Using FreeBSD to

Build a Secure Digital Cinema Server

Nate Lawson

Usenix UseBSD SIG

June 29, 2004

Cryptography Research, Inc.

www.cryptography.com

607 Market St., 5th Floor, San Francisco, CA 94105

CRYPTOGRAPHY RESEARCH

Overview

- Introduction
  - Strength vs. assurance
  - From film to digital cinema
- Building a digital cinema server
- Analysis of outsourced storage threats

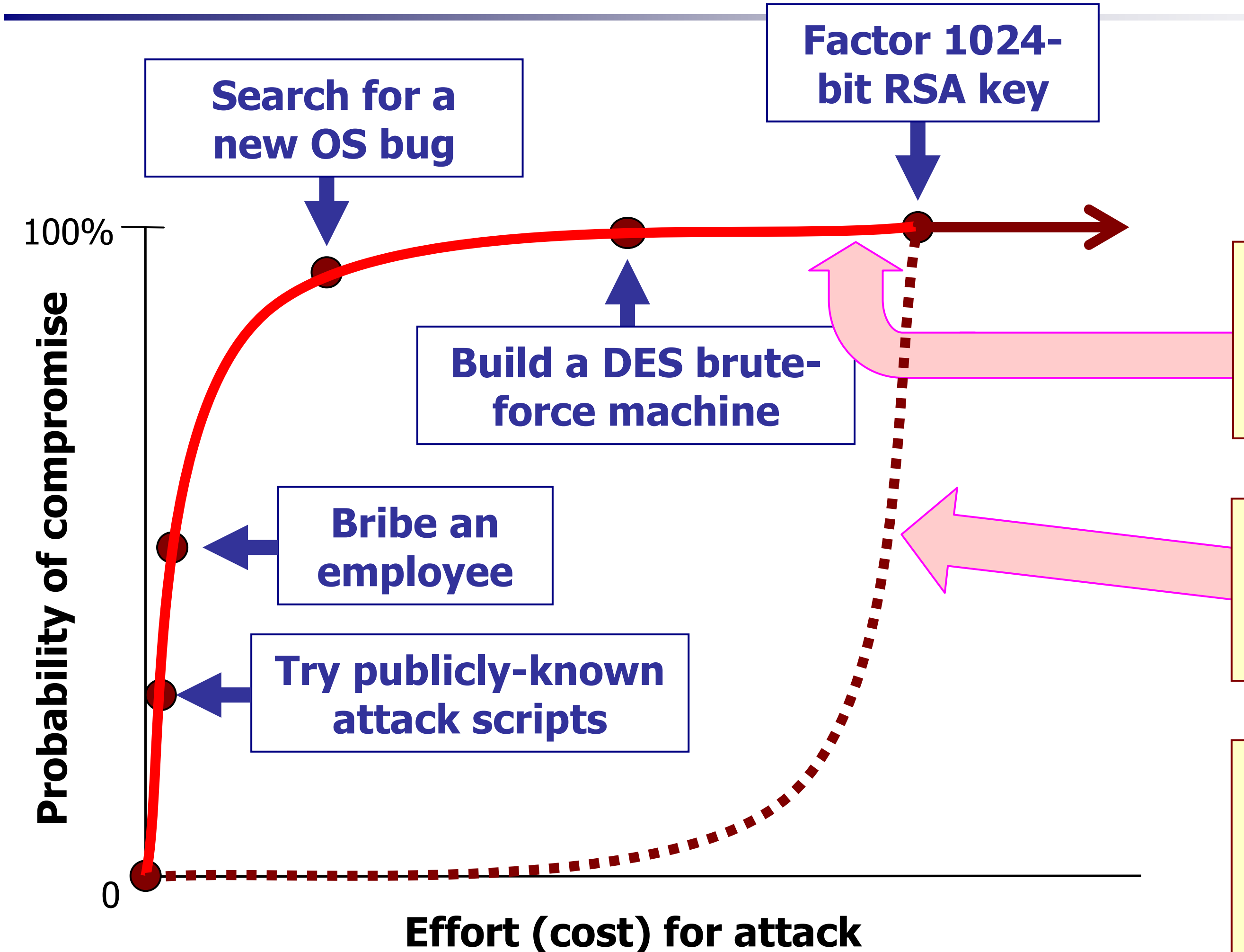
About Cryptography Research, Inc.

