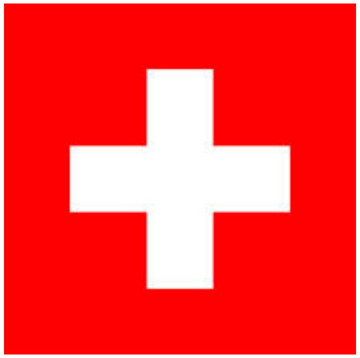
Secret Sharing, Random Numbers, and Information Hiding

Prof. Tom Austin
San José State University
Spring 2014

Summer University 2014



- Summer CS program held in Yverdon-les-Bains, Switzerland.
- Applications are due April 1.
- More details at http://www.cs.sjsu.edu/su/su14/index.htm

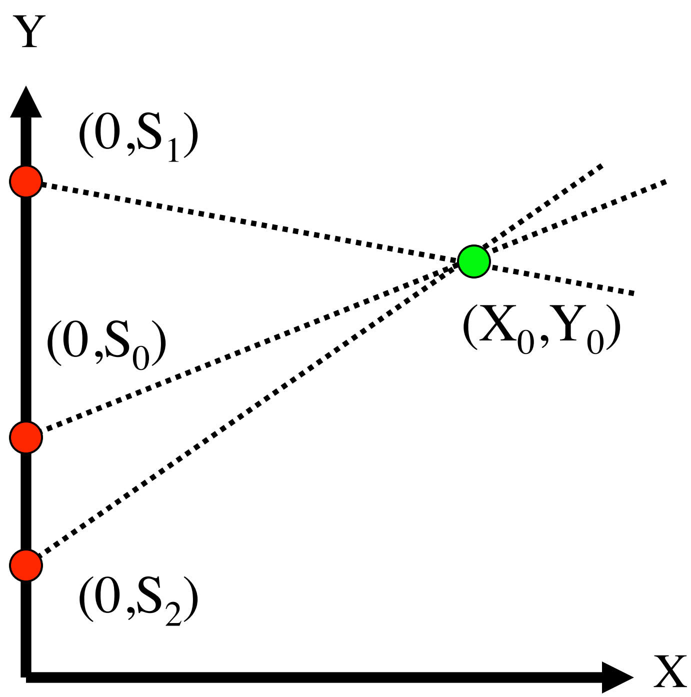
Secret Sharing: Motivation

- Goal: make secret available, but make it hard to peek.
- Divide secret among multiple organizations.
- Separately, the pieces of give no information about the secret.

Suppose you want to share a secret number S between Alice and Bob.

How can you divide it between them?

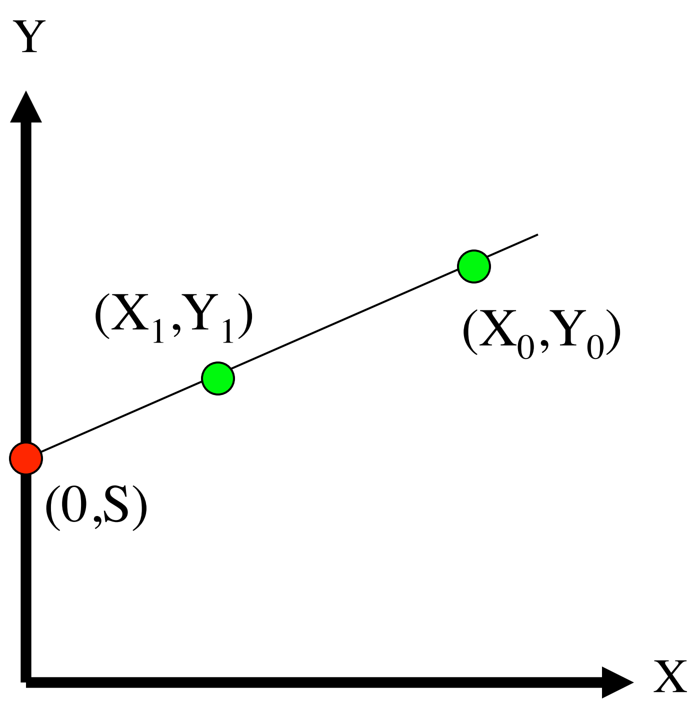
Shamir's Secret Sharing



2 out of 2

- Two points determine a line
- Give (X_0,Y_0) to Alice
- Give (X_1,Y_1) to Bob
- The secret is (0,S), i.e. where the line crosses the y axis.

