DECISION SUPPORT SYSTEMS, inc.

DSSI *METATEMPO: SURVIVING GLOBALIZATION*

SECURE COMMUNICATIONS OPERATIONAL TRADECRAFT

"HOW NOT TO BE SEEN"

11 JANUARY, 2002

DECISION SUPPORT SYSTEMS, INC.

INFO@METATEMPO.COM

HTTP://WWW.METATEMPO.COM

PURPOSE

> *Necessity is the plea for every infringement of human freedom. It is the argument of tyrants; it is the creed of slaves.*
>
> *—William Pitt, British Prime Minister, November 18, 1783*

Trying to be 'all things to all people' can be disappointing for everyone involved—the writer can't possibly satisfy every possible reader, nor can every reader find exactly what he/she/it is looking for. It is with full awareness of the problem that this document is written—an attempt to inform the layperson about secure communications tradecraft in the context of Al-Qaida. This will not satisfy the cypherpunk, nor the military, intelligence, or law enforcement reader, because an attempt will be made to remain objective and tell both of those sides of the story.

The author's personal political position is as such:

- Technology provides many options to make privacy achievable—at a cost, but the fact remains that attempts to control the technology are a 'losing proposition'

- Attempts at control of critical technology—cryptography, steganography, etc.—impede integration of such technology into hardware and software systems, with two deleterious effects:

    - Lack of integration means 'ease of use' affordances aren't available

    - Lack of integration means the technology isn't reliably available as a basic service (operating system calls accessing strong, unescrowed cryptography on every computer motherboard), leading to on-going vulnerabilities that could be solved

- The tradecraft described herein is already in use, so 'denying' it to individuals and organizations isn't possible; the technology is already available

- The tradecraft itself can be used to expose networks, such as Al-Qaida, that may be using it to protect their operations

Those familiar with the cypherpunk position will recognize the first three points; those familiar with the position of military, intelligence, and law enforcement will understand the importance of the fourth point. Perhaps this document will allow each side to appreciate the viewpoint of the other, or it may simply motivate a collective of sides against the author (which is where the subtitle of this document becomes relevant—those familiar with Monty Python will grasp the reference, and appreciate the probable outcome).

Presented in this document:

- The purpose of secure communications operational tradecraft (SCOT)

- A 'best practice' process walk-through for a sample of SCOT

- Possible weaknesses and points of attack on SCOT
- How SCOT, if used by Al-Qaida, can be turned against them to identify covert operators
- Conclusions

In support of the position of this paper, the author presents some additional quotations to establish the mood:

> *Probably the most controversial issue about widely available secure communications is that the same technology can be employed for legally and morally questionable purposes. It has been claimed, for example, that free application of cryptography enables drug traffickers and terrorists to communicate in secret, without the law enforcement officials being able to intercept their messages. In some countries, strong encryption has been banned or the keys have to be escrowed for government officials. With invisibility readily available to anyone with moderate programming skills, it is obvious that any such measures are ineffective. Restrictions on encryption cannot stop criminals from using, but may hurt law-abiding businesses and individuals who could greatly benefit from mass application of cryptographic techniques.*
>
> *—Hidden in Plain Sight—Steganography, Counterintelligence News and Developments, National Counterintelligence Center, Volume 2, June 1998*
> *http://www.nacic.gov/nacic/news/1998/jun98.htm#rtoc4*

> *Just the minute the FBI begins making recommendations on what should be done with its information, it becomes a Gestapo.*
>
> *—J. Edgar Hoover*

> *I would rather be exposed to the inconveniences attending too much liberty than those attending too small a degree of it.*
>
> *—Thomas Jefferson*

PURPOSE OF SECURE COMMUNICATIONS OPERATIONAL TRADECRAFT

Secure communication means a number of things, some of which are difficult to accomplish:

- Make the content of a message unreadable to parties other than the intended one(s)
- Make the meaning of a message inaccessible to parties other than the intended one(s)
- Avoid traffic analysis—don't let other parties know that a connection exists between the communicating parties
- Avoid knowledge of the communication—don't let other parties know the communication channel or pathway exists

Various component technologies are available that contribute toward the goals:

- Cryptography, systems that use transformation processes to turn 'signal into noise,' by obscuring the **symbols** used for communication
- Coding, systems that substitute or alter **meaning**, and thus hide the real message

