Covert Communication & Malicious Cryptography

Aggelos Kiayias

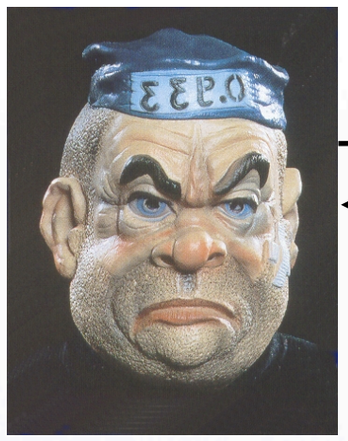
Covert Channel

- A covert channel is a communication channel that
- carries information over certain aspects of objects that are not normally viewed as "information carriers."

A motivating example

two prisoners

Alice



Bob

monitoring

Allowed communication: Poems

Intended communication: Escape plan

Simmons, Crypto 1983

Covert Storage

- Storing information within aspects of objects so that the stored information is inconspicuous.
- Similar to a covert channel but object is "static"

Legitimacy of Covert channels.

- Capability of transferring information from within a hostile environment to outside destination.
- Censorship protection.
- Protection of sensitive data from nosy observers.

Time covert channel

- Sender and receiver have synchronized clocks.
- The sender transmits a steady flow of messages (of irrelevant content).
- If the sender wants to send a "0" then it sends a message in a specific time-slice. Alternatively if the sender wants to send a "1" then it sends it in the corresponding slice.

Time is hard to keep

- Alternatively one may use some other type of redundancy in a communication channel.
- Caveat: one has to employ a carrier channel.

Carrier Channels

- Discovering redundancies in the TCP/IP communication.

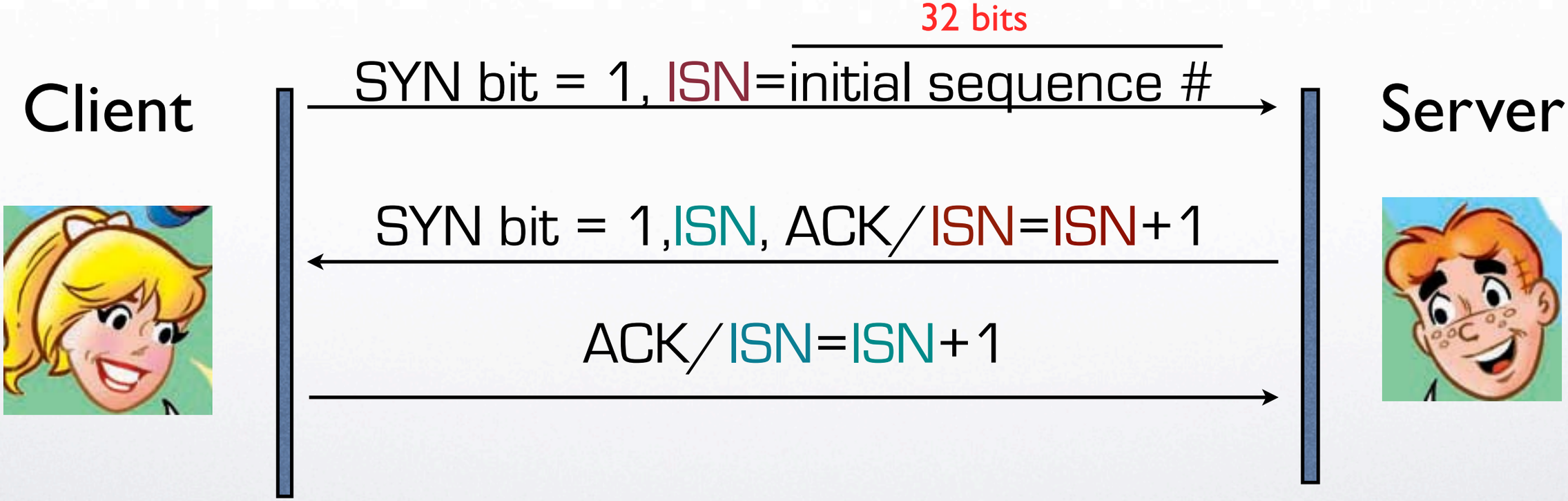
Using TCP/IP Headers

- Identification field within the IP header:
  - used for reassembly of packet data. Gives a unique id tag to fragmented data (16bit)
- Initial Sequence Number field within the TCP header. 32 bits of data.

