Attack of the Clones: Building Clone-Resistant Products

Benjamin Jun
Cryptography Research, Inc.

Feb 14, 2006 – FS-102

CRYPTOGRAPHY RESEARCH

RSA CONFERENCE 2006

Overview

- "Clone (klōn): One that copies or closely resembles another, as in appearance or function" *– American Heritage Dictionary*

- Examples:
  - Knock-off batteries, printer consumables
  - Pirated software, DVDs
  - Software unlock codes

"Dolly", TIME 3/10/1997

The American Heritage® Dictionary of the English Language: Fourth Edition

Agenda

- Cloning & Related Attacks
- Tools and Approaches to Resist Cloning
- Responding When Problems Occur
- Case Studies

Hope it doesn't come to this...

- Our focus: Tamper resistance in embedded systems
  - Learn from high-threat (and highly constrained) systems
  - Many lessons can be applied to pure SW environments

Who am I? What do I do?

- Primary business:
  - Develop & license new security technologies
  - Provide design and evaluation services
  - Major R&D focus on solving real-world security problems
- Industries served:
  - Financial
  - Entertainment / Pay TV
  - Tamper resistance
  - Wireless / Telecommunications
  - Internet

CRYPTOGRAPHY RESEARCH

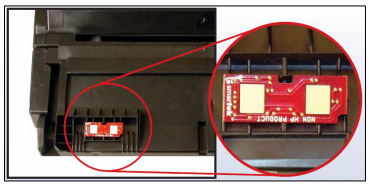
Products incorporating CRI technology secure over $100 billion annually

The Threat Model: Cloning & Related Attacks

Cloning and Related Security Problems

- Consumable cloning
  - Cellphone batteries
  - Remanufactured toner cartridges
- Software unlocking
  - Software activation codes
  - Feature unlocking
- Digital content
  - Pay TV card emulators
  - Pirated DVD's
- Identity cloning / network access
  - Vehicle key duplication
  - Cellphone ID cloning

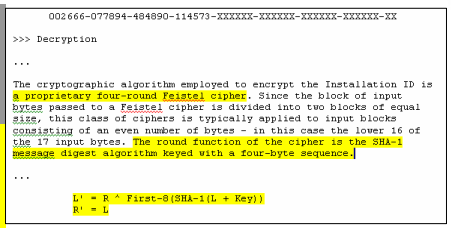
"RAZR V3" Battery


Toner Cartridge Chip

"Coolwireless" store: http://www.scc-inc.com/Engine/EngineDocs/HP42/MaxCoreSolution.pdf

Cloning and Related Security Problems

- Consumable cloning
  - Cellphone batteries
  - Remanufactured toner cartridges
- Software unlocking
  - Software activation codes
  - Feature unlocking
- Digital content
  - Pay TV card emulators
  - Pirated DVD's
- Identity cloning / network access
  - Vehicle key duplication
  - Cellphone ID cloning



002666-077894-484890-114573-XXXXXX-XXXXXX-XXXXXX-XXXXXX-XX

>>> Decryption

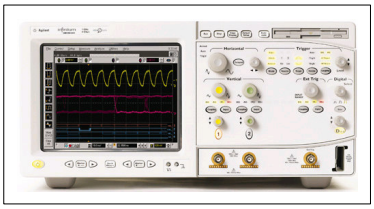
...

The cryptographic algorithm employed to encrypt the Installation ID is a proprietary four-round Feistel cipher. Since the block of input bytes passed to a Feistel cipher is divided into two blocks of equal size, this class of ciphers is typically applied to input blocks consisting of an even number of bytes - in this case the lower 16 of the 17 input bytes. The round function of the cipher is the SHA-1 message digest algorithm keyed with a four-byte sequence.

...

L' = R ^ First-8(SHA-1(L + Key))
R' = L

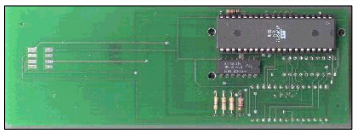
Inside Windows [XP] Product Activation
Licenturion (July 2001)



Agilent 54833D Oscilloscope

Cloning and Related Security Problems

- Consumable cloning
  - Cellphone batteries
  - Remanufactured toner cartridges
- Software unlocking
  - Software activation codes
  - Feature unlocking
- Digital content
  - Pay TV card emulators
  - Pirated DVD's
- Identity cloning / network access
  - Vehicle key duplication
  - Cellphone ID cloning