The popular package Pretty Good Privacy (PGP) produces this sort of output:

```
-----BEGIN PGP MESSAGE-----
Version: PGPfreeware 6.5.8 for non-commercial use <http://www.pgp.com>

qANQR1DBwU4Dou0bwSIJMtcQCACu6aZ6AVbqzHI8HdsDcHxfdh1p3hkGwtpHUQAW
GiSuXNiPa/x2rKl/LbTUKu/OagqtynAzjhdqiFYEtdwz5Suhsd5y8juqc/cSx+js
jiaGdb0/cmxiWTxocg3wTpzhPxErqVBdfjITKJn+eT+HyhwWT6er1/3GEv/fy73x
YECLqqDJmdPDgIENoIbkxh3zs13SSiAxYgyCZrjanozrxz8QCMEEMntOZJIx0xKW
dOSJBsrIH+UK3BdzjNuTvH57Mq8IvUCXqWa6Lvse4sQImeZlyDMQsjyN38QHXKER
lh2CvgD+H3HlNJWKSqbvDn+JNF4fiAzDgdOTeakh1vyqhcHZB/wKmr4MuDaFyQbU
vYTqoiXiENVTvywcsJJepJSxZHtk8jRtEw+h0y/W6ISfrf5k4ZEA1l8b9NGiFTGF
Op5SZ1dIecowob4L+8sttAjal7rSxo7myQoz9cg4N8pK0lf4z5xwoMEKbauACoLQ
gjRd/5PfkEgu99rLVEmd6iIeHhdjov/90r6l99QeYHABVDV+gcHPRQkqlHdAyIqa
Un6075nC64kgj+Qii6/hDYaN0DQgRJLKNVUXDVUIK8iDuqRxao+sTuq9tYfo6+6K
CsAzym+9TKkFz4iFqZDT3Pj6eL5+Wqtf5J7w4gyjw/WOg79b2ZNI1q6JNY2xXzkX
5z+8VCpqyVdaCTo0wfRaIoJO1TEwOJEQendxrtqgQP4kpX594uNLaFM8vB89KHyS
2rEzFTjyzPPd8JZDEyx/pCDT7KolYBtTe4mUHFq/YdnI/BOx0FIu/kwbCLR2oAo=
=3bFQ
-----END PGP MESSAGE-----
```

The symbols and words used to communicate have been transformed into a meaningless (unless you have the key—'reader makes right') block of ASCII. Put this into a standard email message and it's pretty obvious that you and the recipient have something you don't want known.

A code wouldn't 'break the connection' between people communicating, but it could obscure or hide the meaning. Just what does the phrase "my dog just learned how to shake hands" mean, besides the potentially legitimate interpretation? Why 'my' and not 'my <insert family member, neighbor, name,

etc.>'? Why a 'dog' and not a 'cat' (can cats 'shake' hands?)? Why 'shake hands' as a trick, and not 'roll over,' 'stand on his/her hind legs,' or 'catch a frisbee'? Coding mechanisms, if they maintain a proper grammar, are consistent, and fit a plausible pretext, are incredibly powerful 'information hiding' mechanisms. Al-Qaida's tradecraft manual makes a number of references to coded language, and transcripts that have become public from intelligence operations show an on-going use of code.

As interesting and 'strong' as codes are (breaking good coding systems requires a working body of material for analysis, or human intelligence), cipher systems still have greater flexibility. Strong coding systems require pre-arranged mappings of meanings (what symbols or words translate to what), or at least pre-arranged mechanisms to derive the mappings (e.g., book codes). Cipher systems don't require such pre-arrangement and pre-planning (coding vocabularies need to encompass the range of possible communication in advance), and they have the advantage of being susceptible to the application of technology (such as PGP).

Whether cipher or code, secure communication tradecraft remains incomplete—the transaction between the sender and receiver hasn't been 'broken' yet. Again, a number of technologies have been implemented (largely by cypherpunks) to try to 'break' the transaction connection:

- Remailers, including very secure remailers such as Mixmaster, which: take ciphered messages; layer more ciphers on the message contents; bounce the message around using multiple relays; attempt to 'hide' the relay chain by padding, repackaging, and false traffic; finally delivering the message. This process does not assist a 'monitored' sender or recipient, since it will still be obvious that a message was passed; Mixmaster also requires a large body of relays and other support to work well, something it has never properly enjoyed, and which leaves a relay network susceptible to monitoring

- Cut-outs, public or disposable mechanisms that provide 'anonymity' through sheer volume of users/transactions and little or no connection to the 'real' you. Such web-based email systems as Hotmail and Yahoo were used by the 11September2001 sleepers to communicate with each other or with the Al-Qaida command hierarchy according to media coverage. Other cut-outs are available—instant messaging (ICQ, AIM, etc.), chat rooms (AOL, IRC channels, MUDs/MUSHes, etc.)—but such systems work 'best' from stable machines, and U.S. cover operators appeared to rely heavily on public machines (libraries, cyber-cafes, etc.). Even more secure email systems such as Hushmail 'expose' the side of the transaction connected

- Dead-drops, locations (including 'virtual' ones) that are readily available to anyone and where things can be left easily, are a favorite mechanism in intelligence: public lockers; drop boxes; and now in the age of the Internet, USENET newsgroups or public websites. Key features of dead-drops are that they are public, they are deniable ("that isn't mine"), and they are plausible—a pattern of behavior can be developed that, even if monitored, keeps communication below 'observational threshold,' and possibly prevents 'hostile' knowledge of communication taking place at all

'Best practice,' then, would utilize ciphers or codes, some sort of 'intermediation' (such as remailers or relays provide), utilization of a cut-out, and certainly a dead drop.