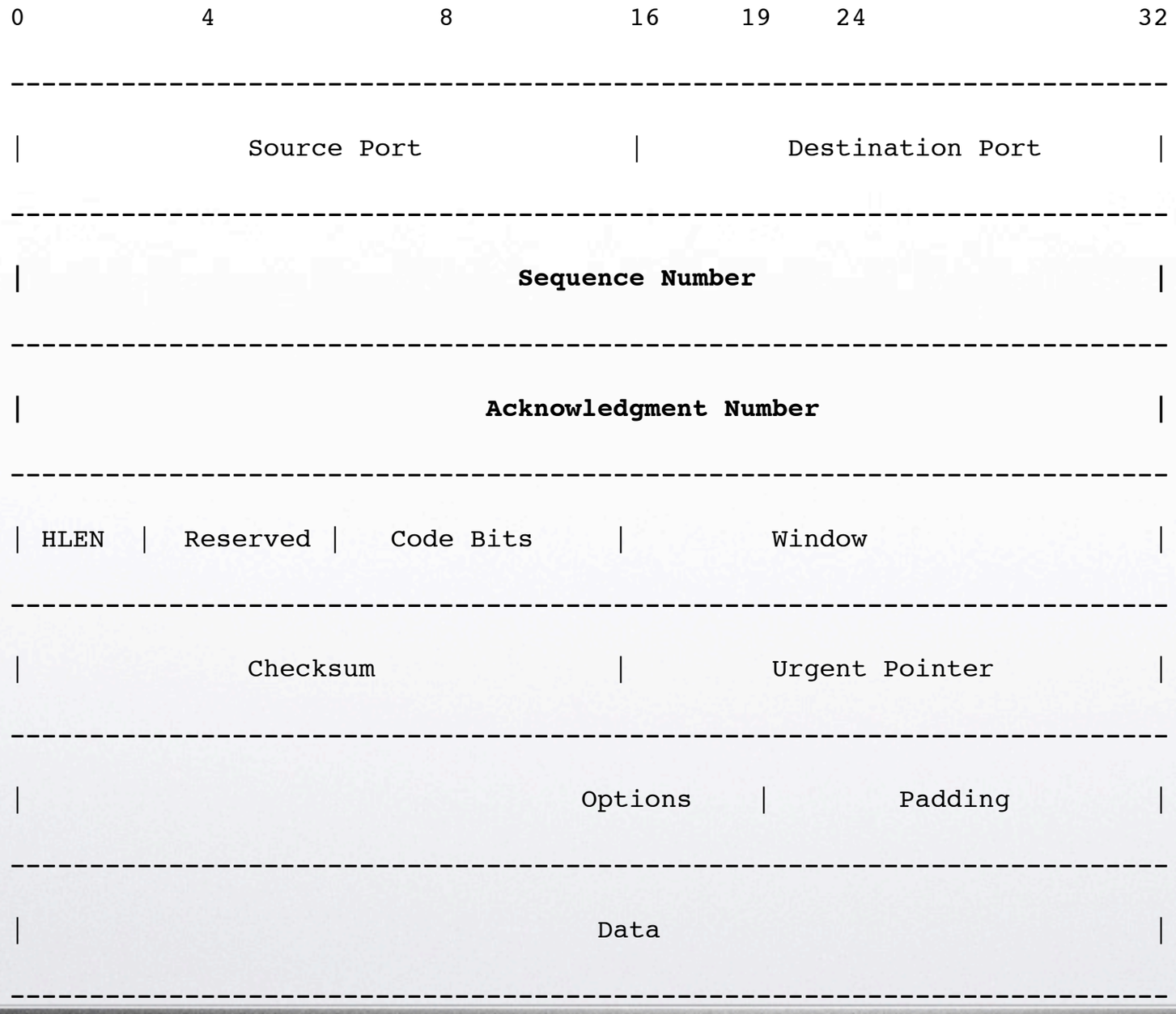
C. H. Rowland, Covert channels in the TCP/IP Protocol Suite, http://www.firstmonday.org/issues/issue2_5/rowland/

