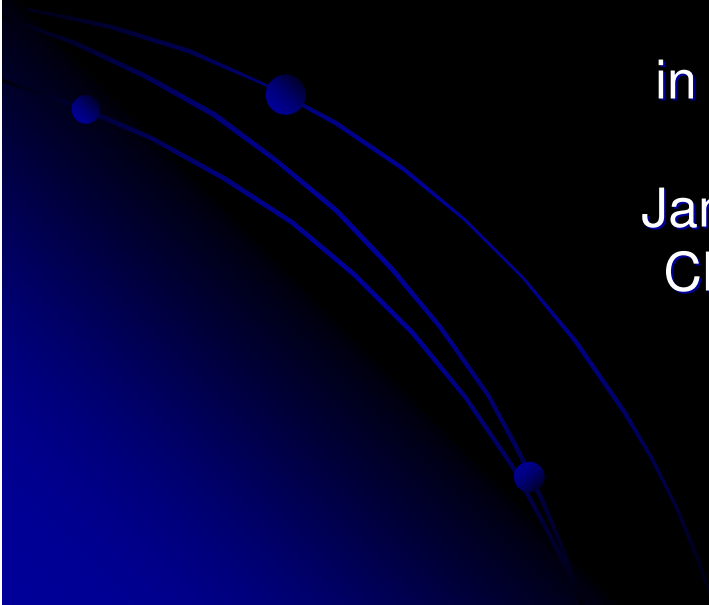


Le'go My Stego

Steganography
in the post Web 2.0 World

James Shewmaker © 2008
ChicagoCon Spring 2008



Today's Agenda

- Today's Agenda
 - Background: Classical Stego
 - Digital Stego
 - Text, Audio, Images/Video
 - Behaviour (protocol timing or inferences)
 - LiveLinkedMyFaceFlickrTubester et al
 - New filesystem: LLMFFT-FS
 - Propagated Aggregate Network-Trunk RAID

Classic Old School Stego

- **Spy vs. Spy**
 - Classified Ads
 - Microdot
- Any media file that samples reality can have its Least Significant Bit (LSB) tweaked with minor disruptions

Classic Digital Stego

- Classic Digital Stego
 - Manipulating that Least Significant Bit
 - Using unused space in a host file
- Examples
 - Digital audio – fairly subtle
 - Even magnitude == zero
 - Odd magnitude == one
 - Digital image – also fairly subtle
 - Even LSB of a pixel == zero
 - Odd LSB of a pixel == one

Distributed Stego

- Newer online videos are often converted to flash video (FLV)
- Take your favourite viral marketing video
 - Encode to FLV before you upload
 - Store data with LSB stego using each frame/tag/box
 - (GIF/PNG/JPEG, etc.)
 - Store parity bit with each audio sample
- Classic/Simple Stego is not quite robust enough to survive video conversions
 - High redundancy might survive conversion
 - If we pick our codec well, it might survive unmolested

Phfft—who needs binary anyway?

- Whitespace in public blog comments
 - Seed arbitrary blog with keywords, then ask Google to find the blog
 - \x20 between words == zero
 - \x20\x20 between words == one
- Misspelt blog comments
 - the == zero
 - teh == one
- These techniques are compressible and subtle enough to likely be overlooked if somebody suspects stego

Creating a Stego Filesystem

- Previous slides could be used for data or metadata
- Pick a method to encode a structure, ie:
 - Use blog comments as metadata for a dually-linked list
 - URL to previous metadata comment
 - URL to datablock
 - URL to next metadata comment
 - Store datablock in video frame/tag/box
 - Store an extra parity bit for the datablock in the audio sample

How the data survives conversion

- Small bit errors from conversion could be detected and corrected with Hamming code-like techniques to survive conversion
- RAID 10 the metadata dually-linked list
 - That is to say mirrored sets of RAID 5
- Now if LSB bits are lost in a single frame/tag/box-we can recover
- Now if the conversion jacks a frame/tag/box-we can recover

Even Hamming code example

- Every power of 2 is a parity bit (4 extra bits)
- For example, store `\xFF`, blanks are parity
 - `__1_ 111_ 1111`
 - 1st bit checks 1, skips 1, then repeats, 5 ones is odd so we get
 - `1_1_ 111_ 1111`
 - 2nd bit checks 2 bits, skip 2...
(2, 3, 6, 7, 10, 11), 5 ones so we get
 - `111_ 111_ 1111`
 - (4, 5, 6, 7, 12), 4 ones so we get
 - `1110 111_ 1111`
 - `1110 1110 1111` –Final encoded

Fixing a bad bit

- 1110 1110 1111 –Final Encoded
- 1110 1110 1011 –Damaged
- [^] [^] -Lies!
- 2 + 8 =10 -bit 10 is bad!
- 1110 1110 1111 -Corrected!
- This will detect 2 bit errors, but correcting more than 1 error requires wrapping all of this parity with more checks

Automating this in reality

- These structures could hold anything
- Put the structures in arbitrary places
 - Some sites mirror
 - Some thieves plagiarize (almost as good as a mirror)
- Ask Google to find them when needed
- “Drive Maintenance” – periodically look up with Google, upload any necessary pieces (to keep redundancy from getting weak)

Alpha Implementation

- LLMFFT-FS over Propagated Aggregate Network-Trunk RAID (PAN-T RAID)
 - You've seen gmailfs—same idea
 - Alpha code written in perl
 - Uses older FLV format
 - Need to rewrite for flash 9 before releasing
 - POC only, no intention of maintaining even a beta quality project
 - Hope to release by August 2008

Bonus Round

- Can we build a pattern out of key frames? (key frames used to seek)
 - Two close keyframes = zero, two sparse = one
- FLV's metadata info frames
 - Store more stego
 - Store a hash/signature to identify datablock and/or datablock tampering
- I'm not Dan Kaminsky, but if I was I'd stash an index in somebody else's DNS ...

References / For More Info

- FLV- <http://www.adobe.com/devnet/flv/>
- Hamming code-
http://en.wikipedia.org/wiki/Hamming_code#General_algorithm