- Founded in 1995 by Paul Kocher
- Past projects
  - SSL 3.0, DES cracker
- Recent and ongoing work
  - Differential power analysis (DPA)
  - Tamper resistance
  - Content security for high-def optical disc format
- Seek to anticipate long-term trends and develop "must have" solutions to complex problems
- Provide security technology and services to companies that build and use security products

CRYPTOGRAPHY RESEARCH

Thinking About Security

Measuring Security



If your curve looks like this, factoring is irrelevant.

You want your curve to look like this.

Security is NOT functionality.

It is the strength of the weakest link.

Strength

How strong is the system against known attacks?

Crypto can provide superb strength

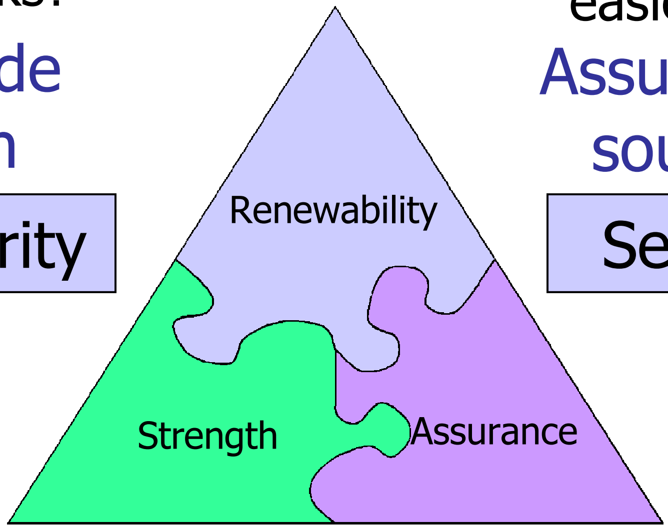
Encryption ≠ Security

Assurance

What are the odds of an easier (unknown) attack?

Assurance comes from sound design/impl.

Secrecy ≠ Assurance



Renewability

What happens after an attack succeeds?

