601.445/601.645
Practical Cryptographic Systems

Introduction

Instructor: Matthew Green

Intro

- What is a Cryptographic System?
  - A security system
  - Uses cryptography
    - Many fascinating ways to get it wrong!
  - “Practical”: People actually use it & depend on it

DVD-Cracking Teen Acquitted

Associated Press 01.07.03

AACS encryption key controversy - Wikipedia, the free

... represented in hexadecimal as 09 F9 11 02 9D 74 E3 5B D8 41

BIZ & IT —

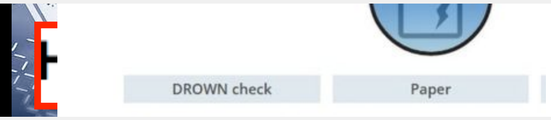
Serious Android crypto key theft vulnerability affects 10% of devices

Bug in Android KeyStore that leaks credentials fixed only in KitKat.

DAN GOODIN - 6/28/2014, 1:00 PM

SPI SPYING —

Trusted platform module security defeated in 30 minutes, no soldering required



Microsoft fixes Windows crypto b the NSA

of the Microsoft's January 2020 Patch Tuesday.

CRITICAL CONDITION —

Hospitals hamstrung by ransomware are turning away patients

Motivation

- Building (successful) systems requires more than cryptographic expertise
  - Though it’s a prerequisite!
- It’s cross-disciplinary:
  - Crypto
  - Information Security
  - Software Engineering
  - Hardware Engineering
  - UI, Policy, etc...

This course

- Not a theoretical course in cryptography
  - We'll cover the basics,
    but aim is to apply cryptography
- Practice-oriented tutorial
  - examine how systems fail
  - how we can design against it
  - what can't we design against
- Driven by your questions & the news

What you'll come away with

- A grounding in cryptographic techniques
  - The right algorithms
  - Strengths & weaknesses, applicability
  - A feel for the design/evaluation process
  - Introduction to standards (e.g., FIPS)
  - Enough to know where to look for more
- Knowledge of our own limitations
  - Building secure systems is hard
    (even for experts)

Grading, Text

- Grading Policy:
  - Written homework assignments (10% of grade, see grading notes below)
  - Programming assignments (40% of grade)
  - Midterm & final exams (20% of grade, combined)
  - Class project (20% of grade)
  - Class participation (10% of grade)
- Texts:
  - Boneh & Shoup: A graduate course
  - Anderson: Security Engineering

Syllabus / Piazza / BB / GS

- https://github.com/matthewdgreen/practicalcrypto/wiki/Fall-2021-Syllabus

- https://piazza.com/class/krxtqrzmixc5xq
(or go to Piazza, search for class name/number, Fall 2021)

- We'll use Blackboard and/or Gradescope to submit assignments (and nothing else!)

Course Schedule

8/30: Introduction

- Weekly written assignment 1 handed out, due Monday 2/3 11:59pm
- Reading: Anderson (Security Engineering), Chapter entitled Cryptography: Sections on Symmetric Crypto Primitives (5.4 in online version)

9/1: Intro to Cryptographic Primitives I (Symmetric key crypto)

- Reading (supplementary, optional): Handbook of Applied Cryptography, chapter entitled "Block Ciphers".
- Programming assignment 1 is out

9/8: Intro to Cryptographic Primitives II (Symmetric key crypto continued)

- Reading: Anderson (Security Engineering), Chapter entitled Cryptography: Asymmetric Crypto Primitives (5.7 in online version).
- Reading (supplementary, optional): Handbook of Applied Cryptography, chapter entitled "Public Key Encryption".
- Weekly Assignment #2 is out, due in one week

Programming

- The assignments in this class involve writing code
  - We will primarily use Go
  - It’s your responsibility to give us working assignments that build/run
  - Anything other than a working assignment is a failure

Project

- This is a project class
  - We will provide a list of project topics
  - These can be done in small groups
  - We will discuss more about this on Weds

Course Guidelines

- Do:
  - Read the news!
    Twitter, HN, ArsTechnica, etc.
  - Bring up interesting topics & recent attacks you'd like to learn more about
- Don't:
  - Cheat***
  - Get me arrested

