

## Diffusion

### and a Key-Recovery Attack on a WM Scheme by Li and Yuan

(Hans) Georg Schaathun

Department of Computing  
University of Surrey

22-23 September 2008



## Watermarking is not Cryptography

Ingemar Cox

*If we don't study watermarking as a cryptographic problem, how do you know that cryptanalysis cannot break it?*

- If it *can be* cast as a cryptographic problem
  - you have to use cryptology in the design,
  - because your adversary may use it in the attack
- Cryptology is a methodology, not just a series of primitives
- Admittedly, Li-Yuan is better seen as a layered system
  - We break the cryptological layer
  - We do not touch the watermarking layer (embedding)
- i.e. Cox' view may stand ... for now



## Do not reuse the key

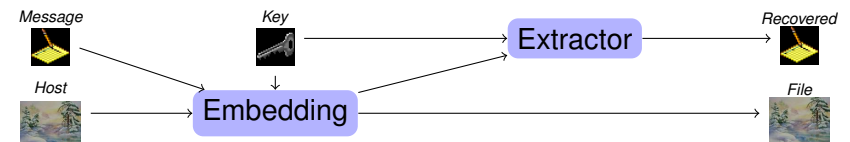
Andrew Ker

- Keys *are* reused in cryptography
  - The one-time pad is not practical
- The solution is **diffusion**
  - Each key bit is spread widely across output
  - Dependency between key and output is *too complex* for analysis
- We shall see *lack of diffusion* later (stay awake)



## Authentication and Watermarking

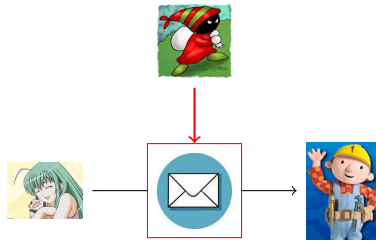
### Digital Watermarking



- Digital Watermarking 'hides' a message in another file (the *host*)
- The watermarked image can replace the cover
  - *Perceptually Equivalent*
- In fragile watermarking
  - The host cannot be modified without destroying the hidden message
- In robust watermarking
  - The hidden message cannot be modified or destroyed without destroying the host



## The Authentication Problem



- Alice sends a message to Bob
- Bob wants to assure that it is authentic
- Eve wants to modify the message and fool Bob

## Authentication Techniques

- Cryptographic solutions
  - Message Authentication Code (MAC) – Secret Key
  - Digital Signatures – Public Key
- Watermarking embeds Authentication Information in the file
  - no appended signature to handle
  - everything fits into the host file format
- Creating and attacking the authentication information
  - remains a cryptological problem
  - layered system (here Cox and I agree)
- It does not matter if the designer agrees
  - I, as an attacker, can use cryptology anyway

## Cryptography

### Authentication Techniques

- Cryptographic solutions
  - Message Authentication Code (MAC) – Secret Key
  - Digital Signatures – Public Key
- Certificate of Authenticity (Signature or MAC)
  - ... appended to the message
  - **does not fit into standard file formats**
- Only Alice can produce a valid certificate
  - well-studied and trusted technology
  - mathematical security

## Authentication Watermarking

- Authentication information is embedded in the file
  - no appended signature to handle
  - everything fits into the host file format
- Some watermarking systems offer extra advantages
  - localisation of changes/errors
  - further analysis of modification processes
- Creating and attacking the authentication information
  - remains a cryptological problem
  - layered system (here Cox and I agree)

## The Li-Yuan System

### Symbols and definitions

- $M \times N$  8-bit grayscale image  $\mathcal{I}(x, y)$
- Security parameter  $b$ 
  - Discard the  $b$  least significant bits of each pixel
  - $\rightarrow$  *significant image*  $\mathcal{S}(x, y)$
- Secret watermark image  $\mathbf{w}$ 
  - $M \times N$  matrix of  $b$ -bits per item (pixel)
  - A shorter key can be expanded using a secure PRNG
- Let  $a(x, y)$  denote the authentication information
  - $b$  bits per pixel (to be computed)
- The watermarked image will be generated as

$$\mathcal{W}(x, y) = 2^b \mathcal{S}(x, y) + a(x, y),$$



## Extraction and Authentication

- Extraction
  - $v(x, y)$  is computed (hash of  $\mathcal{S}$ )
  - $a(x, y)$  is extracted directly ( $= \mathcal{I} \bmod 2^b$ )
  - Extracted watermark  $w'(x, y) = v(x, y) \oplus a(x, y)$
  - Secret watermark  $w(x, y)$  is known
- $w'(x, y) \neq w(x, y)$  indicates an error