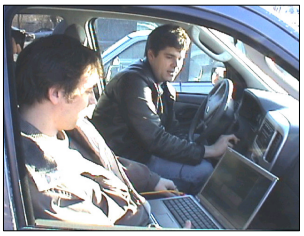


Smartcard Emulator

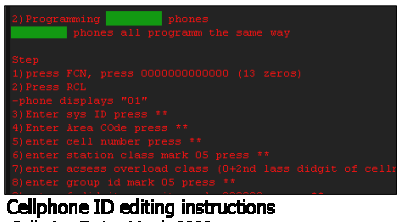
Pirated DVDs, San Francisco

Cloning and Related Security Problems

- Consumable cloning
  - Cellphone batteries
  - Remanufactured toner cartridges
- Software unlocking
  - Software activation codes
  - Feature unlocking
- Digital content
  - Pay TV card emulators
  - Pirated DVD's
- Identity cloning / network access
  - Vehicle key duplication
  - Cellphone ID cloning



Starting car using a DST simulator



Cellphone ID editing instructions
Coflusion E-zine, March 2002

S. Bono, M. Green, A. Stubblefield, A. Rubin, Johns Hopkins University

Meet Your Attacker

- "Joyriders"
  - Can enable for-profit attackers
- Profit-driven attackers: Cost / Benefit
  - NRE to invest if past business successful!
  - "Street" offers premium if cloning difficult

www.counterfeitmini.org

- Goal: Find repeatable attack
  - **Reliable:** *Find break that works 100% of the time*
  - **Cheap:** *Incremental attack cost low*

- Your best defense: ***Make cloning unprofitable***
  - Deter commercial attacks
  - Minimal hassle for legitimate users

Anti-Cloning Tools

Example Scenario

- Validator inspects consumable

Consumable **Validator**

?

- Threats:
  - Interoperable consumable
  - Remanufactured consumable

Perrincraft; Weusa.l Paper OptiCore™

Legal Notices / Proprietary Protocol

" PROTOCOL (C)2006 PCORP "

- Add copyrightable data / patentable elements to protocol
  - Proprietary protocol
  - Verifier checks for presence of these notices
- Provides non-technical protection
  - Provide unambiguous notification and (possible) legal protection
  - Can one rely on NDAs and security through obscurity?

Legal Notices / Proprietary Protocol

" PROTOCOL (C)2006 PCORP "

- Patent protection / patents on circumvention technologies
- Copyright law* and the DMCA
  - 17 U.S.C. grants copyright protection for original expression
  - "Manufacturers of interoperable devices such as computers and software, game consoles and video games, printers and toner cartridges, … may employ a security system to bar the use of unauthorized components… To the extent compatibility requires… a particular code sequence… the code sequence [is precluded] from obtaining copyright." **

* Note: Seek legal advice before relying on this information; ** US Court of Appeals for 6th Circuit, # 03-5400

Challenge-Response

DeviceID, dev_key

chal

Device ID, HMAC$_{dev_key}$ (chal)

F(DeviceID, Master) → dev_key

- Verifier sends an unpredictable challenge
  - Inspects response
- Challenge: Response should be "hard" to clone
  - Verifier needs unpredictable (random) challenges
  - Keys must be protected from exposure
    - Crypto + protocol design
    - Tamper resistance
  - Another TR option: make response calculation "difficult"…

IBM Thinkpad (does not represent actual protocol)

Challenge-Response

DeviceID, dev_key

chal

Device ID, HMAC$_{dev_key}$ (chal)

F(DeviceID, Master) → dev_key

- Design choices
  - ***Public vs. Secret key*** – security / computational trade-offs
  - ***Global vs. Individual keys*** – manufacturing trade-offs
  - Caution: Many details!
- Key management and revocation
  - How to know which keys have been compromised?
  - How to distribute revocation information?

IBM Thinkpad (does not represent actual protocol)

Signed Device Information

certificate, signature

certificate, $\text{Sign}_{\text{sys\_key}}$ (certificate)

chal

$\text{Sign}_{\text{dev\_key}}$ (chal)

Root public key, Revocation list

- Devices personalized at manufacture
  - Unique device information digitally signed (signature / MAC)
- Validation steps:
  - Verify certificate signature
    - Check that ID info is unmodified
  - Check ID info against revocation list
    - Check that ID is not revoked
  - Challenge-response to check binding of keys / cert
    - Prevent simple replay

IBM Thinkpad (does not represent actual protocol)

Signed Device Information

certificate, signature

certificate, $\text{Sign}_{\text{sys\_key}}$ (certificate)

chal

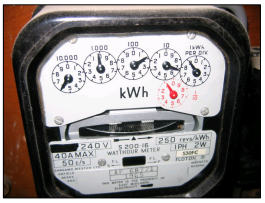
$\text{Sign}_{\text{dev\_key}}$ (chal)

