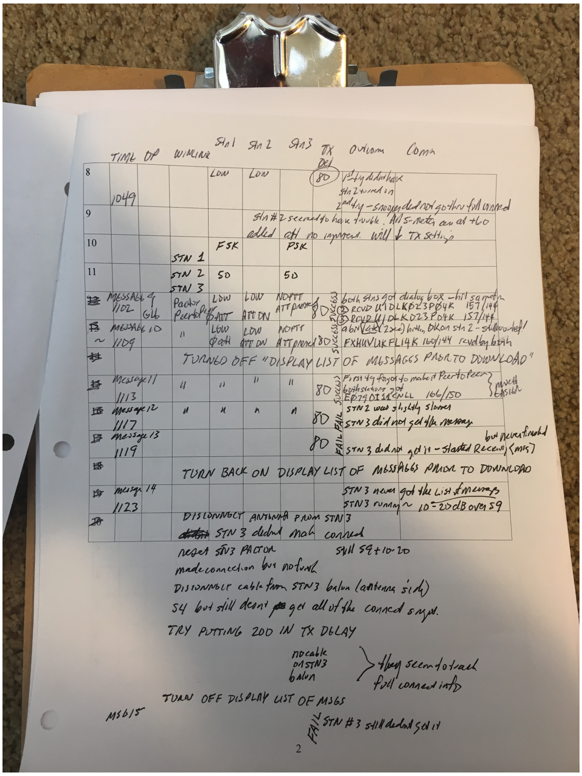


Date/Time	Message ID	Size	Source	Sender	Recipient	Subject
2019/07/12 15:14	E079OI11CNLL	177	NF4RC	NF4RC	KX4Z (P2P)	//WL2K Message #11
2019/07/12 1...	**FXHUVUKFL1...**	**164**	**NF4RC**	**NF4RC**	**KX4Z (P2P)**	**Message #10**
2019/07/12 14:55	UIDLK023P04K	163	NF4RC	NF4RC	KX4Z (P2P)	Message #9
2019/07/10 1...	**DWZYO55MM...**	**1058**	**NF4AC**	**NF4AC**	**KX4Z**	**//WL2K ICS 213: July 2019 Informal Ex Ms**
2019/07/09 19:40	E6U09Z07TU3Y	2330	KI4UWC	KI4UWC	W9ICK...	//WL2K Florida Winlink Net Report July 8, 2019

```
Message ID: UIDLK023P04K
Date: 2019/07/12 14:55
From: NF4RC
To: KX4Z
Source: NF4RC
P2P: True
Subject: Message #9

This is message #9
```

Screen Capture from Snooper Station Computer, showing Message #9, one of three successfully snooped on July 12 2019.



	TIME OF		WINLINK	Stn 1	Stn 2	Stn 3	TX Del	Outcome	Comm
8	1049			LOW	LOW		80	1st try didn't have stn 2 turned on 2nd try - snooper did not go thru full connect	
9				stn #2 seemed to have trouble. All S-meters are at +60 added att no improvement. will ↓ TX settings					
10			STN 1	FSK		PSK			
11			STN 2 STN 3	50		50			
~~12~~	MESSAGE 4 1102	Glb	Pactor Peer to Peer	LOW ØATT	LOW ATT ON	NO PTT ATT parked	80	SUCCESS	both stns got dialog box - hit same btn ② RCVD U1DLKØ23PØ4K 157/144 ③ RCVD U1DLKO23PO4K 157/144
~~13~~	MESSAGE 10 ~ 1109		"	LOW Øatt	LOW ATT ON	NO PTT ATT parked	80	SUCCESS	abort (2x) button, OK on stn 2 - still worked! FXHUVUKFL14K 160/144 rcvd by both
~~14~~			TURNED OFF "DISPLAY LIST OF MESSAGES PRIOR TO DOWNLOAD"						
~~15~~	Message 11 1113		"	"	"	"	80	SUCCESS	First try forgot to make it Peer to Peer } much easier both stations got EØ79OI11CNLL 166/150
~~16~~	message 12 1117		"	"	"	"	80	FAIL	STN 2 was slightly slower STN 3 did not get the message
~~17~~	message 13 1119						80	FAIL	STN 3 did not get it - started Receiving (msg) but never finished
~~18~~			TURN BACK ON DISPLAY LIST OF MESSAGES PRIOR TO DOWNLOAD						
~~19~~	message 14 1123							STN 3 never got the list of messages STN3 running ~ 10-20dB over S9	
~~20~~			DISCONNECT ANTENNA FROM STN 3 STN 3 didn't make connect						

reset STN3 PACTOR still S9+10-20

made connection but no further

DISCONNECT cable from STN3 balun (antenna side)

S4 but still doesn't get all of the connect msgs.

TRY PUTTING 200 IN TX DELAY

no cable on STN3 balun } they seem to track full connect info

TURN OFF DISPLAY LIST OF MSGS

MSG 15

FAIL STN #3 still didn't get it

2

Experimental data capture documenting successes and failures at the spying experiments. Data from July 12, 2019

DAY TWO: Saturday July 13, 2019

Leland Gallup AA3YB (retired Army judge), Jeff Capehart W4UFL (Alachua County EC, NFL Assistant Section Manager) joined me to work together on the WINLINK spying proof-of-concept demonstration.

For perhaps the first hour we discussed the way WINLINK works, answering many questions that Leland had, trying to fathom the arguments related to any security risk (compared, to, say TOR). We went over very carefully why a real monitoring effort would proceed with new software development using the PACTOR monitoring capability...NOT using the demonstration technique we were about to try, which is just a quick and dirty way to prove there is no inherent reason you couldn't succeed by building on code from John Wiseman, freely available. These two gentlemen watched as all three stations were configured and saw how transmissions from the "snooper station" were prevented by interrupting the push-to-talk wiring.