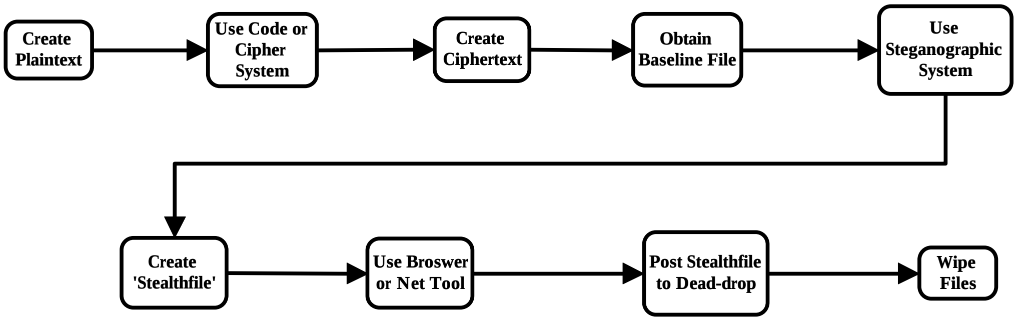
‘BEST PRACTICE’ TRADECRAFT WALKTHROUGH

The goal: establish a communication path that avoids traffic analysis, gives a fighting chance of avoiding detection of the existence of the communication channel, breaks the ‘transaction’ and protects the other side of the transaction even if monitored, employs the best ‘off-the-shelf’ plug-and-play technology, utilizes strong ciphers/codes to protect the contents of a message even if discovered, remains portable, and is backward/forward compatible.

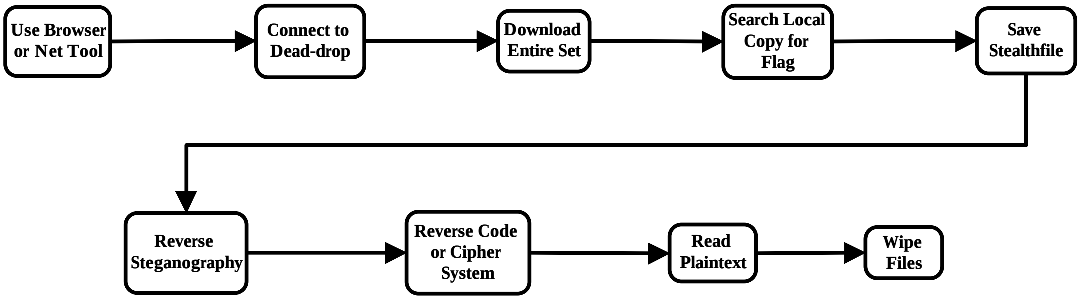
The drawback: real-time communication is not possible, and needs to be given up as part of the trade-off for security. This means ‘positive control’ is difficult to sustain, and so either asynchronous communication or ‘negative control’ must be acceptable.

This is a high-level process flow:

Sender-side of the Transaction



Receiver-side of the Transaction



LOCATION

The first choice that needs to be made is the location of the computer to be used. Using one’s own computer has certain advantages—control of the physical environment, control of the computing environment, easy access, selection of tools and applications, etc. The difficulty is that this makes the system a 'sitting duck'—perhaps a static IP address, a narrow range of connection options, the vulnerability and danger if the system is seized, stolen, or lost. The rule of intelligence, "don't break tradecraft," is even more important to abide by than normal in this situation.

If possible, using a 'throw-away' machine—a public library, an Internet café, a copy/print store that rents computer time, etc.—provides a certain level of anonymity. Such locations generally have a high-speed network connection, a large amount of traffic, a good range of applications, etc. The drawbacks are being limited to what's available on the machine, what can be supplied on disk, or what can be downloaded from the Internet. This is an issue because of the necessity for uncommon tools (cryptography, steganography), perhaps the need for a cipher key (in asymmetric systems like public key cryptography), etc. The 'throw-away' machine either need a disk drive (common in such systems), or the operator can rely on remote systems for support—downloading software, connecting to a special website (Java-based cryptography and steganography packages), connecting to a Unix shell account that has the necessary tools, etc.