Must be able to respond to the unpredictable

Revocation ≠ Renewability

From Film to Digital Cinema

Traditional Cinema Process

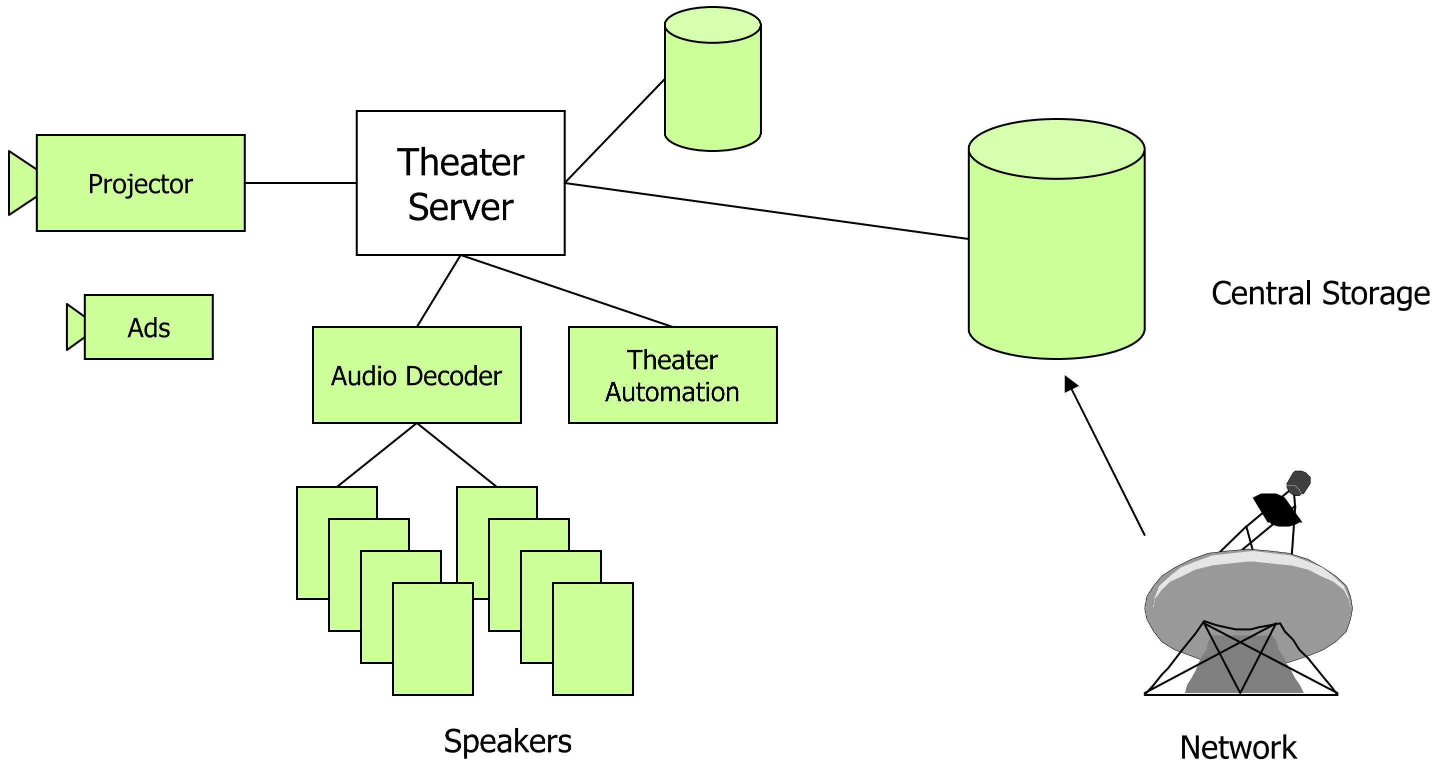
- Production
  - Film cameras
  - Ship dailies via courier
- Post-production (Avid)
  - Transfer film to digital and back
  - Editing, special effects, etc.
- Distribution
  - Make thousands of film prints at $3,000 each
- Projection
  - Projector costs about $30,000
  - Print degrades after a week of viewing

CRYPTOGRAPHY RESEARCH

Digital Cinema Process

- Production
  - Digital cameras at 2K
  - Captured to hard drives
- Post-production (same without transfer steps)
- Distribution
  - Physical media: DVD, tape
  - One-time-use, expensive network connection (demo only)
- Projection
  - Server (GDC, QuVIS, etc.)
    - Stores the movie
    - CODEC (video/audio)
    - Theater control
  - Projector (Christie, Barco, etc.)
    - 2K resolution (some 1K)
    - HD-SDI serial interface for raw video

Digital Theater Equipment



Digital Cinema Market

- 1999
  - 1K projectors and theater servers introduced
- 2000
  - "Titan A.E." transmitted via fiber in Cisco demo
  - "Bounce" transmitted via satellite
  - 30 digital theaters in U.S.
- 2003
  - Loews announces all theaters will have HD (not DC)
  - Digital Cinema Initiative (DCI) requests proposals
  - 90 digital theaters in U.S.
- Ongoing problems
  - No standardization (codec, file formats, etc.)
  - High per-theater cost ($100,000)
  - Who will pay for retrofit? Upgrades?