We then had no success getting the snooper to lock -- and even when we shut down the intended station and worked with the snooper (then allowed to transmit) – frequency offsets displayed by WINLINK software on opposite sides of the link seemed to indicate transmit-receive offsets within the older ICOM725, or perhaps an uneven passband. We were learning a lot more about which pactor lights need to be "green" simply from experience! We finally gave up and concluded we had to have another IC-718 -- which normally connect with only a few Hz error displayed. We pressed my backup IC-718 into service, and quickly rigged the required 13-pin DIN wiring to the pactor modem. Allowing it to transmit, we demonstrated a very good lock and minimal offset frequencies – far better than the IC-725.

However, for the demo to work, we still had to have an ancient PTC-II and the newer PTC-IIIusb stay in perfect lock-step for an entire contact.

After disconnecting the push-to-talk wire as before, we proceeded with the same snooper tests. Leland Gallup wrote all the messages and was in charge of initiating transmission. Multiple times the snooper station would lock and track for some portion -- and then lose the lock. This is possibly due to extreme age or calibration differences between the two receiving modems. However, it was **so close to success** so many times that we persisted, and switched to "display messages before download" with me trying to click both computers simultaneously at the dialog box, and at approximately 1220 both the snooper and the intended recipient captured a short message completely intact and of course it was then perfectly identical to the intended recipient's copy. The message is copied below:

```
Message ID: L1WEXNLKQTY6
Date: 2019/07/13 16:20
From: NF4RC
To: KX4Z
Source: NF4RC
P2P: True
Subject: test message number seven

this is test message number seven.
```

I'm typing on the computer that was the snooper station computer, so the copy above is from the snooper station reception. **That makes the 5th message I've now snooped.**

As discussed thoroughly in Chapter 1, this is a **proof-of-concept,** carried out without writing so much as a single line computer code, that has now demonstrated forever that there is no fundamental reason why WINLINK cannot be intercepted. Simply using monitoring mode, diversity reception, and voting to obtain an accurate packet stream and then the already-extant winlink routines to expand the message....would be a **far more guaranteed method--** it just requires some programming effort!

We made a couple of videos, the first of which was too long to email, and a very short 2nd video was able to be emailed which we sent out to several friends.

WINLINK CAN BE SPIED UPON
We proved it without writing even a single line of computer code.

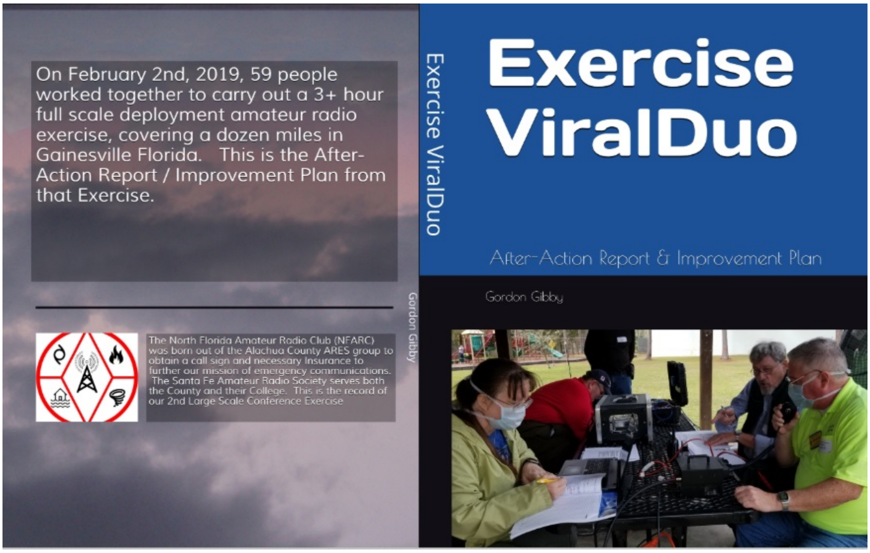
Leland Gallup AA3YB (foreground, retired Army Judge) and Jeff Capehart W4UFL (Alachua County Emergency Coordinator and Asst. Section Manager), witnesses to the final success!

ABOUT THE AUTHOR

Gordon L. Gibby is a practicing physician with previous education in Electrical Engineering. As kids grew up and moved off, more time became available for amateur radio and other hobbies to be resumed. As a high school student, he learned construction from venerable Heathkit vacuum tube transceivers (one of which is still in routine usage -- even for Winlink!) as well as high power vacuum tube amplifier design.

His current amateur radio interest is Emergency Communications, particularly the high speed data communication possible with modern digital software, although the unique possibilities of JS8Call are also quite interesting.

He is currently very involved with the Alachua County, Florida ARES ® group, and with the North Florida Amateur Radio Club. The group is very active, teaching licensure classes, radio theory, construction projects, operating skills, and holding full-scale HSEEP-compliant exercises. Much more information about the group can be found on their simplistic web page: https://www.qsl.net/nf4rc/



On February 2nd, 2019, 59 people worked together to carry out a 3+ hour full scale deployment amateur radio exercise, covering a dozen miles in Gainesville Florida. This is the After-Action Report / Improvement Plan from that Exercise.

The North Florida Amateur Radio Club (NFARC) was born out of the Alachua County ARES group to obtain a call sign and necessary Insurance to further our mission of emergency communications. The Santa Fe Amateur Radio Society serves both the County and their College. This is the record of our 2nd Large Scale Conference Exercise

Exercise ViralDuo

Gordon Gibby

Exercise ViralDuo

After-Action Report & Improvement Plan

Gordon Gibby

A full scale exercise carried out and published by NFARC.