Course Guidelines

Re: hacking attempt (Matt Green, please read...) Inbox ×

Steve Rifkin <steve410@cs.jhu.edu>

- Bring up interesting topics & recent attacks you'd like to learn more about

- Don't:
  - Cheat***
  - Get me arrested

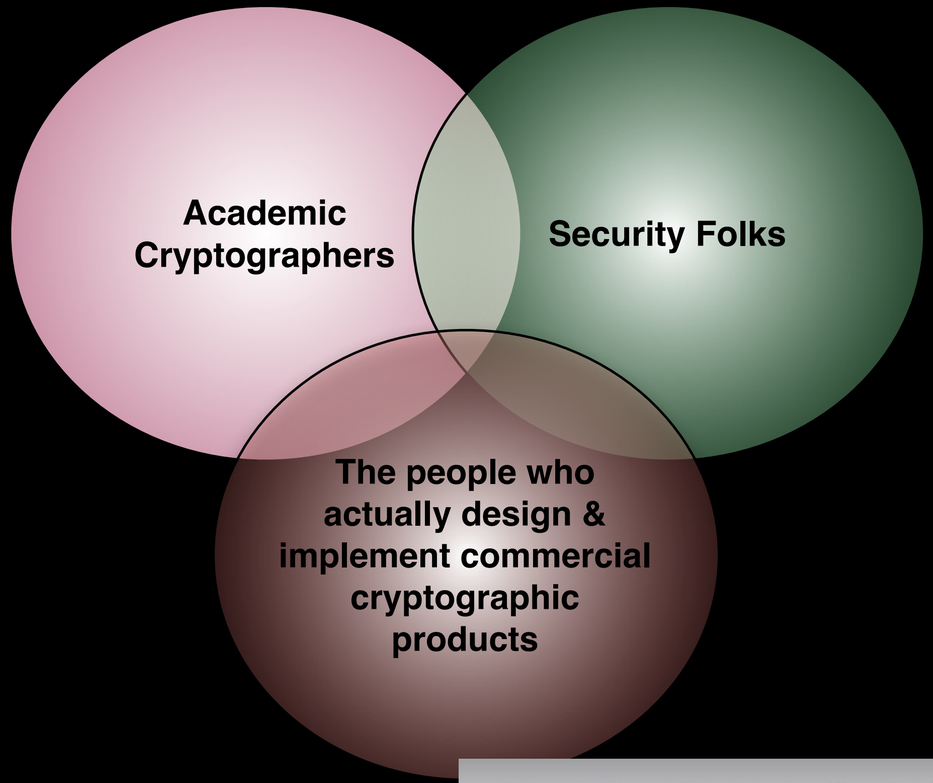
Readings & Assignment

- Assigned each week
  - You must read them, be prepared to discuss in class
  - These wil be covered in written assignments on the homework, as well as exams