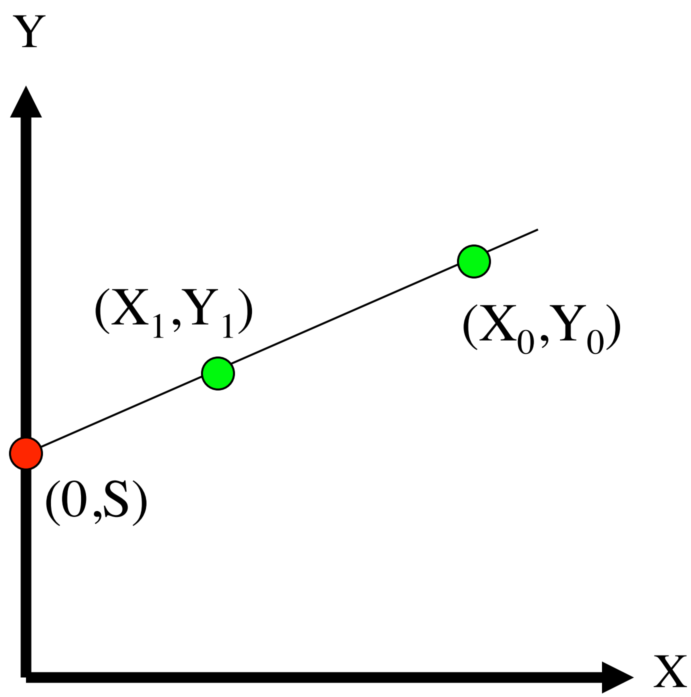
Shamir's Secret Sharing



2 out of 2

- If Alice and Bob cooperate, they can find the secret S
- Also works in discrete case
- Easy to make "m out of n" scheme for any $m \leq n$

Shamir's Secret Sharing

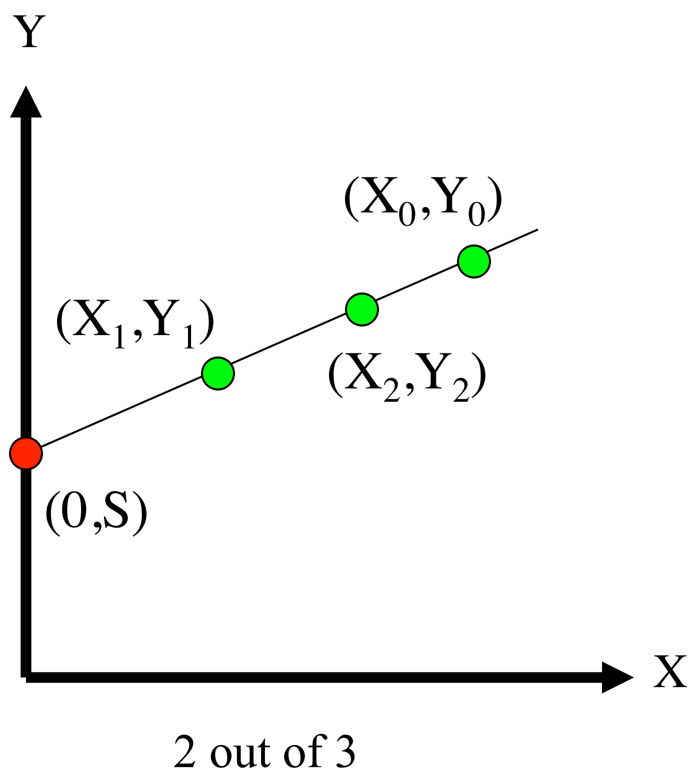


2 out of 2

What if we only want some portion of the principals to cooperate?

