Computing Is Pretty Strange

Encryption & Steganography: Amazing Things To Do with Bits

Lawrence Snyder
University of Washington, Seattle

Encryption

- Encryption is the process of "scrambling" data so it is difficult (impossible?) to understand it
- We encrypt data to keep it private
- Every site that you use as https:// is encrypted
- Familiar example: Caesar cipher:

C: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
E: D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

- What would Julius be encrypted to?

Encryption

- Encryption is the process of "scrambling" data so it is difficult (impossible?) to understand it
- We encrypt data to keep it private
- Every site that you use as https:// is encrypted
- Familiar example: Caesar cipher:

C: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
E: D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

- What would Julius be encrypted to? Mxolxv

More Typically …

- The fixed shift of an alphabet is easy to break

Alternate:

- Sender uses a key, k, to multiply clear byte sequences (recall they're numbers) by k

-- Send encrypted result – looks like gibberish --

- Receiver divides by k to decrypt getting clear

Example

- Let the clear be: "MEET @ 9" and k=13
- Break clear text into 2-letter sequences:
  - ME ET ♭@ ♭9
- Interpret text as numbers
  - 7769 6984 3264 3257
- Multiply by key:

7769×13=100997
6984×13=090792
3264×13=042432
3257×13=042341

- Send encrypted (6-digit) number
- Receiver does the reverse process …

An Alternate: Public Key Encrypt

- The problem with "private key" encryption: the two sides have to meet to agree on key
- Public Key fixes this: The receiver publishes (on Web site, say) a (very very special) key, K
- More importantly, the theory it uses means that *NO practical amount of computing can break the code*
- Here's what you do …

Public Key Process

- Sender breaks up the message into blocks as before
- Sender cubes each block – yup, raises to the 3rd power – and mods it by K, i.e. (<text>3)%K
- Transmit results
- Receiver raises each remainder to a high power determined by prime numbers & known only to him
- Receiver mods by *K*, too, which are

– surprisingly – the original blocks!

- The receiver assembles the message
- Thanks to Euler and Diffie & Hellman

This Is Amazing!!!

Steganography

- The process of hiding information
- Two Greek roots meaning:

  "stego" == "roof" "stega" == "cover"

Why Hide Information?

- Most common reason to hide information is to avoid being caught with it
  - Military and spy documents
  - Repressive governments restricting news/info
  - Avoid others "snooping" – privacy
- Hiding is different than encryption … uses the fact that the searcher doesn't know it's there

Illustrate A Way To Do It

- The Plan …
  - hide "subversive" protest picture in "calendar art"

Guest Image

Host Image

Step 1: Reduce Bits of Guest

- We don't need all of the bits in RGB to get a decent picture

All bits

1011 0100 1101 0011 0001 1100

Left 2 bits of each color

10~~11 0100~~ 11~~01 0011~~ 00~~01 1100~~

Step 2: Replace Bits In Host

- Put guest bits into right 2 bits of host

1111 0100 1101 0011 1011 1101

1011 0100 1101 0011 0001 1100

1111 0110 1101 0011 1011 1100

Compare fog.jpg with stegFog.png

fog.jpg

stegFog.png

Really?
Just Do It!

Let's Look At Them

… and then we'll see the details

Processing Code For Guest→Host

```
PImage crowd, fog;
int i = 0;
int srcw=512;
int srch=346;
int wid=450;
int hi=300;
color c, cprime;

void setup( ) {
  size(srcw, srch);
  crowd = loadImage("egypt.jpg");
  fog = loadImage("fog.jpg");
  image(fog,0,0);
  for (int i=0; i<srcw; i++){
    for(int j=0; j<srch; j++) {
      c = get(i,j);
      if (i<wid && j<hi) {
        cprime=crowd.get(i,j);
        cprime=color(4*(int(red(c))/4) + (int(red(cprime))/64),
                     4*(int(green(c))/4) + (int(green(cprime))/64),
                     4*(int(blue(c))/4) + (int(blue(cprime))/64));
        set(i,j, cprime);
      } else {
        set(i,j,c);
      }
    }
  }
}
```

Encoding Code

```
void draw( ) {
  if (mousePressed) {
    saveFrame("stegFog.png");
  }
}
```

Code To Save Result on Click

How Does It Work

- After the pictures are loaded

```
cprime=color(4*(int(red(c))/4) + (int(red(cprime))/64),
             4*(int(green(c))/4) + (int(green(cprime))/64),
             4*(int(blue(c))/4) + (int(blue(cprime))/64));
```

Clear right 2 bits of host

Extract left 2 bits of guest

New combined color

Code To Extract Image

```
PImage fog;
int flip = 0;
int srcw=512;
int srch=346;
int wid=450;
int hi=300;
color c, cprime;

void setup( ) {
  size(srcw, srch);
  fog = loadImage("stegFog.png");
  image(fog,0,0);
 }

 void draw( ) {
   if (mousePressed) {
     for (int i=0; i<srcw; i++){
     for(int j=0; j<srch; j++) {
      c = get(i,j);
      if (i<wid && j<hi) {
         cprime=color(64*(int(red(c))%4),
                      64*(int(green(c))%4),
                      64*(int(blue(c))%4));
         set(i,j, cprime);
      } else {
        set(i,j,c);
      }
     }
    }
   }
 }
```