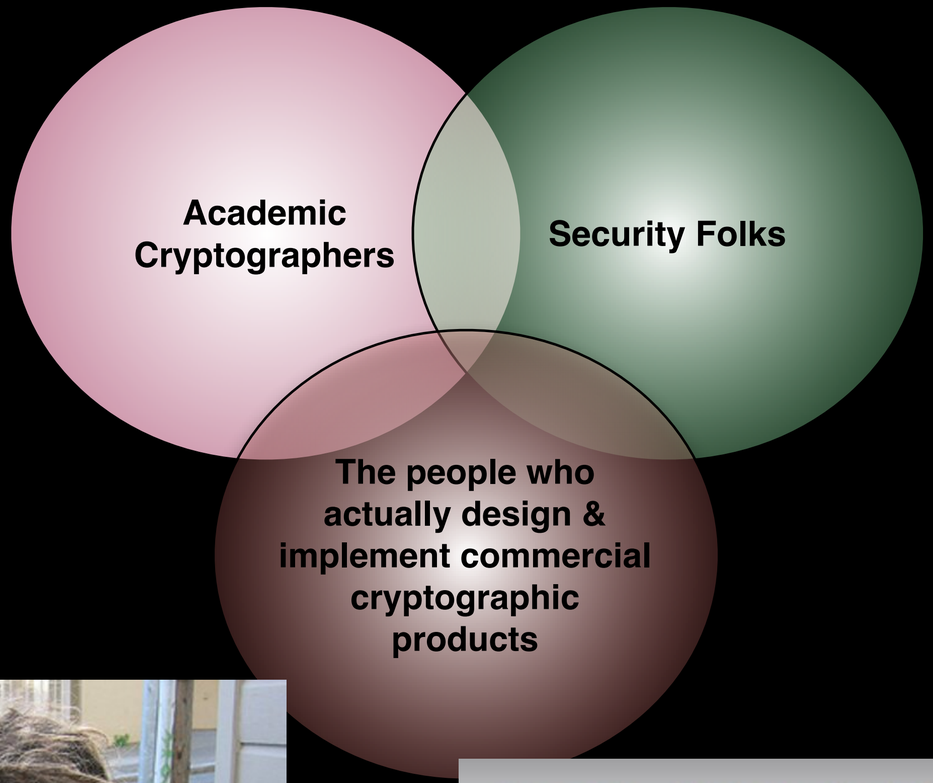
Instruction

- Co-teaching with Alishah Chator
  - Alishah is a senior PhD student in my lab
  - We will rotate classrooms, content will be divided between us
  - Working on TA/CA details right now
  - We may combine classes in the future, depending on COVID and convenience

Today


Academic
Cryptographers
Security Folks
The people who
actually design &
implement commercial
cryptographic
products

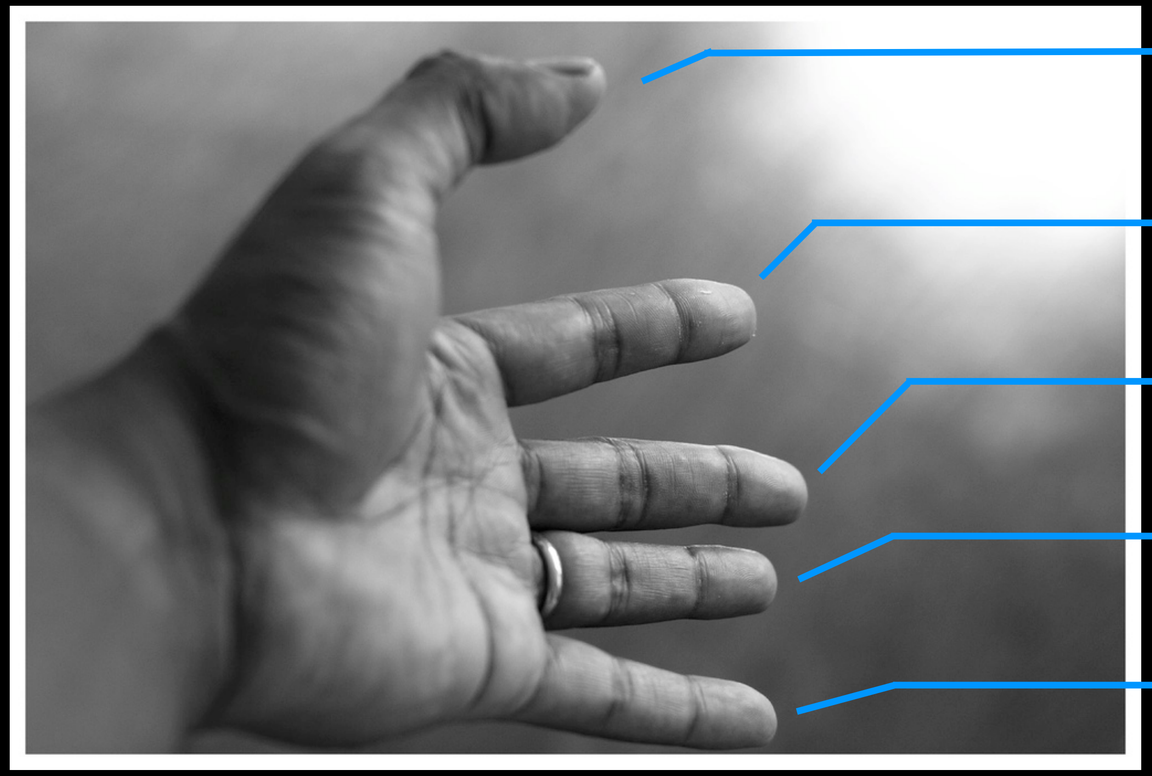

Simple, Unbreakable Encryption.


Academic
Cryptographers
Security Folks
The people who
actually design &
implement commercial
cryptographic
products


Simple, Unbreakable Encryption.

Security Failure

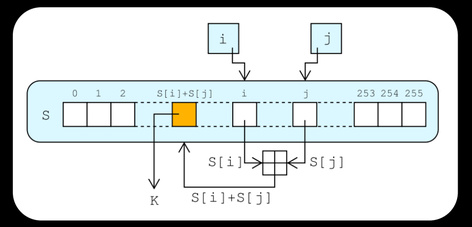
- When security systems fail:
  - Researchers get published
  - $$$ lost
  - Private information compromised
  - People die (?)