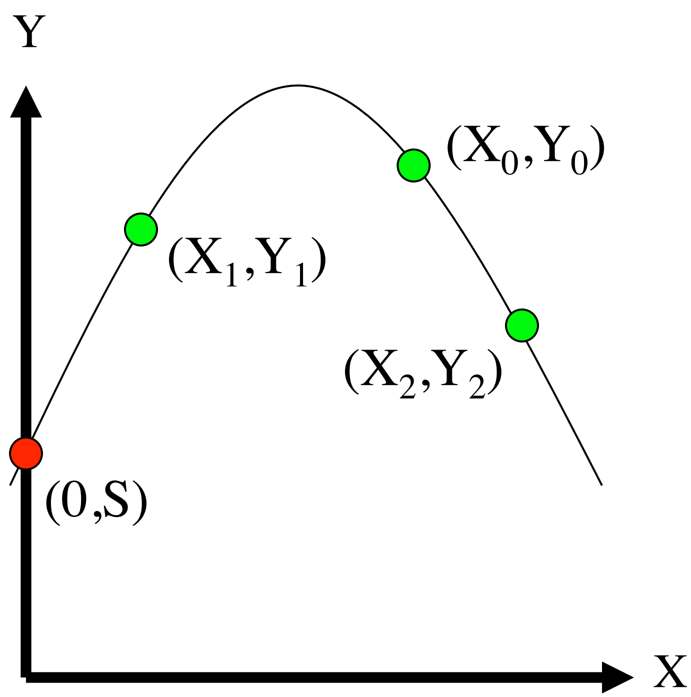
Can we design this approach to support "m out of n"?

Shamir's Secret Sharing



- Give (X_0,Y_0) to Alice
- Give (X_1,Y_1) to Bob
- Give (X_2,Y_2) to Charlie
- Then any two can cooperate to find secret S
- But one can't find secret S
- A "2 out of 3" scheme

Shamir's Secret Sharing



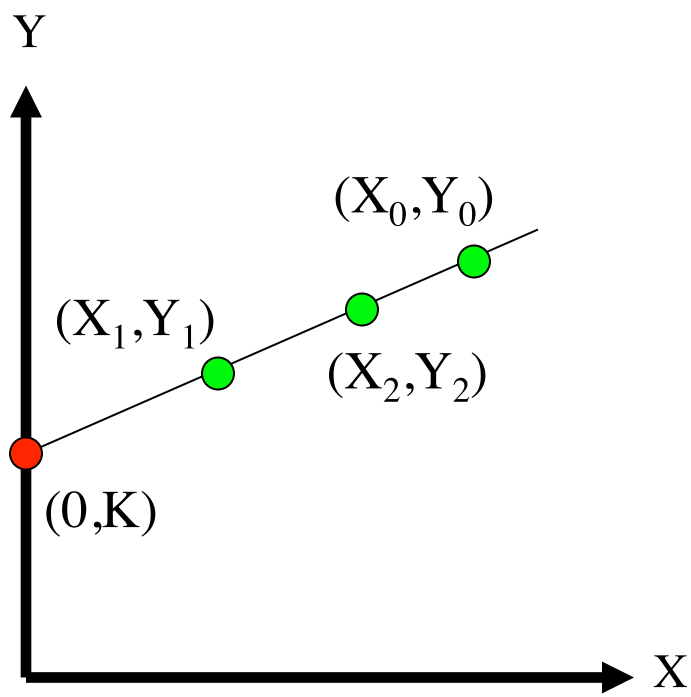
3 out of 3

- Give (X_0,Y_0) to Alice
- Give (X_1,Y_1) to Bob
- Give (X_2,Y_2) to Charlie
- 3 pts determine parabola
- Alice, Bob, **and** Charlie must cooperate to find S
- A "3 out of 3" scheme
- What about "3 out of 4"?