TCP/IP handshake

How acknowledgments work



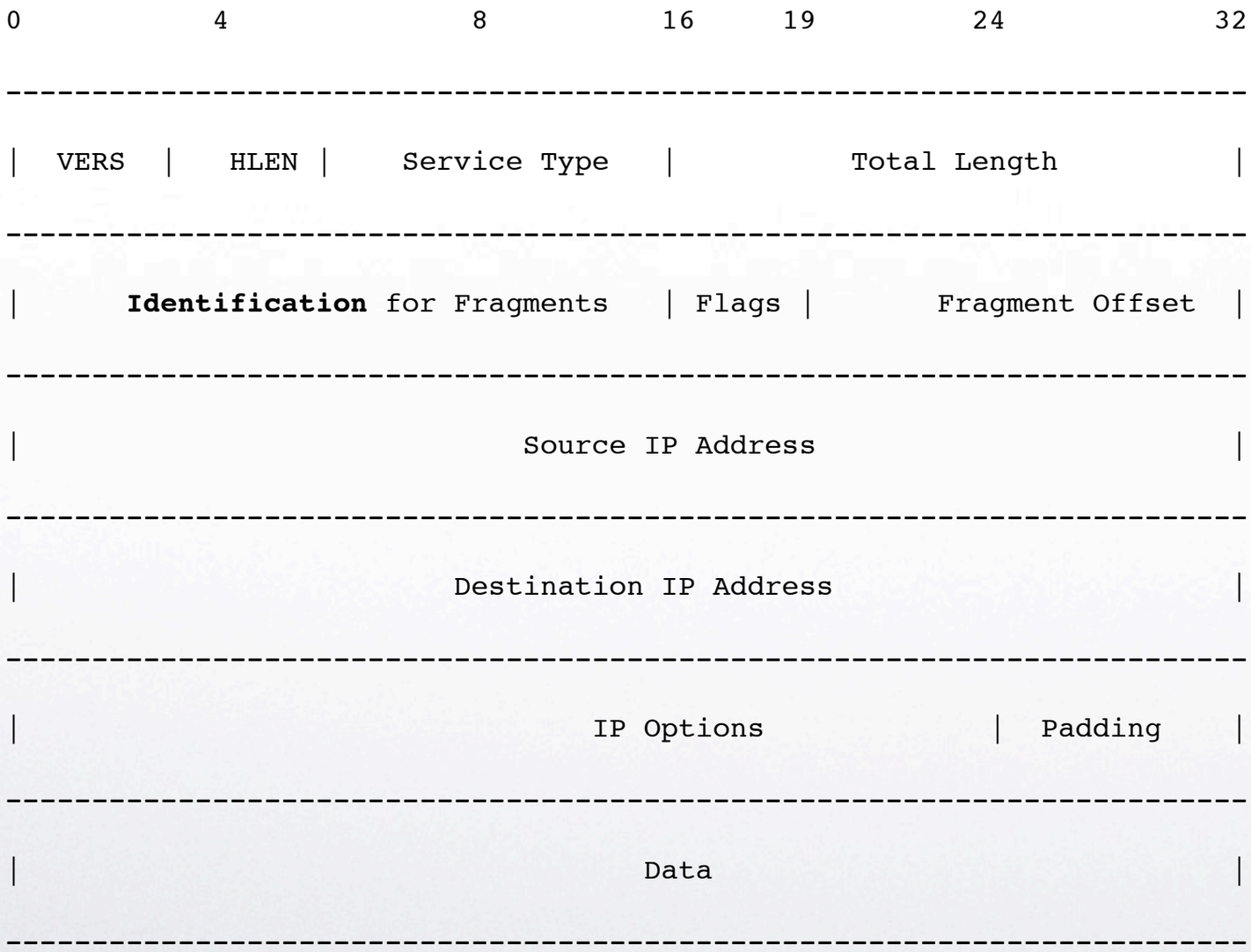
TCP Header



Sequence Number. 32 bits. The sequence number of the first data byte in this segment. If the SYN bit is set, the sequence number is the initial sequence number and the first data byte is initial sequence number + 1.