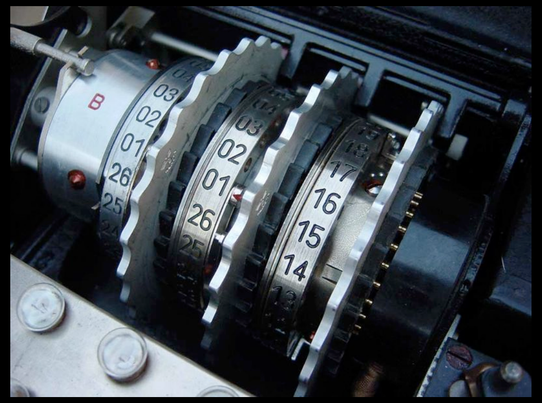


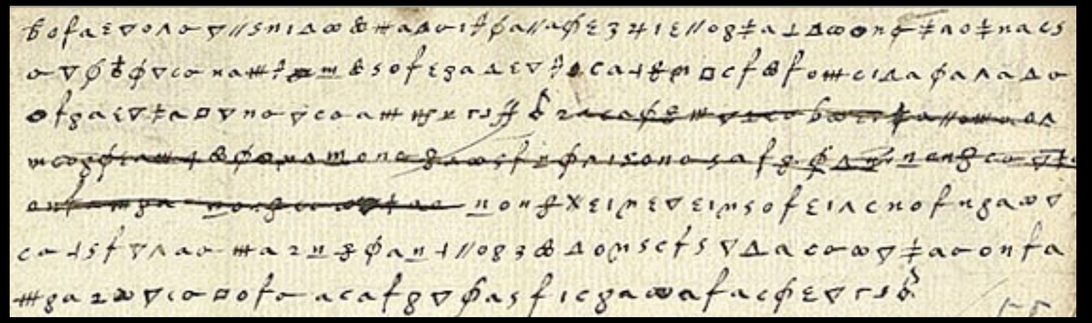
Primitives

Primitives

- Codes, ciphers, encryption schemes, MACs, etc.:
  - Classically, an attack on the “system” meant an attack on the primitive
  - History is littered with broken primitives



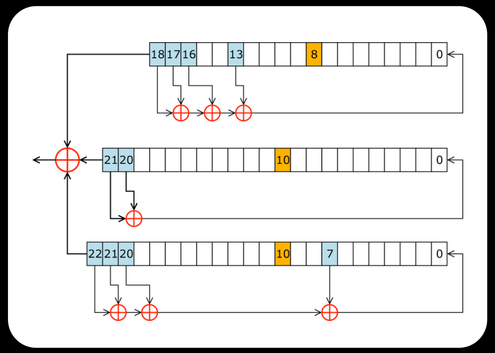




Primitives

- Practical Example: GSM encryption
  - A5/0: No encryption
  - A5/1: Based on LFSRs
  - A5/2: Weakened A5/1
  - A5/3 (KASUMI): New for 3G

GSM™
The wireless evolution



T-Mobile

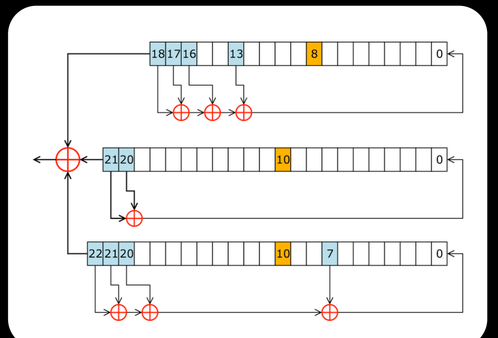
at&t

Primitives

- Practical Example: GSM encryption
  - A5/0: No encryption
  - A5/1: Broken
  - A5/2: Way Broken
  - A5/3 (KASUMI): Dented
    (and 3G vuln. to protocol attacks)
- Deliberately weak cipher design
  - Cost & politics

GSM™
The wireless evolution

T · · Mobile



at&t

Primitives

- Practical Example: GSM encryption
  - A5/0: No encryption
  - A5/1: Broken
  - A5/2: Way Broken
  - A5/3 (KASUMI): Dented
    (and 3G vuln. to protocol attacks)
- Deliberately weak cipher design
  - Cost & politics