Secret Sharing Example

- **Key escrow** — suppose it's required that your key be stored somewhere
- Key can be "recovered" with court order
- But you don't trust FBI to store your keys
- We can use secret sharing
  - Say, three different government agencies
  - Two must cooperate to recover the key

Secret Sharing Example



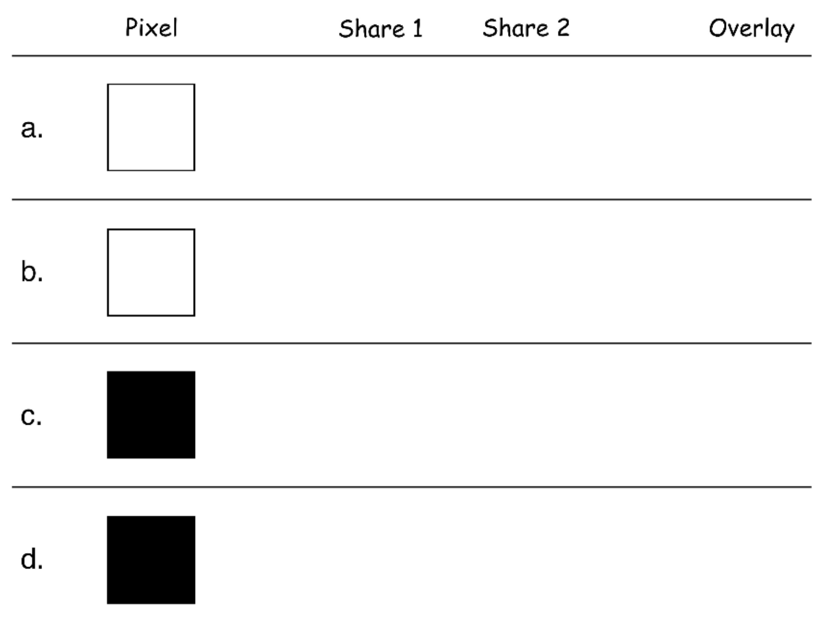
- Your symmetric key is K
- Point (X_0,Y_0) to FBI
- Point (X_1,Y_1) to DoJ
- Point (X_2,Y_2) to DoC
- To recover your key K, two of the three agencies must cooperate
- No one agency can get K

Visual Cryptography

- Another form of secret sharing…
- Alice and Bob “share” an image
- Both must cooperate to reveal the image
- Nobody can learn anything about image from Alice’s share or Bob’s share
  - That is, both shares are required
- Is this possible?

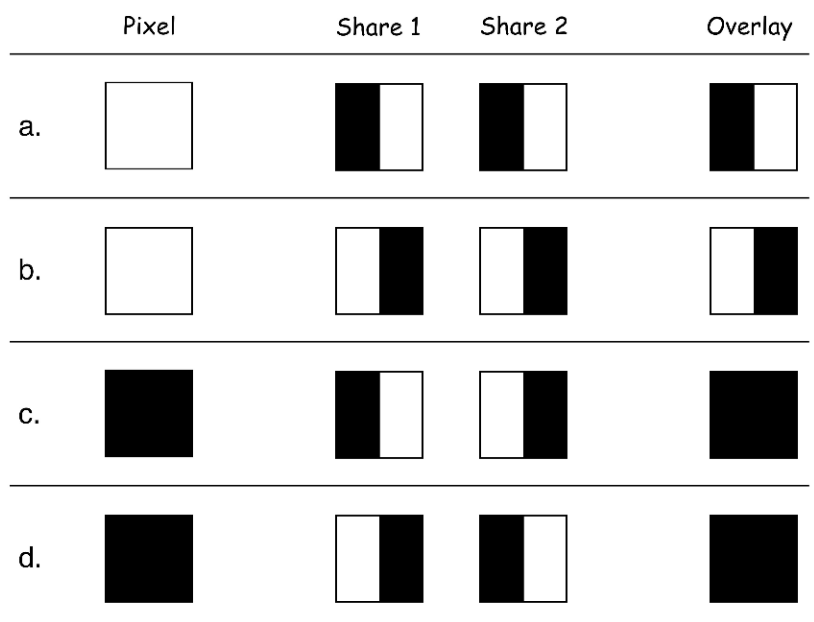
Visual Cryptography

- How to share a pixel?
- Suppose image is black and white
- Then each pixel is either black or white
- We split pixels as shown