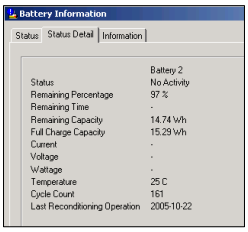
Root public key, Revocation list

- Put attackers on defensive
  - Revoke known bad ID's as they become known
  - Force attackers to repeat certificate + key extraction
- Design choices
  - Certificate structure
  - Public key (signature) vs. secret key (MAC)

IBM Thinkpad (does not represent actual protocol)

Device Self-Tracks History





- Device tracks history in NVRAM
  - Months active, % used, last validator ID, expiration date, …
  - Suspend operation if parameters out of range
- Protect history fields
  - Windows registry is a bad place to store protected state!
  - Memory clearing attacks (UV light), rollback, …

Requirement: Implementation Correctness

- System design must be properly implemented
- Biggest problem: Software Bugs!
  - Buffer overruns, software update mechanism, non-atomic memory writes, unnecessary state duplication, …
  - Incremental attack cost = $0
- Why it's hard
  - Many specs written with security ambiguities
  - Embedded systems lack partitions

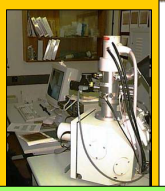
- Goal: Design for validation
  - Create system with human validators in mind
  - Manage design complexity

Requirement: Closed Computation

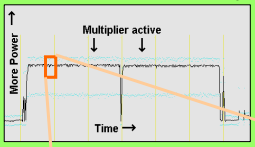
- Secrets must stay within device
- Invasive attacks
  - Probing: circuit boards, data bus
  - Chip decap and image/probe
  - Alter security fuses
- Information leakage attacks
  - Differential Power Analysis (DPA), SPA, Timing…
- Manufacturing test modes
  - Scan, JTAG

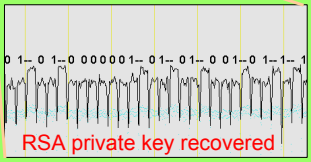
- Goal: Drive them to an invasive attack!
  - Force each incremental attack to be expensive, invasive

Focused Ion Beam



Simple Power Analysis (SPA)





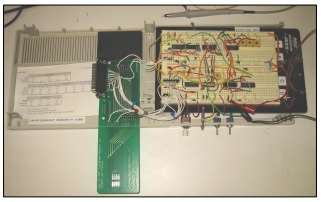
RSA private key recovered

University of Cambridge Department of Materials Science Device Materials Group; Cryptography Research, Inc.

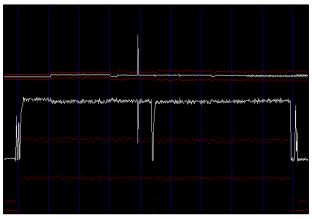
Requirement: Reliable Computation

- Computation must be reliable (correct)
- Devices can be forced to operate "non-deterministically"
  - Alter Vcc, gnd, clock, EM, temperature
  - Result: bad memory reads/writes, comparisons, jumps, ALU errors, …
- Attack process
  - Find glitch that yields unpredictable behavior
  - Study effect and make repeatable

- Goal: Be paranoid about computation
  - Use sensors, canary logic
  - Perform redundant computations



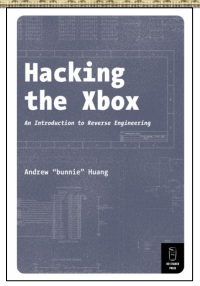
CRI glitching circuit for smart card testing