## A non-cryptographic hash

### Calculating the authentication information

- Main challenge: calculating  $a(x, y)$ 
  - if Eve can calculate  $a(x, y)$  for a false image,
  - ... she has broken the scheme

For each pixel  $(x, y)$ ,

- Consider a  $k \times k$  square region  $N_k(x, y)$  around it
- A  $b$ -bit hash  $v(x, y)$  is calculated from
  - 1  $\mathcal{S}$  on  $N_k(x, y)$
  - 2 least significant bits of  $\mathbf{w}$  on  $N_k(x, y)$
- $a(x, y) = v(x, y) \oplus w(x, y)$  replace  $b$  LSB-s



## The problem

- Each watermarked pixel  $(x, y)$  depend on 26 key bits
  - This includes  $5 \times 5$  bits of  $\kappa := w \bmod 2$
  - And one extra bit  $w(x, y)$  'encrypting'  $v(x, y)$
- A key principle of cryptography is **diffusion**
  - Each output bit should depend on every key bit
- Dependence on 26 bits is insufficient
  - An exhaustive search is possible
  - work on 25 bits of  $\kappa$  at a time
- Proper *Diffusion* would prevent the attack



## Assumptions

- We need two known, watermarked images  $\mathbf{x}_1, \mathbf{x}_2$ 
  - One image is not sufficient
  - More images give faster decoding
- We assume  $k = 5$ 
  - We sketch improvements to be feasible for  $k > 5$
  - ... but the details remain for future work
  - ... the improvements depend on image properties
- We assume  $b = 2$ 
  - $b > 2$  makes the attack **faster**
  - $b = 1$  makes it slower, but additional images can compensate
  - (Note that Li and Yuan claim that increasing  $b$  increases security)



## The idea

### The first round

- Consider a  $5 \times 5$  block at a time
- Exhaustive search :  $2^{25}$  possible subkeys  $\kappa | N_5(x, y)$
- For each tentative subkey  $\hat{\kappa}$ 
  - 1 Extract watermark  $w'_i(x, y)$  ( $i = 1, 2$ ) from  $\mathbf{x}_i$
  - 2 Compare  $w'_1$  and tentative key
    - $w'_1(x, y) \bmod 2 \neq \hat{\kappa}(x, y)$  : reject  $\hat{\kappa}$
  - 3 Compare  $w'_1$  and  $w'_2$ 
    - $w'_1(x, y) \neq w'_2(x, y)$  : reject  $\hat{\kappa}$
- Three (3) bit comparisons are made
  - On average, one key in eight ( $2^3$ ) pass the test



## How to proceed

### The rest of the idea

- Each round considers a new  $5 \times 5$  block
  - ... overlapping with the first
- Number of possible keys *increase at first*
- Rounds 2-3 add five key pixels each
- Round 4 add only 1 ( $6 \times 6 = 36$  pixels total)
- Rounds 5 and 7 add five pixels each
- Rounds 6, 8, and 9 add one pixel each
  - $7 \times 7 = 49$  pixels covered after Round 9
- Thereafter: expected number of tentative keys *will decrease*



## Strong cryptography

- Two problems
  - Short key : weak 'cryptography' at best
    - ... exploited by the basic attack
  - Insufficient diffusion : non-cryptographic
    - ... exploited by improvements (paper only)
- $a(x, y)$  requires the properties of a MAC
  - Eve knows several watermarked images (with  $S$  and  $a$ )
  - Eve cannot produce a new image  $S'$  with matching authentication information ( $a'$ ).
- A proper MAC would prevent our attack
  - There are some works using MAC-s in authentication watermarking
  - ... and some works recognise the importance of cryptography, but use the wrong cryptographic properties.



## The Design Parameters

- Decreasing  $b$ 
  - Fewer keys are excluded in each round
  - But hash collisions become more frequent
- Increasing  $k$ 
  - More keys to consider per round
  - However, if a monochrome region can be found in the image,
    - Only  $k^2$  (not  $2^{k^2}$ ) keys have to be considered
    - By exploiting the simple additive structure of  $S(x, y)$
    - And increasing  $k$  will have marginal effect...



## Conclusion

- Key-Recovery Attack Algorithm on Li and Yuan's Scheme
- Cryptological principles apply
  - If the designer ignores them,
  - then the attacker can exploit them
- Open problem
  - Implement and test the algorithm
  - How secure are other watermarking systems?