Acknowledgment Number. 32 bits.
If the ACK bit is set, this field contains the value of the next sequence number the sender of the segment is expecting to receive. Once a connection is established this is always sent.

IP Header



Identification. 16 bits.
Used to identify the fragments of one datagram from those of another. The originating protocol module of an internet datagram sets the identification field to a value that must be unique for that source-destination pair and protocol for the time the datagram will be active in the internet system. The originating protocol module of a complete datagram clears the *MF* bit to zero and the *Fragment Offset* field to zero

see http://www.networksorcery.com/enp/protocol/ip.htm for description

Bouncing packets

- In the TCP header
  - Set destination IP = a random server
  - Set source IP = your real destination
  - Set ISN = message - 1

TCP Timestamps

```
0        4          8         16     19   24              32
-------------------------------------------------------------
|              Source Port       |      Destination Port    |
-------------------------------------------------------------
|                      Sequence Number                      |
-------------------------------------------------------------
|                   Acknowledgment Number                   |
-------------------------------------------------------------
| HLEN  | Reserved | Code Bits   |          Window          |
-------------------------------------------------------------
|             Checksum           |      Urgent Pointer      |
-------------------------------------------------------------
|                        Options     |        Padding       |
-------------------------------------------------------------
|                           Data                            |
-------------------------------------------------------------
```

```
TCP Option 8:
Kind (8 bits) = 8
Length (8 bits) = 10
TSval (32 bits) = timestamp
    current value of the
    timestamp clock.
TSecr (32 bits) = time
    If ACK=1 then this an
    echo of a SYN timestamp
```

