Steganography
-- hiding in plain sight

David Morgan

What’s steganography?

“Covered writing”

steganos = covered*  graphy = writing

* Antonis Christodoulou, an excellent Spring 2008 CS530 student from Greece, challenged this translation. He said there is more to the meaning of the word in Greek than this. But “covered” is how all the English language technical literature presents it.
What’s steganography?

- steganography
  - embedding information (plaintext) within other seemingly harmless information (covertext) in such a way that no one but the intended recipient would *try* to retrieve it

- *versus* cryptography
  - transforming information (plaintext) into other unintelligible information (ciphertext) such that no one but the intended recipient would *be able* to retrieve it

Further differences

- Steganography
  - hide, without altering
  - obfuscates the fact of communication, not the data
  - preventative - deters attacks

- Cryptography
  - alter, without hiding
  - obfuscates the data, not the fact of communication
  - curative - defends attacks
Non-cyber examples

Targets are inherent, embedded, camouflaged, implicit in their environment. They blend in with the crowd.

animal camouflage

Where’s Waldo?

Other non-cyber techniques

- subset
- null cipher
- Bacon cipher
Dear George;

Greetings to all at Oxford. Many thanks for your letter and for the summer examination package. All Entry Forms and Fees Forms should be ready for final despatch to the Syndicate by Friday 20th or at the very latest, I’m told by the 21st. Admin has improved here, thought there’s room for improvement still, just give us all two or three more years and we’ll really show you! Please don’t let these wretched 16+ proposals destroy your basic O and A pattern. Certainly this sort of change, if implemented immediately, would bring chaos.

Sincerely yours;

Imagine a package is being prepared for you.

This tells you when and where you can get it:
Null cipher – 1st letter

PRESIDENT'S EMBARGO RULING SHOULD HAVE IMMEDIATE NOTICE. GRAVE SITUATION AFFECTING INTERNATIONAL LAW. STATEMENT FORESHADOWS RUIN OF MANY NEUTRALS. YELLOW JOURNALS UNIFYING NATIONAL EXCITEMENT IMMENSELY.

PERSHING SAILS FROM NY JUNE I

24-character message in 204-character covertext (8.50 ratio)

Different covertext, same plaintext

APPARENTLY NEUTRAL'S PROTEST IS THOROUGHLY DISCOUNTED AND IGNORED. IS MAN HARD HIT. BLOCKADE ISSUE AFFECTS PRETEXT FOR EMBARGO ON BYPRODUCTS, EJECTING SUETS AND VEGETABLE OILS.

PERSHING SAILS FROM NY JUNE I

24-character message in 176-character covertext (7.33 ratio)
**Bacon’s cipher**

Have fun

aabbb aaaaa baabb aabaa aabab baabb abbaa

Burger with fries tasty but not for health

uses a “bilateral” alphabet
each letter has 2 possible fonts (or cases)

---

**What’s this one?**

USc athletics is surpassed by computer science

HINT: starts with same letter as previous because BUrge == UScat
A less obvious bilateral alphabet

The Biliteral Alphabet
From Bacon’s De Augmentis Scientiarum

This plate is reproduced from Bacon’s De Augmentis Scientiarum, and shows the two alphabets as designed by him for the purpose of his cipher. Each capital and small letter has two distinct forms which are designated “a” and “b.” The biliteral system did not in every instance make use of two alphabets in which the differences were as perceptible as in the example here given, but two alphabets were always used; sometimes the variations are so minute that it requires a powerful magnifying glass to distinguish the difference between the “a” and the “b” types of letters.

from The Philosophical Research Society
at http://www.prs.org/gallery-bacon.htm

Doing it with computers

- Steganography – hiding a file inside of another
  - typically hiding text inside of a media file
  - normally used for the transportation of secretive information

- Operating System
  - unused memory
    - slack space
    - unallocated space
  - hidden partition
  - normally used to hide data from investigators

- Network
  - unused bits in packet headers
  - spread spectrum, frequency shifting
Photo as cover - any difference?

Least Significant Bit Manipulation

- Idea is that the least significant bit of a byte can change with little change to the overall file
- Consider a 8-bit grey scale image
  - One pixel of information is stored using 8 bits.
  - There are 256 different variations of grey.
LSB continued

- Change in the LSB information of some area of the image will not be noticeable by naked eye.
- Utilizing this fact the message is embedded

```
10101101 00101010 10100010 10010001 10...
```

```
10101100 00101011 10100011 10010000 10...
```

LSB advantages and disadvantages

- **Advantages**
  - Does not change the size of the file
  - Is harder to detect than other steganography techniques

- **Disadvantages**
  - Normally must use the original program to hide and reveal data
  - If the picture with the hidden information is converted to another format, then the hidden data may be lost
Some network examples

- **embedding** data directly
  - in header fields, and/or
  - in payload
- **expressing** data by network event timing
  - data is just patterns
  - can be non-material
  - e.g., morse code

IP packet header

<table>
<thead>
<tr>
<th>Version</th>
<th>IHL</th>
<th>Type of Service</th>
<th>Total Length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identification</td>
<td>Flags</td>
<td>Fragment Offset</td>
<td></td>
</tr>
<tr>
<td>Time to Live</td>
<td>Protocol</td>
<td>Header Checksum</td>
<td></td>
</tr>
<tr>
<td>Source Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Destination Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Options</td>
<td>Padding</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

32 bits

fields available for embedding steganographic data
TCP packet (segment) header

32 bits

<table>
<thead>
<tr>
<th>Field</th>
<th>4 bits</th>
<th>4 bits</th>
<th>16 bits</th>
<th>24 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source Port</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Destination Port</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sequence Number</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Acknowledgment Number</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Offset</td>
<td>4 bits</td>
<td>Reserved</td>
<td>20 bits</td>
<td>Window</td>
</tr>
<tr>
<td>Data</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Checksum</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Urgent Pointer</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

your data ... next 500 octets

Put ‘em where they don’t belong
because you can

*fields available for embedding steganographic passengers
The protocols don’t restrict