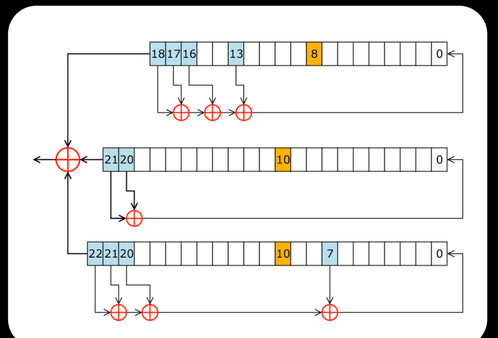
The New York Times

Cellphone Encryption Code Is Divulged

By KEVIN J. O'BRIEN
Published: December 28, 2009

BERLIN — A German computer engineer said Monday that he had deciphered and published the secret code used to encrypt most of the world's digital mobile phone calls, saying it was his attempt to expose weaknesses in the security of global wireless systems.

T-Mobile



at&t

Primitives

- Practical Example: GSM encryption
  - A5/0: No encryption
  - A5/1: Broken
  - A5/2: Way Broken
  - A5/3 (KASUMI): Dented
    (and 3G vuln. to protocol attacks)
- Deliberately weak cipher design
  - Cost & politics

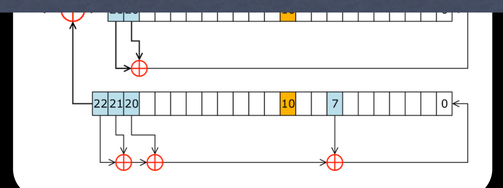
threatpost
The Kaspersky Lab Security News Service

January 11, 2010, 4:57PM

A Second GSM Cipher Falls

A group of cryptographers has developed a new attack that has broken Kasumi, the encryption algorithm used to secure traffic on 3G GSM wireless networks. The technique enables them to recover a full key by using a tactic known as a related-hey attack, but experts say it is not the end of the world for Kasumi.

T-Mobile



at&t

Primitives

- Practical Example: GSM encryption
  - A5/0: No encryption
  - A5/1: Broken
  - A5/2: Way Broken
  - A5/3 (KASUMI): Dented
    (and 3G vuln. to protocol attacks)
- Deliberately weak cipher design
  - Cost & politics

GSM™
The wireless evolution

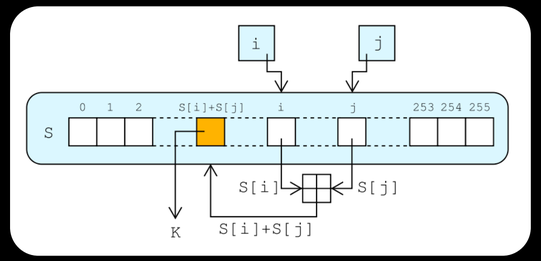
T-Mobil

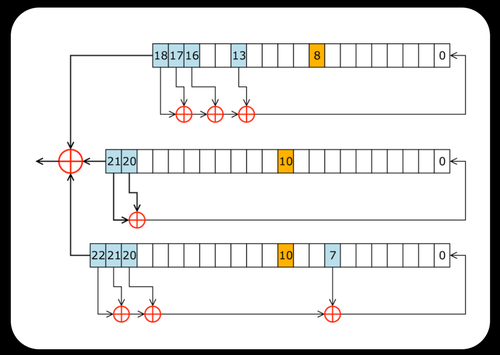
The vulnerability in the GEA-1 algorithm is unlikely to have been an accident, the researchers said. Instead, it was probably created intentionally to provide law enforcement agencies with a "backdoor" and comply with laws restricting the export of strong encryption tools.

"According to our experimental analysis, having six correct numbers in the German lottery twice in a row is about as likely as having these properties of the key occur by chance," Christof Beierle of the Ruhr University Bochum in Germany, a co-author of the paper, said.

Primitives

- Typical problems:
  - Using the wrong ones (& homebrew crypto)
  - Or using the right ones... wrong
- E.g., RC4 in WEP





Virtual Matrix Encryption (VME) is a data security method and apparatus that provides an exceptional degree of security at low computational cost. The data security arrangement differs from known data security measures in several fundamental aspects. Most notably, the content of the message is not sent with the encrypted data. Rather, the encrypted data consists of pointers to locations within a virtual matrix, a large (arbitrarily large), continuously-changing array of values.