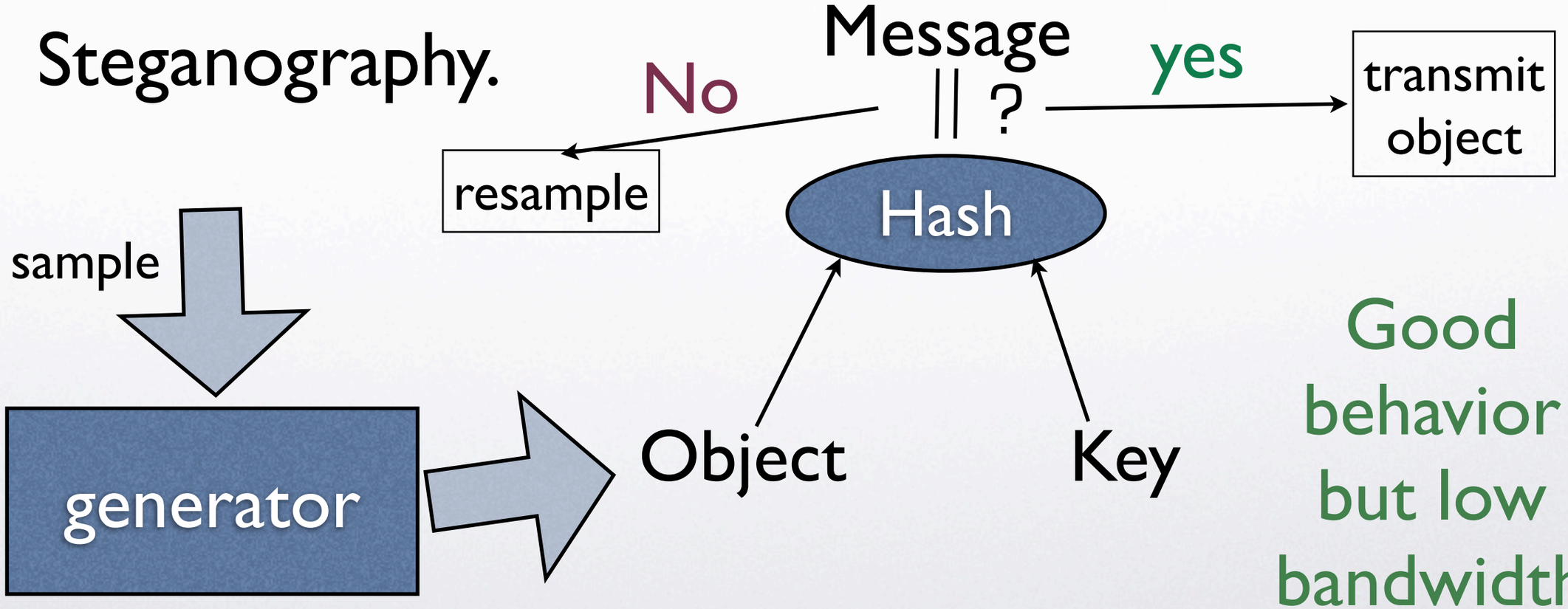
Using TCP Timestamps

Giffin, Greenstadt, Litwack, Tibbetts, PET 2002

- Use low order bits of TCP time stamp to hide encrypted message (a modified Kernel will delay packets accordingly)
- Use high order bits of TCP Time Stamp + Key to inform receiver which bit is transmitted: if T is the high order bits then Hash(T, K) = i would be the index of the bit transmitted in the low order bit.
- Repeat the above till all message is transmitted.
- Problem: low order TCP timestamps may not really be random (as in the ciphertext) - so channel bandwidth would have to be kept low.

Rejection Sampling

- General technique for embedding data in a covert fashion.
- Steganography.



Good behavior but low bandwidth

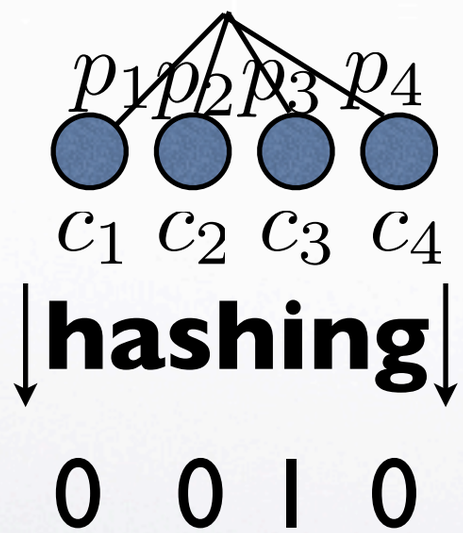
Rejection Biasing

- Pick a cover object. If it hashes to the bit to be transmitted send it. Else pick a second one and send it (no matter the hash)