From what is known of Al-Qaida, both approaches are used: the 11September2001 operators used public machines in a variety of locations; Indian, Kashmiri, Pakistani, and Afghan operators appear to have used their own computers, probably because of the lack of 'public' machines. Such machines should eventually provide clues, not only to specific operations, but to the operational tradecraft used by different elements of the Al-Qaida networks—allowing such tradecraft to be exploited against them.

CREATING THE PLAINTEXT

Plaintext is the 'clear' message that is to be exchanged between communicating parties. Computers provide a number of possible applications in which to create the plaintext, but the best application is one that creates the most 'raw' text possible. This reduces the size of the message dramatically, and the lack of formatting means less of a possible weakness to cryptanalysis (software packages that are word processors have well-known, well-defined file structures—structures that could be looked for when trying to 'break' the cipher/code that 'wraps' the plaintext). If it is necessary to 'save' the file while working on it, non-descript filenames are essential. Non-textual data is more complex—generally larger in size, structured, etc.—and may require special care, including being broken up into smaller files, and more than one layer of encryption.

USE CODE OR CIPHER-SYSTEM

Codes and ciphers have previously been discussed. Technology has advanced to the point where robust, automated coding systems can be implemented that rival cipher-systems. For the time being, ciphers have the upper hand.

Public-key systems (like PGP) should be open source and well-reviewed for vulnerabilities and flaws; cipher-systems that rely upon secrecy of the process, what is known as 'security through obscurity,' may harbor any number of unknown, and unresolved, problems. Public key systems should use 'large' keys—the number of bits, while slowing the encryption/decryption process down and being 'bulkier,' directly relates to the security afforded by the cipher-system. Sometimes bigger is indeed better.

Private-key systems (like the Data Encryption Standard, or DES) should use keys that are: as long as possible; are non-English; would not appear in a dictionary; and substitute symbols for letters or include symbols somewhere in the word or phrase. Good 'password hygiene' is a topic that a great deal of material is freely available on if you look on the Internet.

'Stream' systems also present some options. For example, a transcendental number such as pi, while deterministic (defined as "permitting at most one next move at any step in a computation," in other words, static, since the digits of pi are not random) can provide complex password schemes. A simple example would be sending a 'spam' message to one's partner in communication (along with a couple of thousand other people, just for camouflage), where the message didn't matter at all, but the **date** did. One could search the digits of pi (see http://www.angio.net/pi/piquery) for the date, and use the digits around the date, or at some set displacement, as the key for a file at a pre-arranged location. For example, searching for 11September2001, or 9112001, gives back: 25521170483883885632 **9112001** 16313965423993509775; or one can go in 9112001 digits for 68932968823686390526 20849; the possibilities are limited only by the imagination.

Application of a robust coding or cryptographic system to the plaintext file creates a 'ciphertext' file, which should look like noise or have no 'mapable' meaning discernable. If there are headers or footers (such as in the PGP example shown previously, marking the start and finish of the block), these need to be stripped off. Again, a non-descript filename is essential if the file is saved to disk.

'BASELINE' FILES

To use the steganographic application to 'hide' the ciphertext file, another file is necessary, the one that the ciphertext will be hidden within. Some discussions of steganography refer to this sort of file as the 'carrier' file, and the resulting file as the 'stego-carrier.' In the interest of not overloading the term 'carrier' (which is already in use in computer terminology), the 'clean' file herein is referred to as a 'baseline' file, and the combined ciphertext-baseline file is referred to as a 'stealthfile.'

Selection of a baseline file is critical in the tradecraft—it needs to fit the pretext of the exchange. To use a dead-drop, the baseline file should fit in with the other message traffic. Image files, JPG format in particular, provide the greatest range of possible pretexts (family photos, astronomical images, something that blends into the alt.* USENET hierarchy, etc.). Other possible file types are possible—text files (the 'stego' package can encode into text document whitespace, including a rather clever single- or double-space after periods), MP3s (which have the advantage of submerging into the peer-to-peer file exchange 'underground'), etc.

Baseline files need to be large enough (filesize) to 'absorb' the ciphertext encoded in without seeming suspicious or altering the real 'signal' of the baseline file (impacting on the image of a JPG, noise in an MP3, etc.). Large ciphertext files can be split up and spread across a number of baseline files if necessary—a number of JPG images form series that can be exploited for such purposes.

While blending in is critical, sticking with the overall theme and trends of the dead-drop, some signature or indicator to identify the right file or message is still important. This is part of the dilemma of the tradecraft—hiding in high-volume newsgroups, chat areas, etc. provides camouflage, but also increases the possibility that the communication may well be missed by the other party or parties involved.