CRYPTOGRAPHY RESEARCH

The Project Begins…

Digital Cinema Case Study

- Digi-Flicks approaches CRI to solve perceived barrier to DC adoption
  - Security concerns holding back deployment of digital cinema
  - Existing equipment manufacturers not focused on security
- Design goals for prototype system
  - Transport-independence for movie
  - Strong crypto
  - Multi-factor authentication
  - Flexible authorization policy
  - Reliable playback even with communication failures
  - Rapid development (4 months)

Movie Transport Analysis

- Network
  - DSL: low cost (~14 hours)
  - OC3: high cost per theater (~2 hours)
  - Satellite: high cost but amortized (~5 hours)
- Physical (24 hours)
  - DVD: 4.5 GB per disc
  - Hard drive or tape: 150 GB per drive
- Less than 300 theaters, shipping hard drives most cost effective
- Otherwise, use satellite

Content Security Analysis

- Threat model for projection booth
  - Physical enclosure can be bulky
  - Cost not a huge issue
  - Limited access by projectionist
- Compare to consumer electronics
  - Must be small, light, cheap
  - Unlimited access by user
- Simple prototype design
  - Encryption substitutes problem of protecting huge movie file with protecting small key
  - Derive key through multi-factor scheme with online approval
  - Physical security left to later product design effort

CRYPTOGRAPHY RESEARCH

Theater Server Analysis

- Components
  - Large case with many custom boards
  - Loaded with custom ASICs
  - 33 Mhz PowerPC
  - 64 MB ram
  - UNIX-like OS
  - 8 hard drives (4 drive stripe, mirrored)
- Developer info
  - No documentation available
  - No APIs
  - Expected to use vendor-provided tools

CRYPTOGRAPHY RESEARCH

Bulk Data Interface

- Connectors available for I/O
  - Ethernet
  - SCSI
  - Serial
    - Too slow
  - Analog/digital video in
    - Too complex, no compatible hardware
- Internal access would require case mods

CRYPTOGRAPHY RESEARCH

Control Interface

- How do you hook into the playback process?
- Install code on the theater server
  - No way to hook into playback path (API)
  - Not enough available CPU
- Copy plaintext movie onto theater server
  - Unknown filesystem format
- Serial interface allows limited command line control

Ethernet Interface

- First candidate for data interface
- Initial evidence: good
  - 10/100 port = ~10 MB/s = 1 hr. per movie
  - TCP/IP support
- Further inspection: bad
  - Slow transfers (1 MB/s)
  - Proprietary Windows tools (no FTP)

CRYPTOGRAPHY RESEARCH

SCSI Interface

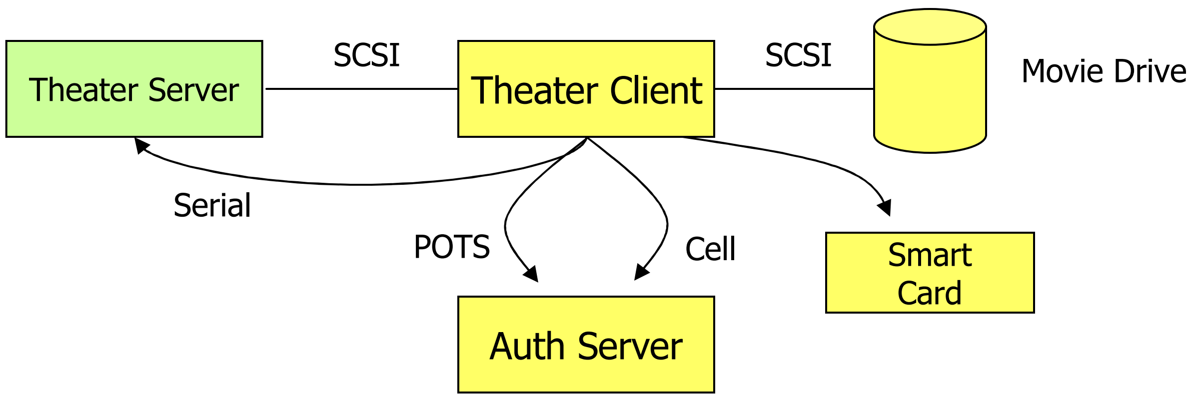
- Initial evidence: good
  - Ultra2W = 80 MB/s = 7 minutes per movie
  - Faster than real-time transfers
  - External connectors so no case mod needed
- Further inspection: bad but salvageable
  - Four independent channels apparent solution to lack of software concurrency
  - Single drive accesses occur at 1/4th the total rate
  - Workaround: extra read-ahead to make up the difference