Primitives

- Sometimes the “right” primitives stop being right...
  - The great Hash Function Adventure of 200X (MD5 broken, SHA1 broken, …)

Google Security Blog

The latest news and insights from Google on security and safety on the Internet

Announcing the first SHA1 collision

February 23, 2017

Posted by Marc Stevens (CWI Amsterdam), Elie Bursztein (Google), Pierre Karpman (CWI Amsterdam), Ange Albertini (Google), Yarik Markov (Google), Alex Petit Bianco (Google), Clement Baisse (Google)

Cryptographic hash functions like SHA-1 are a cryptographer’s swiss army knife. You’ll

MD5 considered harmful today

Creating a rogue CA certificate

Primitives

- Sometimes the “right” primitives stop being right…

July 30, 2021

Preparing for the Post-Quantum Migration: A Race to Save the Internet

LinkedIn Facebook Twitter Send Embed

WOMBLE BOND DICKINSON

[co-author: Nicholas Acevedo]

Most people don’t know, or care to know, about cryptography. Without cryptography, the internet privacy that we all rely on for transmitting

WRITTEN BY:

Womble Bond Dickinson

Contact + Follow

Robert Botkin + Follow

PUBLISHED IN:

Blockchain

Protocols

Protocols

- Classical cryptographic protocol:

Encrypted
Message

Attacker

Caesar, Centurion: Wikipedia. Barbarians by Flickr user skitzianist used under a Creative Commons license.

Protocols

- Modern cryptographic protocol:

Key Exchange, Validation,
Content Delivery, etc.

Attacker

Image of Jon Johansen from http://nanocr.eu/picture.php, used under a Creative Commons license.

Protocol examples:

- Vehicle remote control/immobilizer
  - Only legitimate owner can start the car/ unlock the doors, etc.

Protocol examples:

- Vehicle remote control/immobilizer
  - Early systems used fixed Serial Number

(SN)

Hello, SN

(SN)

Protocol examples:

- Vehicle remote control/immobilizer
  - Early systems used fixed Serial Number
  - Vulnerable to “replay attack”

(SN)

Hello, SN

Hello, SN

(SN)

Protocol examples:

- Solution: Challenge-Response
  - “Identification Friend or Foe”
  - Key is never broadcast over the air

(SN, k)

Challenge

MAC(k, Challenge)

(SN, k)

MITM

- Man in the Middle Attack
  - Route communications between car & keyfob
  - Don’t have to break the protocol --- just abuse it

(SN, k)

Challenge

MAC(k,

Challenge

MAC(k,

(SN, k)

MITM

- Not just theoretical...
  - Anderson [Chap 2]
  - Military radars use a similar technique to identify friendly aircraft
  - How do we fix this?

S.A. Plane

Challenge

MAC(k,

S.A. Radar

Challenge

MAC(k,

Cuban Attacker

Round Trip Timing

- The case of DTCP-IP
  - Content transport protocol
  - Concern: prevent user from sharing content over the Internet,
  - Ensure that Sink is within 7ms of Source

Source

Key Exchange

Challenge

Response

Verify ≤7ms RTT

Sink

Round Trip Timing

Establish Shared Key K

For attempt N = 0 to 1023

N, C = HASH1(K | N)

R = HASH2(K | N)

Check that R= HASH2(K + N)
and response time within 7 ms
(If not, retry)

Ok, I'm happy!

Check that C = HASH(K | N)

This check happens
way too late!

Implementation

Implementation

- Sadly, this is where most systems fail
  - Particularly if they're software-based

Oracle Security Alert #37
Created: 1 August, 2002
Updated: 5 August, 2002
Updated: 9 August, 2002
Updated: 24 September, 2002

OpenSSL Security Vulnerability

Description:

There are remotely exploitable buffer overflow vulnerabilities in OpenSSL versions prior to 0.9.6e.
These vulnerabilities may allow a remote attacker to execute arbitrary code or perform a denial-of-service (DoS) attack.