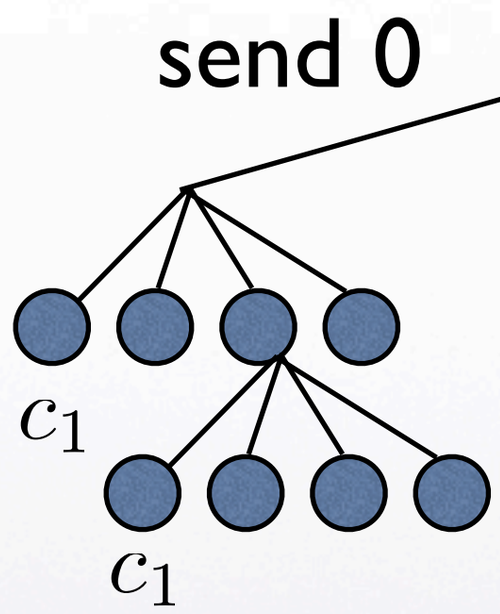
Rejection Biasing

$$\sum_{i=1}^{4} p_i = 1$$

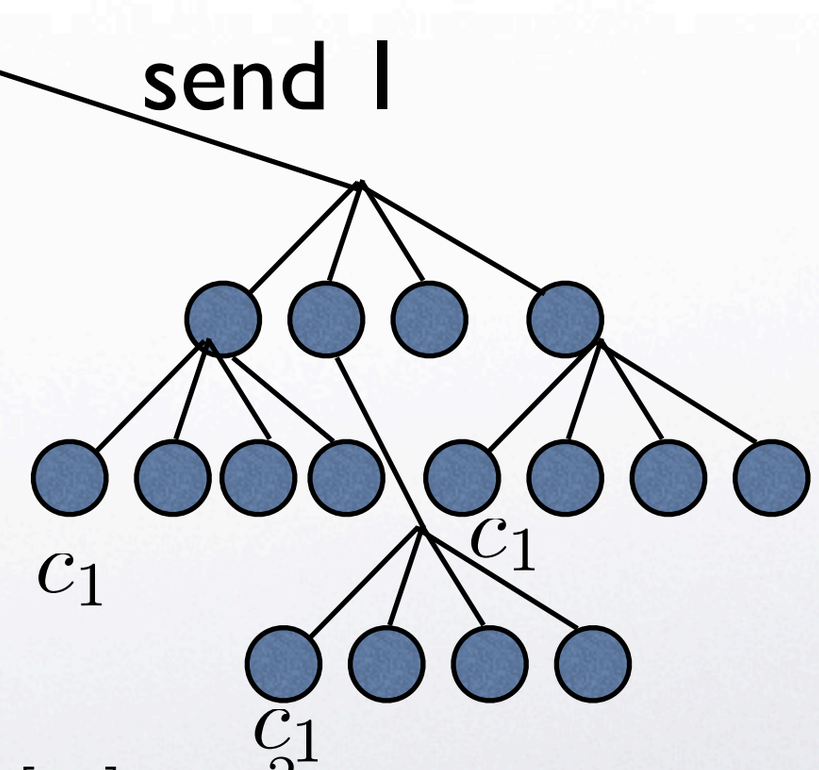
cover distribution



compare:



$$Pr[c_1] = p_1 + p_3 p_1$$



$$Pr[c_1] = p_1^2 + p_2 p_1 + p_4 p_1$$

Rejection Biasing Theorem

- If one transmits totally random data then rejection biasing is indistinguishable from the cover distribution!
- Trick to exploit that : use cryptographic techniques to make data you want to transmit indistinguishable from random

Malicious Cryptography

- "Using cryptography against Cryptography": introduced by A. Young, M. Yung.
- Observation #1: cryptographic data are ideal carriers of covert communication; such communication can be used to defeat encryption mechanisms.
- Observation #2: viruses equipped with a public-key can deliver surprising damage.

The Leaking Key Generator

- A company sells a cryptographic key generator.
- Do you trust that the data you encrypt under this generator is secure?

RSA Key generator

How to produce a key of length W

```
repeat for ever
    p = random bitstring of length W/2;
    if p>=2^{W/2} + 1 and p is prime then break;
repeat for ever
    q = random bitstring of length W/2;
    if q>=2^{W/2} + 1 and q is prime then break;
if length(pq)<W or p=q restart
if p>q then swap(p,q)
output(p,q)
```

Your public-key is n=pq

RSA Key Generator 2

```
s = random bitstring of length T;
repeat for ever
    p = PRNG(W/2, s); /* stretch s to length W/2 */
    if p>=2^{W/2} + 1 and p is prime then break;
repeat for ever
    q = random bitstring of length W/2;
    if q>=2^{W/2} + 1 and q is prime then break;
if length(pq)<W or p=q or truncate(T,pq)!=s restart
if p>q then swap(p,q)
output(p,q)
```

```
truncate(T, X) = T least
significant bits of X
```

What is happening here?

Crypto vs. Crypto

- We just used a public-key as a covert channel.
- and we used it to transmit its corresponding secret-key.
- What does this mean for commercial cryptographic libraries?
- What does it mean for Open-Source?

The Cryptovirus

- A virus that is equipped with a public-key and an encryption algorithm.

- A cryptovirus can be used for a *Cryptoviral Extortion Attack*.
- Also for stealing information subliminally.

see http://www.cryptovirology.com/ maintained by A. Young

Cryptoviral Extortion

- **Phase I**: Virus is dormant and replicates. It carries a public-key. It also contains an encryption function and a random number generator.

- **Phase II**: Virus is activated; it generates a one-time key K and uses a symmetric cipher to encrypt the contents of the victim's hard-disk. Then it uses the public-key to encrypt K that results in a ciphertext C.

- **Phase III**: Virus saves C and pops up an alert box to inform victim. It provides instructions on how to communicate with Virus writer + ransom info. Virus Terminates.