CRYPTOGRAPHY RESEARCH

Authentication

- Multi-factor authentication requires all participants to be present to derive a valid key
  - Hard drive containing the movie
  - Smart card
  - Online exchange with auth server
- Flexible policy
  - Use flat file of allowed theater client, card, movie tuples
  - Allows auth server to implement more complex policy separately
- Result: custom protocol to achieve this with a minimum of round trips

Theater Client Prototype

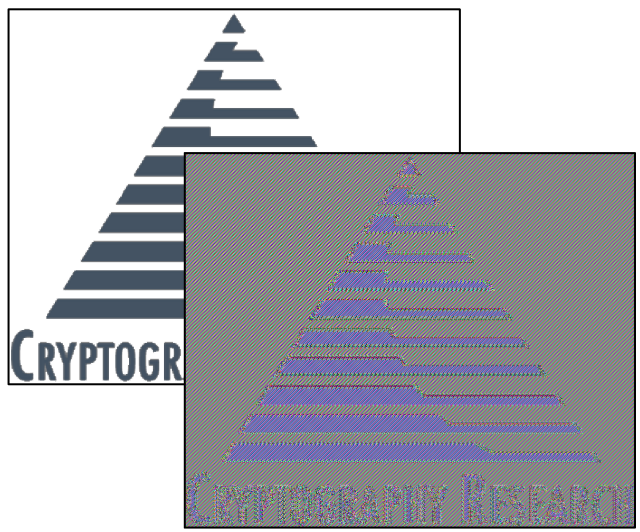


- Transparently encrypt/decrypt block data in real-time
- Store movie onto drive the same way
  - Format new volume through theater client
  - Copy movie onto new volume
- Dialup auth step with cellphone backup

IT Storage Security

Good Crypto, Poor Design

- With crypto, the details of a design really matter
- Recently-introduced commercial disk encryption product used 3DES ECB
  - Strong cipher, inappropriate mode of operation



3DES in ECB mode

Storage Crypto Products

- Two main camps
  - Filesystem (CFS, TCFS, EFS)
    - Encrypt file contents and name
    - Don't encrypt metadata (size, attributes, etc.)
  - Block (PGPdisk, BestCrypt, GEOM, CGD, LoopAES)
    - Encrypt block data below filesystem layer
    - Incompatible with FS tools (backup, volume management)
- All have similar approaches
  - Cipher strength and key length main focus
  - Block storage: try to avoid data expansion
  - No integrity protection
    - Chaining (CBC, CFB) hides similarities in the plaintext
    - Does *not* prevent modification