APPLYING STEGANOGRAPHY

There are numerous applications that will 'stego' the ciphertext into the baseline/carrier file; rather than 'date' this document by providing links or specific applications, the interested reader should use their favorite search engine and use 'steganography' as their search term (which, besides software, will turn up any number of excellent papers on the subject, the history, the technology, and 'steganalysis'). Regardless of the selection of baseline file, a number of stego applications are freely or cheaply available at any point in time to satisfy a user. Caution must be taken, however, so a serious user should keep current on steganalysis; many packages leave indications that the baseline file has been altered, defeating the purpose of the tradecraft entirely. At a minimum, the tradecraft practitioner should compare the output of the stego software, the 'stealthfile,' with the original baseline file. If casual inspection reveals a difference, a different stego application or baseline fall are called for; the serious professional relying on the tradecraft for operational purposes should perform his/her own technical steganalysis to ensure security.

USING THE DEAD-DROP

The best location for a dead-drop is one that is high volume, high throughput, and trafficked enough to provide both anonymity and a plausible reason for being there. Physical dead-drops have varied greatly—public libraries, public parks, shopping malls, newsstands, movie theaters, etc. Casual interaction is what the Internet is all about, whether interactive in realtime (chat rooms, instant messages, etc.) or asynchronous (websites, email, USENET newsgroups, etc.), which make much of the Internet a perfect dead-drop.

Personal preference of the author is the USENET newsgroup—there are many ways to anonymously post to a newsgroup, relays, anonymizers, etc. that it provides the greatest flexibility, and also explicitly 'cuts' any connection between sender and receiver. The alt.* hierarchy of newsgroups provide immense variety, any number of plausible interests, and such volume of traffic that stealth communication is a needle in the haystack.

The World Wide Web is second-best as a dead-drop because of the more tangible connection between communicating parties; the more obscure the website (even 'free' homepages), the easier interaction through it is to track.

Instant messages (IMs) and electronic mail are also possibilities, but a connection can be even more firmly established than web-based interaction. IMs or chat rooms are less tenuous, and establish a level of interaction between communicating parties (violating the desire to avoid traffic analysis). Email provides some flexibility (throw-away accounts, like a cut-out; fake unsolicited advertising ('spam')), perhaps enough to meet trade-offs—more possibility of a connection being established by observation, in exchange for more immediacy in communication.

Hybrids are clearly worthwhile—unless newsgroups are constantly monitored, a recipient may miss a message. Some sort of spam email could provide the equivalent of a 'mailbox flag' (another piece of physical tradecraft—leaving an indicator somewhere that communication is desired or a dead-drop active), or as mentioned early, communicated a way to generate the key. Peer-to-peer (P2P) filesharing systems provide similar channels—for instance, looking for an MP3 of an obscure artist, anytime the MP3 becomes available on a system, there's something new stego'ed into it. P2P may 'connect' the two parties in an obvious way, but again perhaps with enough protection for the comfort of the parties.

What's important to stress, going to back to the introductory comments of this document, is that just about any element of the Internet can be turned into an element of communication tradecraft. While some of the available tools make it easier—plug-and-play—by no means are they the only mechanism to accomplish or approach secure communications.

CLEANING UP

The tradecraft process leaves a variety of traces on the machine that need to be taken care of, if limiting the forensics is a concern. The applications used (word processing, etc.) leave cache files; any work saved to disk leaves evidence behind even if 'deleted'; web browsers cache, acquire 'cookies,' etc. Finding all files created or altered during the session is an essential element, since they need to be subjected to a 'disk wipe' tool (again, a number of shareware or free applications are available), using 'military specification' erase (6-12 times overwrite, with patterns intended to make forensic reconstruction 'impossible').

REVERSING THE PROCESS

As can be seen in the process diagram at the start of this section, most of the tradecraft elements are symmetric—you just reverse the process. Some care does need to be taken, however, depending on the trade-offs and dead-drops selected. For example, it would be a break in tradecraft to only download one message from a USENET newsgroup; instead, the party should download an entire newsgroup (as well as other, ‘cover’ newsgroups) and search through the local copy. Not doing so would create a trail of unusual behavior and betray the specifics of the dead-drop (location, the identifier, etc.).

WEAKNESSES AND POINTS OF ATTACK ON THE TRADECRAFT

Getting 'sloppy,' hedging, or making bad trade-offs can introduce weaknesses or points of vulnerability to attack into the tradecraft:

- Traces can easily be left in the machine—files, keys, plaintext, cookies, the applications used for coding/ciphering and the steganography, etc. Forensics are getting better continually, and diligence in tradecraft is commonly lacking. Access to the necessary applications may be limited by the choice of location, requiring download, disk access, or connection to a remote shell (another set of security issues)

- The plaintext could be written in the system by the user or application, stored in a cache, be in a known and structured file format, be too large because of the filetype, etc.