Vulnerability in Citrix Presentation Server could result in cryptographic settings not being correctly enforced

Console Hacking 2008: Wii Fail

Is implementation the enemy of design?

marcan and bushing
Team Twiizers

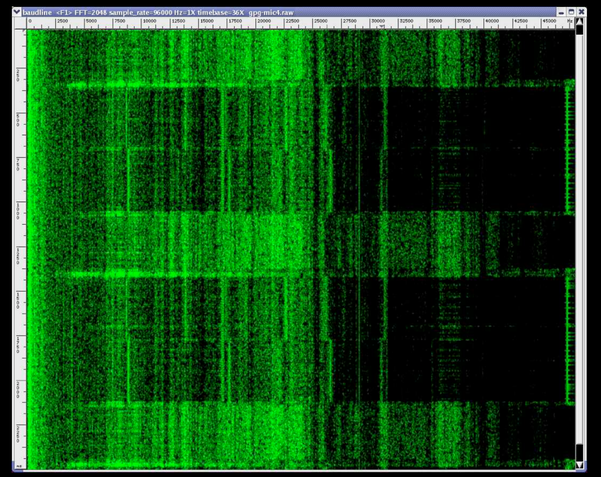
Implementation

KIM ZETTER SECURITY 01.08.16 7:00 AM

NEW DISCOVERY AROUND JUNIPER BACKDOOR RAISES MORE QUESTIONS ABOUT THE COMPANY

- Typical problems:
  - Poor protocol implementation
  - Bad PRNGs
  - Software vulnerabilities
  - Untrusted platforms
  - Side channel attacks
  - Weak hardware

USN-612-2: OpenSSH vulnerability



Source: Tromer, Acoustic Cryptanalysis: http://people.csail.mit.edu/tromer/acoustic/

Software

- Routine coding errors
  - Use strcmp() instead of memcmp()
  - Don’t check your buffer bounds
  - Don’t check your malloc() responses
  - Code anything secure on Windows
  - Write your own OpenSSL
  - Use the real OpenSSL...

Software

- Routine coding errors
  - Use strcmp() instead of memcmp()
  - Don’t check your buffer bounds
  - Don’t check your malloc() responses
  - Code anything secure on Windows
  - Write your own OpenSSL
  - Use the real OpenSSL...

Software

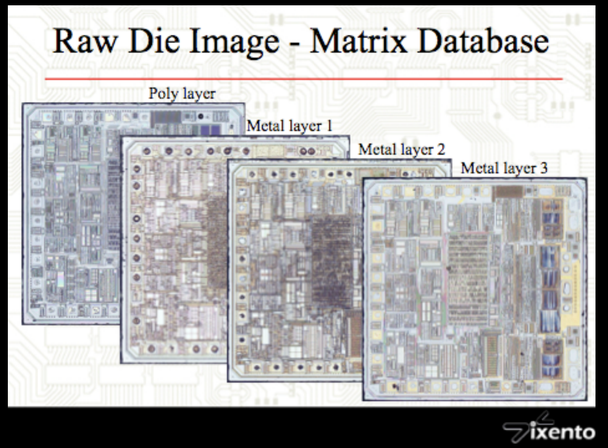
- More sophisticated issues
  - Which cryptographic libraries to use?
  - How to manage keys?
    (hint: not like this)

```
#define DESKEY ((des_key*)"F2654hD4")
```

  - Will keys be booted out into swap?

Hardware

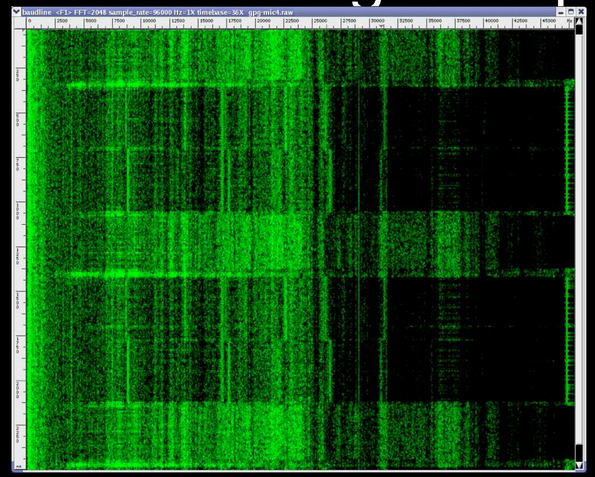
- May lead to a false sense of security
  - The notion that the bad guys can’t crack open/reverse engineer your system
- Tamper-evidence
  - Detect malicious activity
- Tamper-Resistance
  - Better
  - Depends on who’s tampering, and how.



Thanks Quing Wang and iXento.

Side-Channel Attacks

- Even when perfectly implemented
  - System can leak information through a "side channel":
    EM, power consumption, audio, timing
  - E.g., recovering RSA keys via low-bandwidth audio -- using a cellphone!



