Copyright Protection

- Digital collections are very easy to copy:
  - Music, videos, text...
  - Easy to protect digital copyrighted content
  - An active topic in this lecture.
- Should content be protected? (not our main topic)
  - Music is often shared for personal enjoyment.
  - It should not conflict with intellectual property doctrine.
- Can content be protected?
  - Persistent piracy is considered in copying.
  - Technology can potentially prevent small-scale copying:
    - Keeping honest people honest.

Method 1: Copyright Crawlers

- From here on all oys use music as an example.
- Suppose we had a ContentAware hash function:
  - $O : \text{music} \rightarrow \text{short strings}$
  - Satisfying:
    1. $O_1$ and $O_2$ are two clips (e.g. music files) that play the same song then $O_1 = O_2$.
    2. Given a clip $O$ a pirate cannot create an acceptable clip $O'$ such that $O = O'$.
- A hash function must resist all signal processing tricks.
- Do such hash functions exist?
  - Many do (some claim to have them).

Examples

- iGigarc Interlaced images.
- Digi: Digital images.
- Livakumar: Tampered.
  - crawl is the OIL (looking for academic plagiarism).
  - Several success stories.

Light improvement: Watermarking

- ContentAware hash functions are not effective.
- Idea: at the recording studio embed a hidden Watermark in the music file:
  - $\text{Wmark}(O) \rightarrow \text{watermarked version of music file}$
  - With information embedded in it.
  - Retrieve $\text{Wmark}(O)$ takes the watermarked music file and outputs the embedded Watermark $\text{Wmark}(O)$.

- Properties:
  - Watermark must be inaudible.
  - Watermark should be robust:
    - $O$, $\text{Wmark}(O)$
  - Given a clip $O$ a pirate cannot create an acceptable $O'$ such that $\text{Wmark}(O') = O'$.
  - Note: watermarked content must resist all signal processing tricks.
    - Resampling/cropping/label past (filtering).
Issues

- Copyright crawler uses Retrieve algorithm.

■ Benefits:
  - Copyright crawler does not need to maintain all copyrighted material.
  - Does not need to be hash.
  - Watermarking music seems to be an easier problem.

■ Same problems as before:
  - Does not defend against anonymous postings.
  - High overhead.

Robust Watermarks

- Object typically mined. Retrieve alg is kept secret.

- a robust watermarking system listed.
  - Known:

  - binark: general tool for removing image watermarks.
  - Obvious all watermarking scheme.

  - chaljngs:
    - Broken database all
    - Obje mark

Watermarking Images (>200 papers)

- DigiMarc: embeds creator's serial number.
  - Add or subtract small random quantities from each pixel. Embedded signal kept secret.

- Signafy (NEC).
  - Add small modifications to random frequencies of entire Fourier Spectrum.
  - Embedded signal kept secret.

- Caronni: Embed geometric shapes in background.

- Signum Tech. (SureSign).

Watermarking Music (>100 papers)

- Aris Tech (MusicCode):
  - Date: 2 bits/sec of music

- Solara (EDNA)
  - Used by Liquid Audio.

- Argent:
  - Embed full text information.
  - Frame based: info inserted at random areas of signal
  - Secret key determines random areas.

Method 2: policy watermark

- No copyright crawlers.
- Embed usage policy as watermark in music file.
- Every music player in the world works as follows:
  - Use Retrieve algorithm to check if watermark exists.
  - If so, play music only if policy is satisfied
  - (e.g. payment, authorized player, etc.).

- Big problems with this:
  - How to upgrade all music players? Why would consumers agree?
  - Retrieve algorithm is in the public domain.
  - Makes watermarking an even harder problem.
  - Open source players will ignore embedded policy.
- Seems to be the approach preferred by RIAA.

Method 3: Fingerprinting

- No copyright crawlers. No big brotherigail players.
- Completely passive.
- Basic idea:
  - embed a unique user into each sold copy.
  - If user posts copy to web or Napster, embedded user identifies user.
- Problem:
  - Need ability to create distinct and indistinguishable versions of object.
  - Pollution: two users can compare their objects to find parts of the fingerprint.
race Revoke schemes

Example: DSS

- DSS: Content Scrambling System
  - Used to protect DSS movies.
  - Each DSS player manufacturer i has key $d_i$, e.g., $d_{sony}$
  - Embed same key $d_{sony}$ in all players from Sony.
  - Every DSS movie $M$ is encrypted as follows:
    - $Enc_{key}(M) \times d_{sony}$, a random key.
    - $E_{key}(M) \times d_{sony}$
  - About DSS manufacturer keys.

Problems with DSS

- DSS: Extracted manufacturer key from DSSing software player.
  - Could then decrypt any DSS movie that could be played on the DSSing player.
  - M2AA revoked DSS key: disabled all DSS players.

- Bigger problem:
  - Encryption algorithm in DSS is based on DSSRS.
  - Very fast: video rate decryption on weak DSS player.
  - Very weak: one manuf. key can get all keys.

Better revocation technique

- Basic idea: embed a distinct key in every player.

Players:

- Every node $v$ has an associated key $d_v$.
- Every player corresponds to leaf node.
- Key for player $i$: all keys on path from root to leaf $i$.

Revocation

- Initially: encrypt all content with key at root.
  - Any player can decrypt content.
  - When player $i$ is revoked encrypt content-key so that all players can decrypt other than player $i$. 
ow to tell which player to revoke?

- When pirate publishes single key on Internet, the pirate knows which key to revoke.
- Low can use AAA to tell which keys all added in player.
- Our system can interact with player and determine how to revoke that player.