- The cipher-system may have been poorly selected and have cryptanalytical weaknesses, too small a key chosen, poor selection of private key(s), inadequate entropy, various technical breakthroughs may occur that impact on the cipher-system's viability (quantum computing, application of Riemann's extension of Euler's zeta function, etc. undermining public key cryptography)

- Bad selection of the baseline file—implausible in the context of the dead-drop, inadequate size to hide the ciphertext, a legacy filetype, easily comparable baseline and stealthfile (exposing the tradecraft because of filesize differences, different data, etc.)

- Steganographic systems have weaknesses—you get what you pay for, and many packages are free because they are experimental. Signatures can be developed for stego approaches that can be scanned for. The easy availability of stego software means the applications can be reverse engineered or tested with a variety of plaintexts to find vulnerabilities

- Stealthfiles may have detectable alterations, may not fit in the pretext of the dead-drop, may have had a bad trade-off in the ratio between the ciphertext and the baseline original

- Web browsers cache, leave cookies, leave a trail (such as an IP address) unless measures are taken (relays, an anonymizer), and suffer numerous security failures

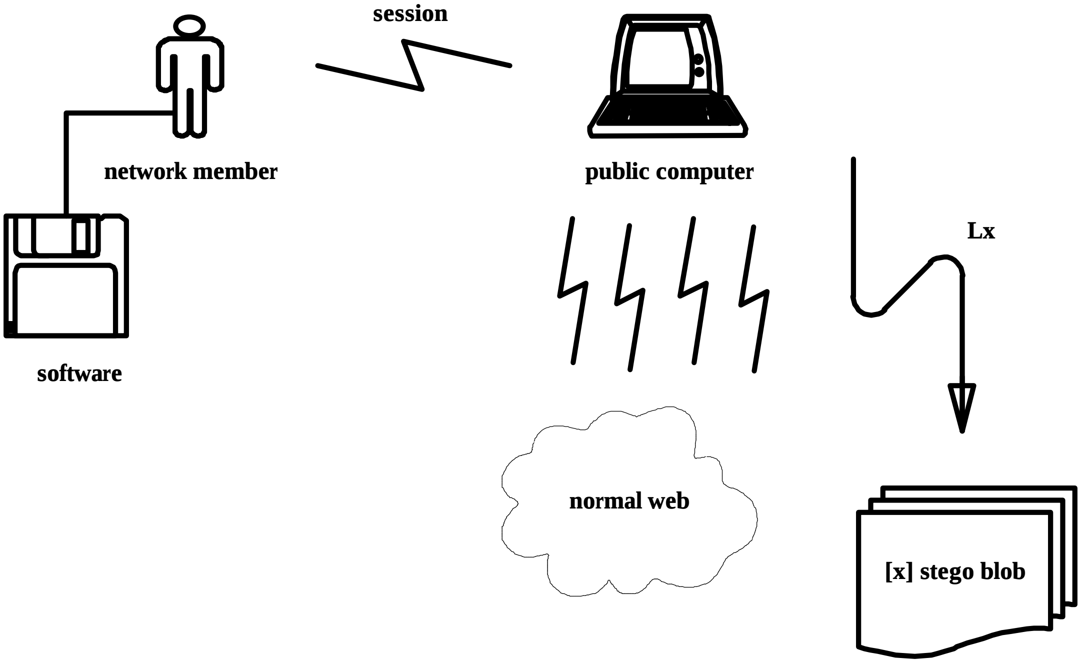
- Dead-drops may have inadequate 'cover' traffic or activity, or too high a 'churn' (message turnover). Many of the best USENET newsgroups are limited by ISPs because of the nature of their content. The web is a static target—logs, traceroutes, sniffers, cookies, Java exploits, etc. Instant messages and email establish a proximity between the sender and receiver

When followed appropriately, the tradecraft works, and works well. Even the slightest hedging off the tradecraft leads to weaknesses and vulnerabilities. Tradecraft is often taught as a 'rote' process—something to be followed dogmatically, without understanding. If you don't understand the technology, you'll get sloppy, because you won't understand all the ways you can be attacked.

Most important of all—even with the tradecraft, the approach can be weakened by a determined adversary, as long as they are willing to bend or break a few rules.

TURNING THE TRADECRAFT AGAINST THE USERS

This is a diagram of a possible interaction involved in using the tradecraft:



A member of Al-Qaida (A) connects to the Internet via a public (P1) or private (P2) computer (C) during a session (S). During S, the normal web is accessed, as well as a link (Lx) to a page that contains a 'blob' of data within which another file is inserted using steganography ([x]). Network member A has either brought a disk with the un-stego and deciphering software (to look at x while at the public computer), he/she will save [x] to disk for later un-stego and deciphering, or C is P2, a private computer which will have the software installed already.