- IP “identification” field’s value
  - “An internet header field carrying the identifying value assigned by the sender to aid in assembling the fragments of a datagram.”
  - RFC 791, “Internet Protocol”

- TCP “sequence number” field’s value
  - “When new connections are created, an initial sequence number (ISN) generator is employed which selects a new 32 bit ISN. The generator is bound to a … clock … [but] not tied to a global clock in the network, and TCPs may have different mechanisms for picking the ISN’s.”
  - RFC 793, Transmission Control Protocol

Proof-of-concept covert channel demo

- Named “covert_tcp” by Craig Rowland
- client/sender and server/receiver roles
- client places data in either
  - IP header’s “identification” field, or
  - TCP header’s “sequence number” field
- server knows, fetches the data out

http://www.firstmonday.org/Issues/issue2_5/rowland/
Fields alternatively utilized

Simultaneous screenshots

client/sender (on 192.168.1.20)

```
[root@V1 root]# ./covert_tcp -server -dest 192.168.1.132 -source 192.168.1.20 -source_port 1234 -dest_port 80 -file covert_data_to_send
Covert TCP 1.0 (c)1996 Craig H. Rowland
(crowland@psionic.com)
Not for commercial use without permission.
Destination Host: 192.168.1.132
Source Host : 192.168.1.20
Originating Port: 1234
Destination Port: 80
Encoded Filename: covert_data_to_send
Encoding Type   : IP ID
Client Mode: Sending data.
Sending Data: A
Sending Data: B
Sending Data: C
```

file content: ABC

server/receiver (on 192.168.1.132)

```
[root@V1 root]# ./covert_tcp -dest 192.168.1.132 -source 192.168.1.20 -file captured_data.txt
Covert TCP 1.0 (c)1996 Craig H. Rowland
(crowland@psionic.com)
Not for commercial use without permission.
Listening for data from IP: 192.168.1.20
Listening for data bound for local port: Any Port
Decoded Filename: captured_data.txt
Decoding Type Is: IP packet ID
Server Mode: Listening for data.
Receiving Data: A
Receiving Data: B
Receiving Data: C
```
Packet dump seen at server
-- using IP identification field

Letter | Ascii code | \(\text{ASCII} \times 256\)
--- | --- | ---
A | 65 | 16640
B | 66 | 16896
C | 67 | 17152
D | 68 | etc etc

Simultaneous screenshots

client/sender (on 192.168.1.20)

server/receiver (on 192.168.1.132)
Packet dump seen at server
-- using TCP sequence number field

```bash
[root@localhost ~]# tcpdump -nA -r seq.server.cap | cat -n
reading from file seq.server.cap, link-type EN0, proto (Ethernet)
2  E: .......
3  IP 192.168.1.132.80 > 192.168.1.20.1284: R 0:0:0:0:0:0:0:0:0:0:0:0:0:0:0:0
4  E: .......
6  E: .......
7  IP 192.168.1.132.80 > 192.168.1.20.1284: R 0:0:0:0:0:0:0:0:0:0:0:0:0:0:0:0 Q: win 0
8  E: .......
10  E: .......
11  IP 192.168.1.132.80 > 192.168.1.20.1284: R 0:0:0:0:0:0:0:0:0:0:0:0:0:0:0:0 Q: win 0
12  E: .......
[root@localhost ~]#```

Voice stream as cover
- invent your own protocol

Table 1. Header fields and their function

<table>
<thead>
<tr>
<th>Type of field</th>
<th>No. of bits</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>P (Parameter)</td>
<td>4</td>
<td>Indicates parameter that is transmitted inside the watermark</td>
</tr>
<tr>
<td>S (Side)</td>
<td>1</td>
<td>Indicates the side of the communication (1 - sender, 0 - receiver)</td>
</tr>
<tr>
<td>C (Continuity)</td>
<td>1</td>
<td>Describes if a packet contains the beginning or continuation of the parameter indicated in the field P (1 - beginning of new parameter, 0 - continuation of the last parameter)</td>
</tr>
</tbody>
</table>

Exemplary values of the field P are shown below:
0001 – authentication or integrity parameter (32 bits)
0010 – informational parameter 1 (32 bits)
0011 – informational parameter 2 (32 bits)
0100 – informational parameter 3 (32 bits)
0101 – post-authentication and integrity parameter (32 bits)

"Covert channel for improving VoIP security"
Voice stream as cover
- embed your protocol in VoIP packets

```
“Covert channel for improving VoIP security”
```

Photo, voice work well because
… bad is good enough

- for human consumption
- our crude senses have high “error” tolerance
- right and wrong – mom never taught me the difference
  – (slightly) wrong colors look the same as right
  – (slightly) wrong voice sounds the same as right
Even more covert...

- signal timing
- port knocking

Even more covert...

- Embed nothing in the non-surreptitious channel
- Instead express covert info by timing channel’s non-covert exchanges

http://userpages.umbc.edu/~chauhan2/CMSC691I/Embedding_Covert_channels_into_TCP_IP.ppt
Port knocking...

- encode information in sequence of “open port” requests
  - the port sequence is the information e.g.
    - port 72, port 69, port 76, port 76, port 79 requests, signify “hello”
  - the port sequence is a password/comboination to “unlock” a response e.g.
    - port 10004, port 10030, port 10012, signify “turn on httpd on port 80”

- receiver observes, optionally reacts
  - daemon sniffs network
  - process watches firewall log

http://www.portknocking.org/

Products

- s-tools
- outguess
- various others