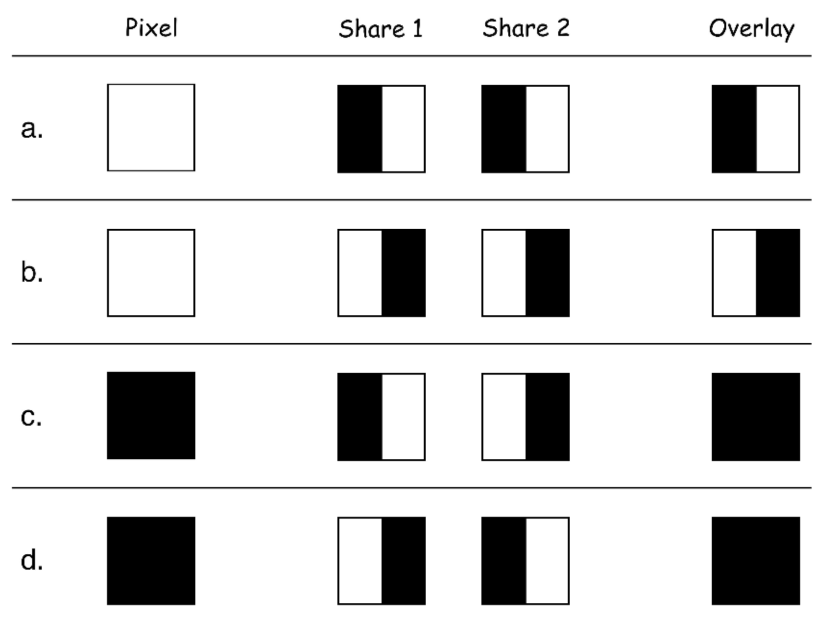
Visual Cryptography

- How to share a pixel?
- Suppose image is black and white
- Then each pixel is either black or white
- We split pixels as shown



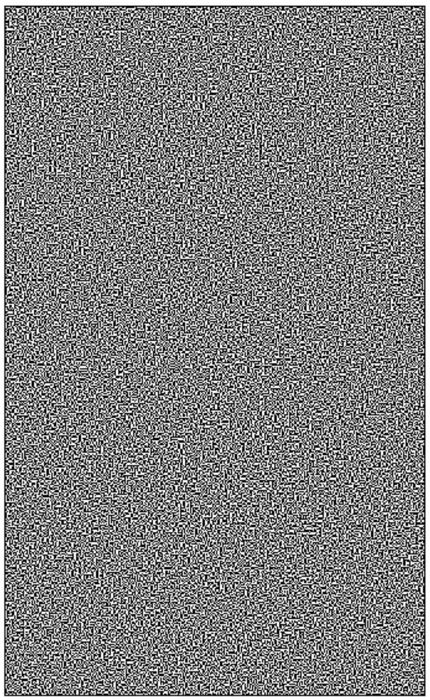
Sharing a B&W Image

- If pixel is white, randomly choose a or b for Alice's/Bob's shares
- If pixel is black, randomly choose c or d
- **No information** in one "share"

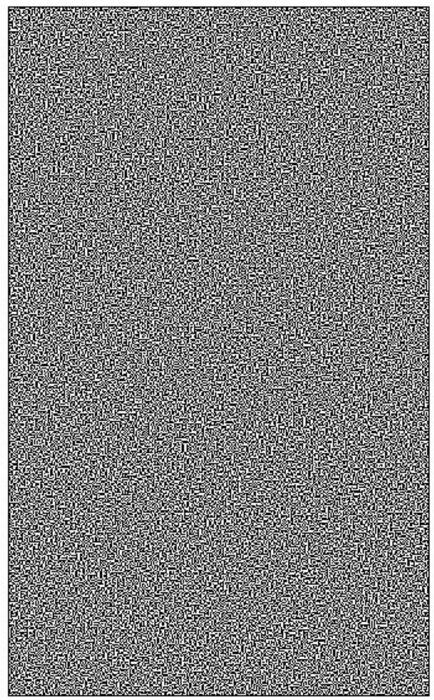


Visual Crypto Example

- Alice's share



- Bob's share



- Overlaid shares

Visual Crypto

- How does visual “crypto” compare to regular crypto?
- In visual crypto, no key…
  - Or, maybe both images are the key?
- With encryption, exhaustive search
  - Except for a one-time pad
- Exhaustive search on visual crypto?
  - No exhaustive search is possible!