How Does It Work

- Read in the file, and then on mouse click, pull out the bits and make a picture

```
cprime=color(64*(int(red(c))%4),
             64*(int(green(c))%4),
             64*(int(blue(c))%4));
```

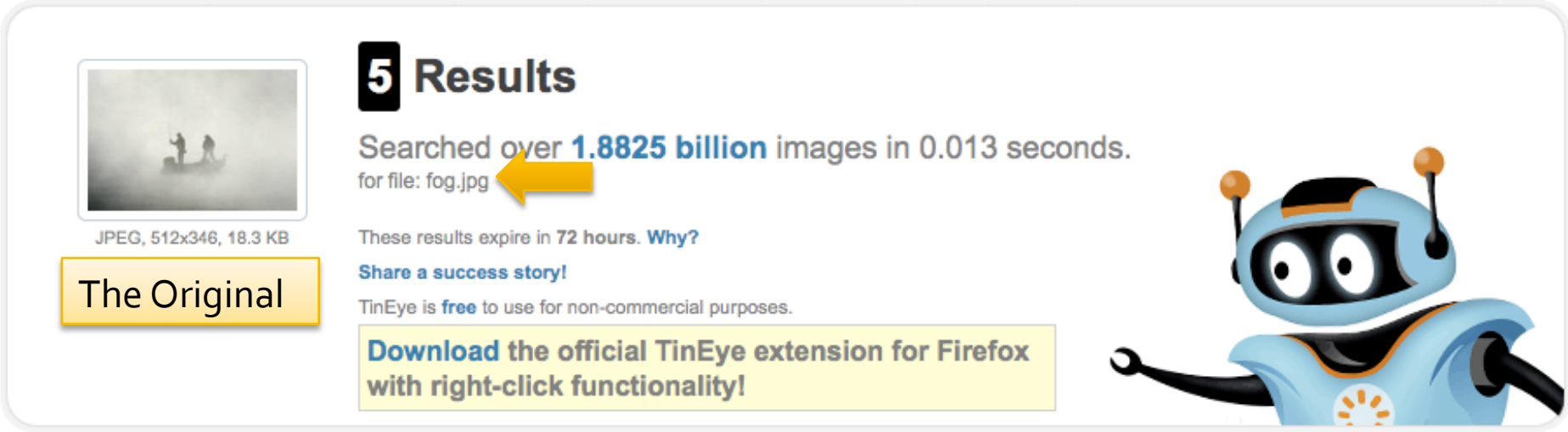
Remove right 2 bits

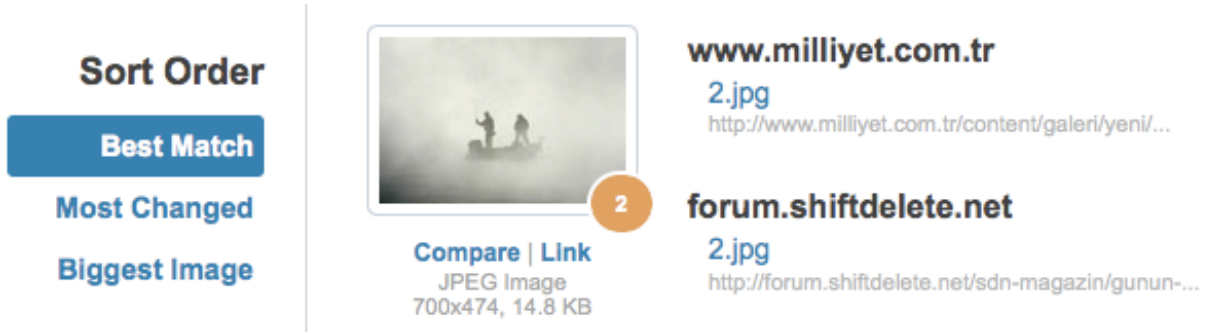
Make them left 2 bits for each color

New color

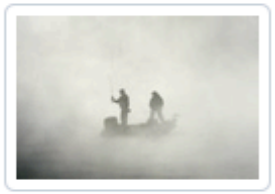
How Much Is Coded Like Original?

- Run A Test … www.tineye.com





Check The "Steganized" File



PNG, 512x346, 144.4 KB

Steganized

5 Results

Searched over **1.8825 billion** images in 2.609 seconds.
for file: stegFog.png

These results expire in **72 hours**. **Why?**

Share a success story!

TinEye is **free** to use for non-commercial purposes.

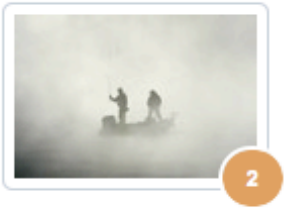
Download the official TinEye extension for Firefox with right-click functionality!

Sort Order

Best Match

Most Changed

Biggest Image



2

Compare | **Link**
JPEG Image
700x474, 14.8 KB

www.milliyet.com.tr
2.jpg
http://www.milliyet.com.tr/content/galeri/yeni/...

forum.shiftdelete.net
2.jpg
http://forum.shiftdelete.net/sdn-magazin/gunun-...