Various cryptographic systems, as well as steganographic systems, leave 'signatures' or format indications as to the fact that something is ciphered, or that an arbitrary blob of data contains stego'ed data. This is the vulnerability—network members, by using this technology, are 'self-identifying' as someone with something to hide. Given the area of operations of Al-Qaida, cryptography and steganography is not commonly in usage—it is highly probable that someone engaging in the tradecraft is either a network member, or interesting and worth looking at anyway.

How is this useful? Knowledge of the tradecraft allows us to use it in 'turning the tables'—assets of the network are now actually identifiable, as opposed to nameless. Capitalizing on the vulnerability now requires the application of technology and some potential 'low risk' operations.

The initial weak-point is the computer system (C) used to access files that contain stego'ed data ([x]). If this is P1, a public system, there are two potential scenarios:

- Network member A has brought software with him/her to un-stego and decipher [x]; this means that A will be entering the password (or provide access to the private key using his/her personal password) on the local, network-accessible machine. This key (K_x) can 'sniffed' from the session S and transmitted out to the network (not in the clear, and not obviously—a specialized computer virus can accomplish this task)

- Network member A has brought a disk with him/her onto which will be saved the data blob that contains [x]. This disk will eventually need to be inserted into another computer (P2) that will be used to un-stego and decipher [x]; P2 may or may not be connected to the Internet. The virus resident on P1 will be transmitted to P2 with the disk transfer (a flaw in anti-viral software packages is that they look for known viral signatures and relatively unsophisticated viral attacks; such a virus will not be detected because it will not have a known signature, and will be very sophisticated in its approach), and will then take one of two approaches based upon the connection-status of P2. If P2 is Internet-connected, K_x will be transmitted out, just as if on P1. If P2 is not Internet-connected, it will be necessary to 'jump the airgap' a second time; the virus on P2 will write K_x back out to the disk (in hidden form)—available for a virus on P1 to access the next time the disk is available to it

If network member A is already using P2, a private machine, they will need to be identified at a later step in the process and subverted at that point (more below).

Getting the special virus into computer C initially, or gaining the benefit, will occur through one of three ways:

- Remote installation—public networks in the area of operations for Al-Qaida assets will be mapped and penetrated. This will mostly involve businesses, educational institutions, and public facilities like libraries or 'cyber-cafés.' Penetration of these sorts of networks or computers is generally trivial (and a task that can be automated), but some of them may have good security or system administration, which will require a 'local install'

- Local install—some computers may require physical access from the mission team in order to disable the local security controls. As these systems are public, a simple 'pretext' cover should be sufficient—an operations team member posing as a student, or for a 'cyber-café,' a back-packing tourist. The virus can be inserted into the machine through a disk, or by the team member accessing a normal-looking webpage that would handle the job with the member's assistance

- 'Dependency' install—some computers may be 'hard' targets, and these will be handled by either mapping all input/output network traffic and installing software traffic sniffers along all the possible I/O paths (Internet network architecture being what it is, any particular system has traffic 'choke points' that can be subverted), or by introduction of a 'trojan' attack on the system through local-network proxies that accomplishes the penetration. Systems that are private, such as P2, can be located in reverse using this mechanism as well—sniffing all traffic connecting to sites making stego-containing data blobs available, and tracking the path back to systems that request the data blob

The virus that accomplishes much of the technical 'dirty work' is reasonably simple in function:

- Sniff the session (S) and maintain a log of critical interaction elements (email addresses, keys, links, telnet connections and login/password, etc.); sessions are defined by periods of interaction punctuated by clear demarcations (gaps in time, power cycling, etc.)

- Examine incoming data blobs for cryptographic signatures or for evidence of interaction with stego'ed data blobs; this requires a small database of cipher and stego signatures, which can be lightly encoded to avoid detection

- Upon interaction with a cryptosystem, ciphered data, steganographic system, or stego'ed data blob, notify a remote system of the interaction. The most bare notification packet is the IP address, Lx, and K_x. Notification packets can be hidden as cookie interactions, stealth cookies, stego'ed into other valid network packets, transmitted using a covert channel semaphor, or any number of very light communication mechanisms. Such packets can either be managed in real-time (which can be referenced against an IP map of penetrated sites, to identify immediately where the network asset physically is, in the event a direct action or special reconnaissance operation is worthwhile), or buried through dead-drops (posted to a newsgroup, dropped into a remailer spool, posted in a stego'ed form to a fake webpage, etc.)

- The log of network member A's session (S) can be sent from the computer based on various trade-offs (public or private, usage, urgency, etc.), again using secure mechanisms to move the data to an anonymous and untraceable back-end

- At reasonable intervals, the virus will 'check in' by sending the IP address of computer C using the secure channel. These contact 'pings' are an acknowledgement and also allow a detailed map of penetrated IPs. Lack of contact means that the situation for that IP address needs to be checked

Gathering the data sent by the network of viruses could be accomplished by a spidering collection system, set to look in the various places available as dead-drops for the vital details. This would develop into detailed intelligence models of contact networks, nodes used for access, sites used to get C4I instructions from, the keys necessary to un-stego/decipher message traffic, and the content of the message traffic.