Glitch impulse & power trace during successfully-glitched RSA CRT

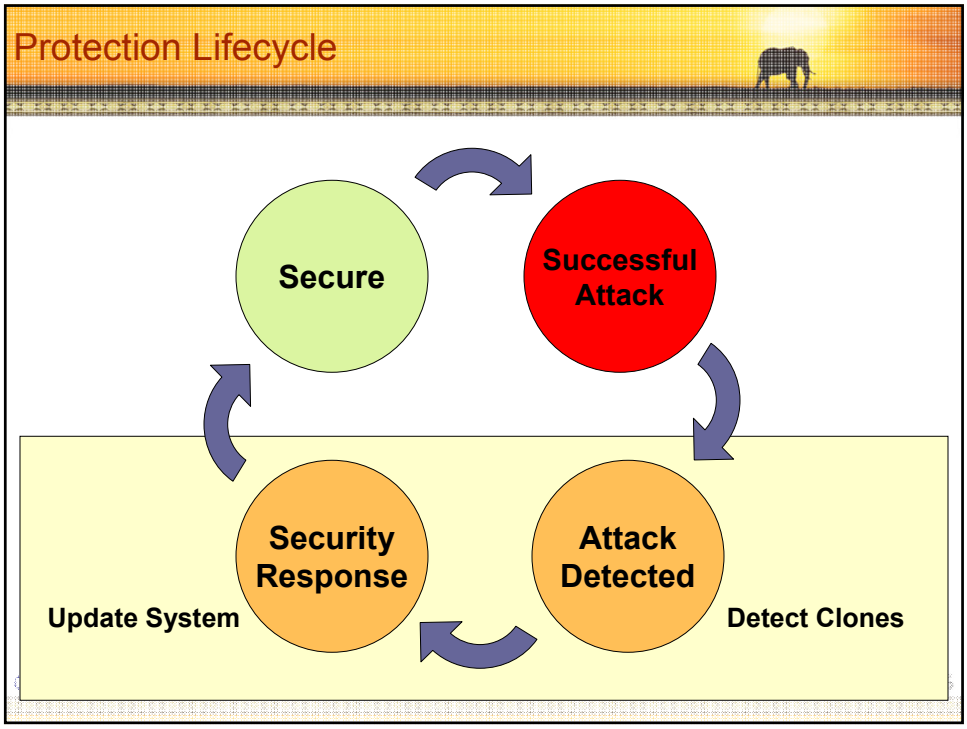
Protect the Validator!

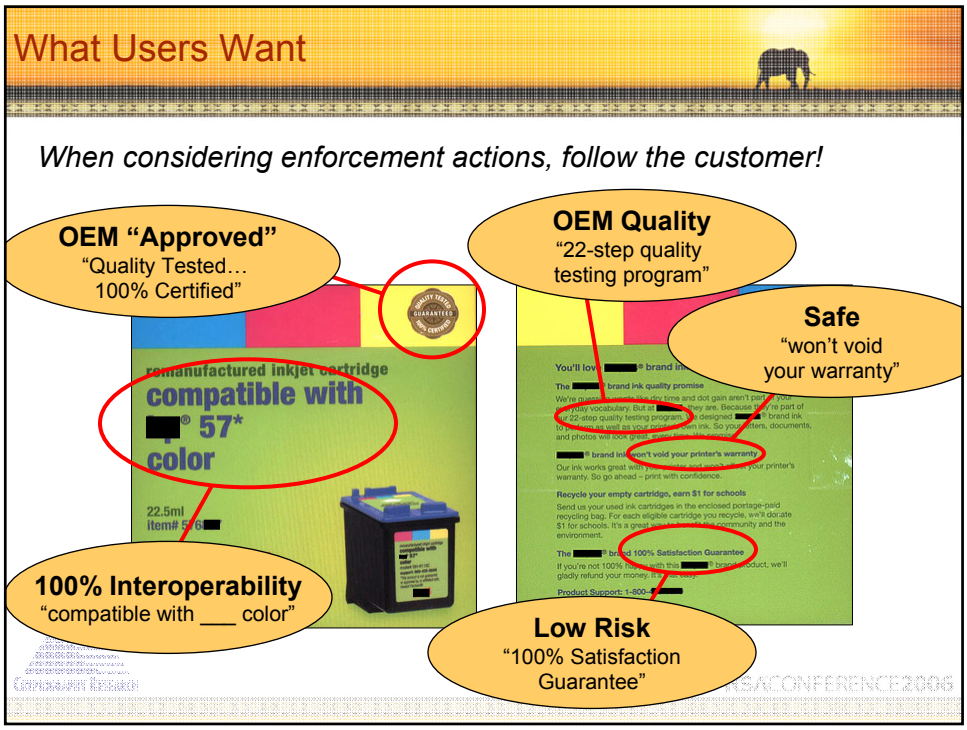
- Extract authentication keys
  - Issue for symmetric-key systems
- Bypass authentication process
  - Patch verification code
  - Change verification keys (browser root key attack)
- Selective amnesia
  - "Forget" revocation data, usage history, date
- "Bootstrap" with a valid device
  - Exploit bugs after validation has completed
- Many more attacks possible…



Cartridge: Wikipedia

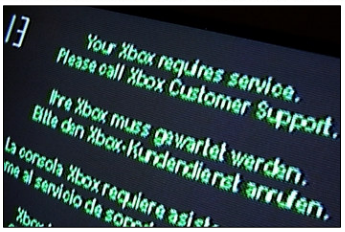
When Bad Things Happen to Good Systems…


Protection Lifecycle
Secure
Successful Attack
Security Response
Attack Detected
Update System
Detect Clones


What Users Want
When considering enforcement actions, follow the customer!
OEM "Approved"
"Quality Tested… 100% Certified"
OEM Quality
"22-step quality testing program"
Safe
"won't void your warranty"
compatible with 57* color
100% Interoperability
"compatible with ___ color"
Low Risk
"100% Satisfaction Guarantee"

Clone detected! Now what?