Visual Crypto

- Visual crypto — no exhaustive search…
- How does visual crypto compare to crypto?
  - Visual crypto is "information theoretically" secure — true of other secret sharing schemes
  - With regular encryption, goal is to make cryptanalysis computationally infeasible
- Visual crypto an example of **secret sharing**
  - Not really a form of crypto, in the usual sense

Could we design a secret sharing system using one-time pads?

Random Numbers in Cryptography

"Random" Numbers

- Random numbers are widely used outside of security:
  - statistical modeling
  - simulations
  - random samplings
- For these uses, numbers need to be "statistically random" (they need to *appear* to be random).

Random Numbers in Security

- Random numbers used to generate **keys**
  - Symmetric keys
  - RSA: Prime numbers
  - Diffie Hellman: secret values
- Random numbers used for nonces
  - Sometimes a sequence is OK
  - But sometimes nonces must be random
- These numbers must be difficult to guess.

Random Numbers

- Cryptographic random numbers must be statistically random and **unpredictable**
- Suppose server generates symmetric keys…
  - Alice: K_A
  - Bob: K_B
  - Charlie: K_C
  - Dave: K_D
- But, Alice, Bob, and Charlie don't like Dave
- Alice, Bob, and Charlie working together must not be able to determine K_D

Non-random Random Numbers

- Online version of Texas Hold 'em Poker
  - ASF Software, Inc.

- Random numbers used to shuffle the deck
- Program did not produce a random shuffle
- A serious problem or not?

Card Shuffle

- There are 52! > 2^{225} possible shuffles
- The poker program used "random" 32-bit integer to determine the shuffle
  - So, only 2^{32} distinct shuffles could occur
- Code used Pascal pseudo-random number generator (PRNG): Randomize()
- Seed value for PRNG was function of number of milliseconds since midnight
- Less than 2^{27} milliseconds in a day
  - So, less than 2^{27} possible shuffles

Card Shuffle

- Seed based on milliseconds since midnight
- PRNG re-seeded with each shuffle
- By synchronizing clock with server, number of shuffles that need to be tested < 2^{18}
- Could then test all 2^{18} in real time
  - Test each possible shuffle against "up" cards
- Attacker knows **every card** after the first of five rounds of betting!

Poker Example

- Poker program is an extreme example
  - But common PRNGs are predictable
  - Only a question of how many outputs must be observed before determining the sequence
- Crypto random sequences not predictable
  - For example, keystream from RC4 cipher
  - But "seed" (or key) selection is still an issue!
- How to generate initial **random** values?
  - Keys (and, in some cases, seed values)

What is Random?

- True "randomness" hard to define
- **Entropy** is a measure of randomness
- Good sources of "true" randomness
  - Radioactive decay — radioactive computers are not too popular
  - Hardware devices — many good ones on the market
  - Lava lamp — relies on chaotic behavior

Randomness

- Sources of randomness via software
  - Software is (hopefully) deterministic
  - So must rely on external "random" events
  - Mouse movements, keyboard dynamics, network activity, etc., etc.
- Can get **quality** random bits by such methods
- But **quantity** of bits is very limited

The Bottom Line

"The use of pseudo-random processes to generate secret quantities can result in pseudo-security"

Information Hiding

A boat, beneath a sunny sky
Lingering onward dreamily
In an evening of July —
Children three that nestle near,
Eager eye and willing ear,
...
— Lewis Carroll, *Through the Looking Glass*

A boat, beneath a sunny sky
Lingering onward dreamily
In an evening of July —
Children three that nestle near,
Eager eye and willing ear,
...
— Lewis Carroll, *Through the Looking Glass*

Information Hiding

- Digital Watermarks
  - Example: Add "invisible" identifier to data
  - Defense against music or software piracy
- Steganography
  - "Secret" communication channel
  - Similar to a **covert channel** (more on this later)
  - Example: Hide data in image or music file

Watermark

- Add a “mark” to data
- Visibility of watermarks
  - Invisible — Watermark is not obvious
  - Visible — Such as **TOP SECRET**
- Robustness of watermarks
  - Robust — Readable even if attacked
  - Fragile — Damaged if attacked