A comprehensive spider system, one that crawls Internet websites looking for stego-containing data blobs, can also work the system from the other side. Once Lx is identified, the site or the I/O network surrounding it can be penetrated or sniffed to track access to stego'ed data blobs. This will provide the IP address of P2, which can then be penetrated or sniffed to further extend the maps and models of Al-Qaida's network.

Over time, the system would develop an extensive detail on the keyspace used by Al-Qaida; this will provide access to a great deal of their ciphered and stego'ed materials, even without direct access to some network members (materials discovered by the spider system may be immediately decipherable, if the key has already be added to the known keyspace database).

Given the use of chat and interactive forums, sniffing Instant Message servers for signatures also makes sense. Servers for swapping MP3s and other rich media types may also be priorities to spider and seek signatures from. Chat rooms might require the use of software agents (simple artificial intelligence systems) that can scan chat room interaction for exchange of stego'ed data.

This volume of data would provide direct insight (and hopefully advanced warning) about Al-Qaida operations. Traffic analysis and content analysis will lead to an understanding of the function and process of the network, particularly identification of which individuals provide the necessary support functions. These personnel are the individuals to target in order to degrade or destroy the functionality of the Al-Qaida network.

The necessary next steps to proceed against Al-Qaida in such a fashion would be:

- Development of a detailed operations plan

- Development of a detailed technology architecture to accomplish the tasks
- Creation of a current, comprehensive cipher and stego signature database
- Implementation of the software packages necessary (virus, spider, penetration automation, analysis support system)
- Creation of a database of potential IPs in the Al-Qaida network area of operations
- Insertion or penetration of computer systems with virus attack
- Gathering, analysis, and direct action operations

What is most striking about this is how similar it appears to be to what has emerged from the U.S. Federal Bureau of Investigation (FBI) about their ‘Magic Lantern’ project. According the media reports, the FBI’s technology has grown along lines similar to what has been presented here, in reaction to increasing use of cryptographic technology (and, perhaps, steganography).

In the context of the ‘war on terror,’ this may not seem so terrible a thing, but what must be remembered is that the FBI initiated these projects well in advance of the events of 11September2001. History will judge whether such actions were prophetic, or motivated by less-than-noble aspirations. Even the name of the technology, Magic Lantern (certainly better than ‘Carnivore,’ the previous sniffing technology), is worth considering. Is it a way to cope with encryption, something long referred to as being the genie ‘out of the bottle’ or should that be ‘out of the lantern.’ Or should images of comic book heroes be conjured up:

> “In brightest day, in blackest night,
> no evil shall escape my sight!
> Let those who worship evil’s might,
> beware my power.. Green Lantern’s light!”
> --The ‘Green Lantern’ Oath, Copyright DC Comics

Cypherpunks will continue to create tools to protect privacy, and perhaps the ‘war on terror’ makes comic book heroes (or at least aspirations) of the FBI.

CONCLUSIONS

It is dangerous to be right when the government is wrong.

—Voltaire

This document is a companion piece to two other documents:

Hunting the Sleepers, which can be found at:

http://www.metatempo.com/huntingthesleepers.pdf

An Analysis of Al-Qaida Tradecraft, which can be found at:

http://www.metatempo.com/analysis-alqaida-tradecraft.html

Al-Qaida worries about its communications security, and will clearly use whatever technology benefits them—the use of codes is encouraged, and the network clearly wanted technology for secure communications. The U.S. clearly takes the possibility of Al-Qaida utilization of codes and ciphers seriously enough that it requested review and limited presentation of bin Laden communications and other Al-Qaida media releases.

The U.S. sleepers, the 11September2001 operators, hedged off the tradecraft, but did use public systems, cut-out email providers, etc. It is unknown at this time how well they managed to cover their tracks. If nothing else, law enforcement has extraordinary powers, vast resources to draw upon, and manpower to follow up with any possible lead.

India and elsewhere appear to have encountered suspicious materials on seized computer systems, including some indications that the tradecraft discussed herein is used by covert operatives, at least in such areas of operation.

Regardless of the motivation of the FBI in initiating projects such as Magic Lantern, efforts at understanding and exploiting vulnerabilities in Al-Qaida tradecraft are going to be undertaken.

As presented, such tradecraft can provide security, but could also be used to literally self-identify Al-Qaida network members. Seized computers and captured/arrested Al-Qaida members may shed further light on this over time. If they are indeed using the tradecraft for secure communications, then even this can be turned on them, continuing the hunt for the sleepers.