- Reject device outright
  - Immediate shutdown
  - Works if user understands link to clone usage
- Inform user
  - Notify user that device is unapproved
  - Threaten to void warranty
- Perform in degraded mode
  - Run slower, "calibrate" more often
  - Slower boot-up with error message
- Phone home
  - Use network connection, request user call 1-800 #



www.command-tab.com

Online Connectivity

- Effective responses require communication with in-field verifiers
  - Deliver new authorizations, revocation data
  - Update security code, verification process
- If device regularly online, this is easy…
  - Can require online activation, updates, audits
  - Peripherals: OS drivers can go online
- Online activation
  - Phone call (network access)
  - Unlock code function of device serial #, verifier information
  - Option: Activate at retailer or distributor

TiVo Rear Panel

Microsoft Telephone Activation

Microsoft Corporation

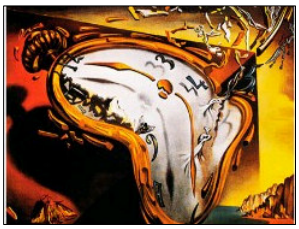
Update Channels

- Creative connectivity
  - Broadcast-only
  - Occasional connectivity
  - Retail-purchased consumables
  - Security module swap-out
- A spoonful of sugar helps the medicine go down…
  - Feature updates
  - Digital content
- Consider other channels
  - In-band signaling (MPEG “null” packets)
  - “Mesh / Viral” updates

- Concern: Response may not be uniform
  - Even 15% revocation can spoil attacker business model

Time… The Final Frontier

- Device timekeeping can help
  - Enforce usage rules, device expiration, content access policies
  - Challenging to do securely!
    - Battery operated clock = $$
    - Time adjustment (accidental / intentional)



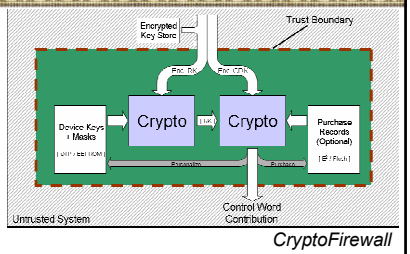
- Example: “Broadcast” time updates
  - Vendor signs Julian timestamps
    “YYDDD, $sign_k$(YYDDD)”
  - All devices store latest signed timestamp
  - In every handshake, signed date packets are exchanged
  - Replace timestamp if packet (a) has valid signature, and (b) is more recent

Explosion, Salvidor Dali

Case Studies

Example: CryptoFirewall™

- Problem: Pay TV Piracy
  - Attacks: Re-keying, software bug exploit, hardware modification, PC emulation, …
  - Lost revenue per pirate device >$3000
  - Pirate “research” NRE >$500K



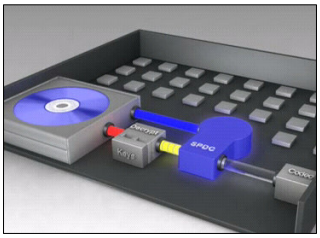
CryptoFirewall

- Strategy: Maximize TR with conventional silicon
  - Goal: Require incremental attacks to use invasive methods

- Add independent ASIC that contributes to session key
  - Crypto enforcement of viewing policies
  - Critical keys never leave security boundary
  - High entropy netlist frustrates reverse engineering, emulation
  - No scan / debug modes, canary circuitry, …

Example: Self-Protecting Digital Content™

- Problem: Optical media content theft
  - Attacks: Content ripping, P2P, disc duplication, DeCSS, players with known bugs, …
  - Formats have 20+ year lifespan
- Strategy: Active response
  - Add interpreter to playback devices
  - Disc-based content code enables playback



Self-Protecting Digital Content

- ***Detect piracy:*** Forensic Marking
  - Content code renders content differently on each playback
  - Player-specific information embedded in content
- ***Respond to piracy:*** Renewability
  - Content code queries player state, makes playback decisions
  - Content code delivers RAM-based player updates

Conclusion

- Cloning is a huge and growing problem
  - Razor / razorblade model extending to more products
  - Product value in software features, system interaction
  - Cloning attempts to take advantage of this value
- Substantial challenges
  - Tamper resistance on low-cost platforms
  - System recovery despite periodic connectivity
- ***Design to make cloning unprofitable***