Watermark Examples

- Add **robust invisible** mark to digital music
  - If pirated music appears on Internet, can trace it back to original source of the leak
- Add **fragile invisible** mark to audio file
  - If watermark is unreadable, recipient knows that audio has been tampered (integrity)
- Combinations of several types are sometimes used
  - E.g., visible plus robust invisible watermarks

Watermark Example (1)

- Non-digital watermark: U.S. currency

- Image embedded in paper on rhs
  - Hold bill to light to see embedded info

Watermark Example (2)

- Add **invisible** watermark to photo
- Claimed that 1 $inch^2$ contains enough info to reconstruct entire photo
- If photo is damaged, watermark can be used to reconstruct it!

Steganography

- According to Herodotus (Greece 440 BC)
  - Shaved slave's head
  - Wrote message on head
  - Let hair grow back
  - Send slave to deliver message
  - Shave slave's head to expose message — warning of Persian invasion
- Historically, steganography used more often than cryptography

Images and Steganography

- Images use 24 bits for color: **RGB**
  - 8 bits for red, 8 for green, 8 for blue
- For example
  - **0x7E 0x52 0x90** is this color
  - **0xFE 0x52 0x90** is this color
- While
  - **0xAB 0x33 0xF0** is this color
  - **0xAB 0x33 0xF1** is this color
- Low-order bits don’t matter…

Images and Stego

- Given an uncompressed image file…
  - For example, BMP format
- …we can insert information into low-order RGB bits
- Since low-order RGB bits don’t matter, result will be “invisible” to human eye
  - But, computer program can “see” the bits

Stego Example 1

- Left side: plain Alice image
- Right side: Alice with entire *Alice in Wonderland* (pdf) “hidden” in the image

Non-Stego Example

- Walrus.html in web browser

"The time has come," the Walrus said,
"To talk of many things:
Of shoes and ships and sealing wax
Of cabbages and kings
And why the sea is boiling hot
And whether pigs have wings."

- "View source" reveals:

```
<font color=#000000>"The time has come," the Walrus said,</font><br>
<font color=#000000>"To talk of many things: </font><br>
<font color=#000000>Of shoes and ships and sealing wax </font><br>
<font color=#000000>Of cabbages and kings </font><br>
<font color=#000000>And why the sea is boiling hot </font><br>
<font color=#000000>And whether pigs have wings." </font><br>
```

Stego Example 2

- stegoWalrus.html in web browser

"The time has come," the Walrus said,
"To talk of many things:
Of shoes and ships and sealing wax
Of cabbages and kings
And why the sea is boiling hot
And whether pigs have wings."

- "View source" reveals:

```
<font color=#000101>"The time has come," the Walrus said,</font><br>
<font color=#000100>"To talk of many things: </font><br>
<font color=#010000>Of shoes and ships and sealing wax </font><br>
<font color=#010000>Of cabbages and kings </font><br>
<font color=#000000>And why the sea is boiling hot </font><br>
<font color=#010001>And whether pigs have wings." </font><br>
```

- "Hidden" message: **011 010 100 100 000 101**

Using Steganography

Alice

Message encoded in
low-bits of image

Message encoded in
low-bits of image

Bob

What are they
up to?

What can
Trudy do?

Trudy

Some formats (e.g. HTML) are easier for people to read, but no harder for computers.

We can hide info in unimportant bits, but Trudy can overwrite those bits as well.

Steganography

- Easy to hide info in **unimportant bits**
- Easy to **destroy** info in unimportant bits
- To be robust, must use **important bits**
  - But stored info must not damage data
  - Collusion attacks are another concern
- Robust steganography is tricky!

Information Hiding: The Bottom Line

- Not-so-easy to hide digital information
  - "Obvious" approach is **not** robust
  - **Stirmark:** tool to make most watermarks in images unreadable without damaging the image
  - Stego/watermarking active research topics
- If information hiding is suspected
  - Attacker may be able to make information/watermark unreadable
  - Attacker may be able to read the information, given the original document (image, audio, etc.)