Cryptoviral Extortion II

wishful thinking phase: Victim contacts IT Department.

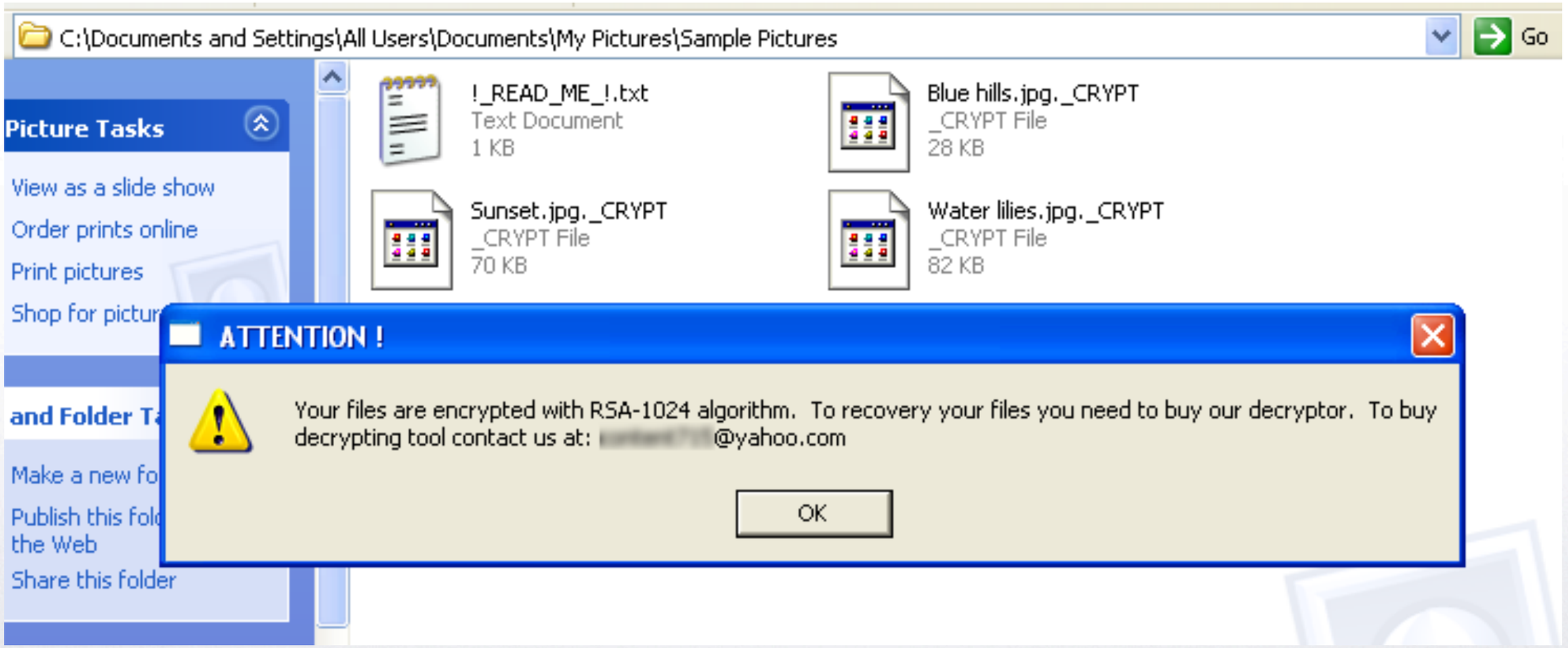
IT department experts analyze virus code and attempt to recover the data....Will this work?

Phase IV:Victim contacts virus writer, gives the ciphertext C and pays ransom. It receives the key K to recover the hard-disk data.

In practice

- GPcode trojan.
  - encrypts files .doc, .html, .jpg, .zip
  - First versions flawed - current ones uses RSA 1024 bits for public-key and 128-bit RC4 for symmetric encryption.

The Extortion



screenshot from http://people.csail.mit.edu/tromer/gpcode/

Subliminal Information Stealing

- Another type of cryptoviruses:
  - using private information retrieval techniques to scan database for information they are interested in.
  - using subliminal channels they leak the information to attacker.