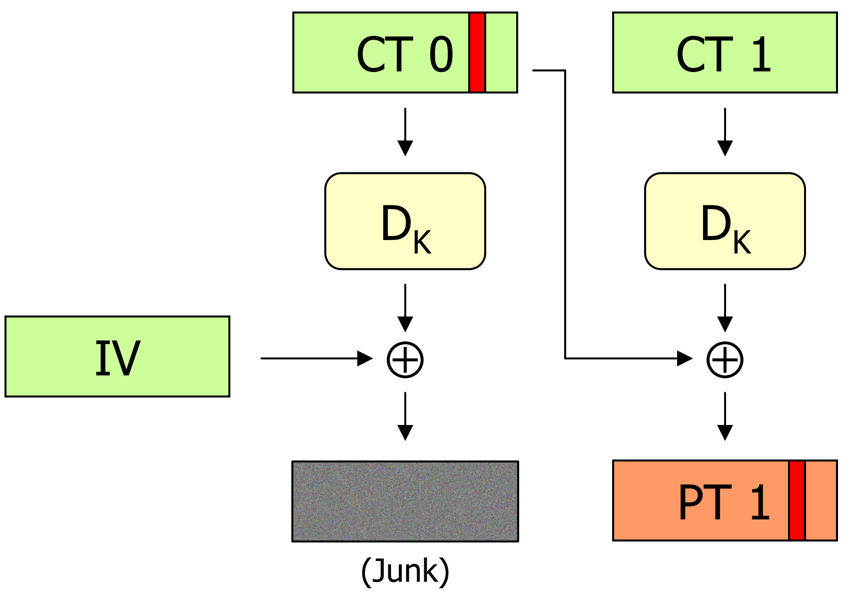
Threat Models

- Design stage: provides clear security requirements
- Deployment: usage expectations clearly dictated
- Storage threat models (increasing leverage)
  - Attacker has one-time read-only access to ciphertext
  - Attacker has repeated read-only access to ciphertext
  - Attacker has one-time read-write access to ciphertext
  - Attacker has repeated read-write access to ciphertext
- Most storage crypto products only anticipate the first threat model
  - Other threats becoming more common
  - Example: warm spare linked to outsourced storage company via SAN

CRYPTOGRAPHY RESEARCH

CBC and Integrity Protection

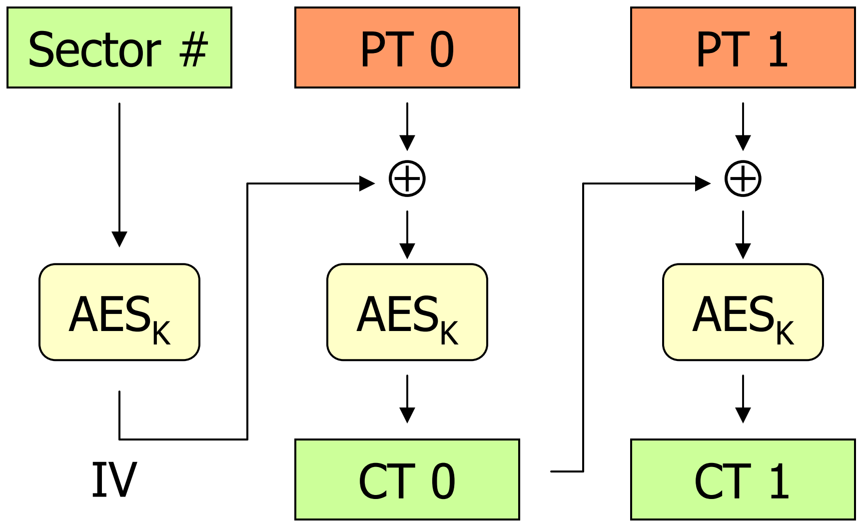
- CBC does *not* provide integrity protection
  - Bit(s) flipped in CT_{N-1} results in bit(s) flipped in PT_N
  - Also completely garbles PT_{N-1}
- Changes to the IV allow bit flips with no garbling



Example: CGD

Description[1]

- Block encryption for NetBSD disks
  - Creates a virtual partition and encrypts/decrypts data
  - Single key passed in via user program
- Encryption: CBC chaining with Enc.(sector #) as IV



1. R. Dowdeswell, J. Ioannidis; "The Cryptographic Disk Driver"; USENIX 2003 FREENIX track

Example: CGD

Threat Model Analysis

- Threat Model: one-time read-only access
  - Privacy maintained, assuming key was managed properly
- Threat Model: repeated read-only access
  - Key is constant per volume ⇨ IV constant per-sector
  - Same data written to a sector gives same ciphertext
- Threat model: one-time read-write access
  - Identify important block and modify it
  - Examples
    - Modify password file on encrypted disk to allow an attacker access to the system
    - Move sector location, causing new IV to be XORd into contents
- Threat model: repeated read-write access
  - Turns above into an adaptive attack

Solving Integrity Threats

- Add a message authentication code (MAC)
  - Cryptographically-strong integrity check (e.g., SHA-1 HMAC)
  - Some performance hit
- Threat model: attacker only has offline (cold) access
  - Performance enhancement possible
    - Lazily update MAC on writes (along with write cache)
    - Check MAC on reads, mark sector as good in bitmap
- Lesson: be sure you know your threat model

CRYPTOGRAPHY RESEARCH