Usage

Usage

- Unfortunately, users may be your greatest foe
  - Weak password choices,
    refusal to change defaults
  - Insistence on backdoors, fail-open mechanisms
  - Loss of key material, data
  - And so far we’re talking about the honest users!

Usage

- Insider attacks:
  - Almost impossible to deal with
  - Ultimately relies on policy, vigilance
  - Where possible:

•minimize trust

•provide for renewability

Concept

Concept

- Certain things cannot be done
  - Perfect (software) DRM
    (i.e., user can watch/play/use the encrypted content, but can't decrypt it themselves)
  - Cryptographic software obfuscation
    (general case)
- Ok, if you understand:
  - These systems can at most slow down the attacker

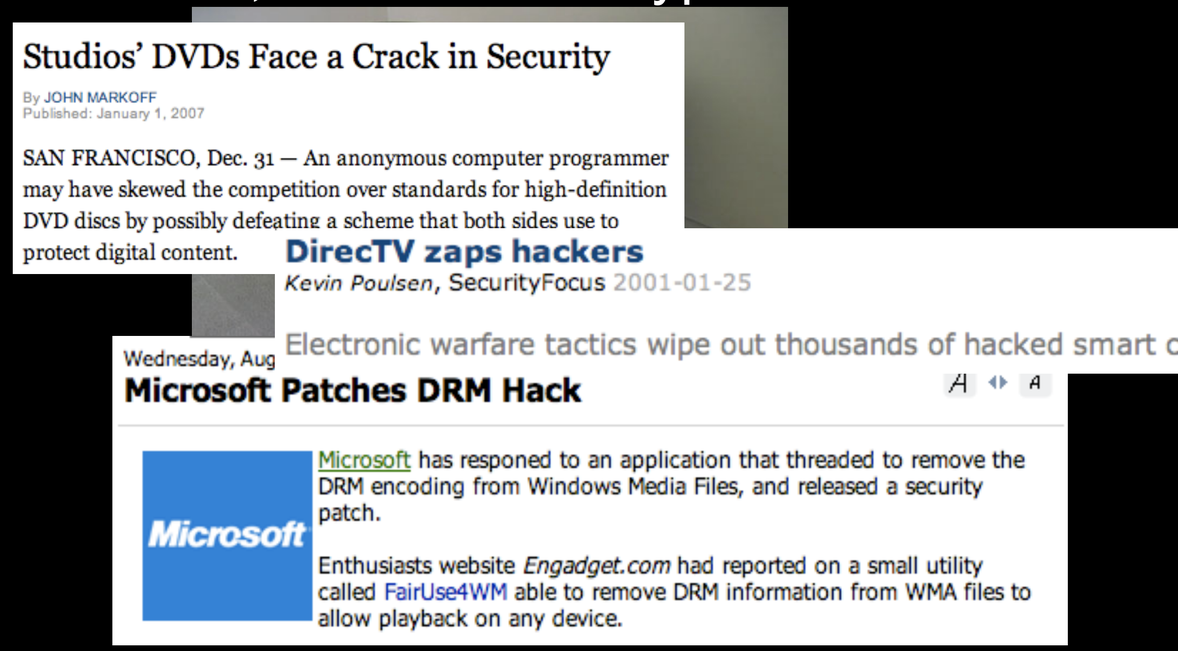


Image source: http://chris.musgrave.org/projects/infiniteloop/index.html

Kerckhoffs' Principle(s)

- Auguste Kerckhoffs (1835-1903)

1. The system must be practically, if not mathematically, indecipherable;
2. It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience;
3. Its key must be communicable and retainable without the help of written notes, and changeable or modifiable at the will of the correspondents;
4. It must be applicable to telegraphic correspondence;
5. It must be portable, and its usage and function must not require the concourse of several people;
6. Finally, it is necessary, given the circumstances that command its application, that the system be easy to use, requiring neither mental strain nor the knowledge of a long series of rules to observe.

Public Domain image from David Kahn's The Codebreakers, credited to "Eugen Drezen, Historio de la Mondo Lingvo (Leipzig, 1931), 102".

Kerckhoffs' Principle(s)

2. It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience;

"The enemy knows the System"
-- Claude Shannon's Maxim

Public Domain image from David Kahn's The Codebreakers, credited to "Eugen Drezen, Historio de la Mondo Lingvo (Leipzig, 1931), 102".

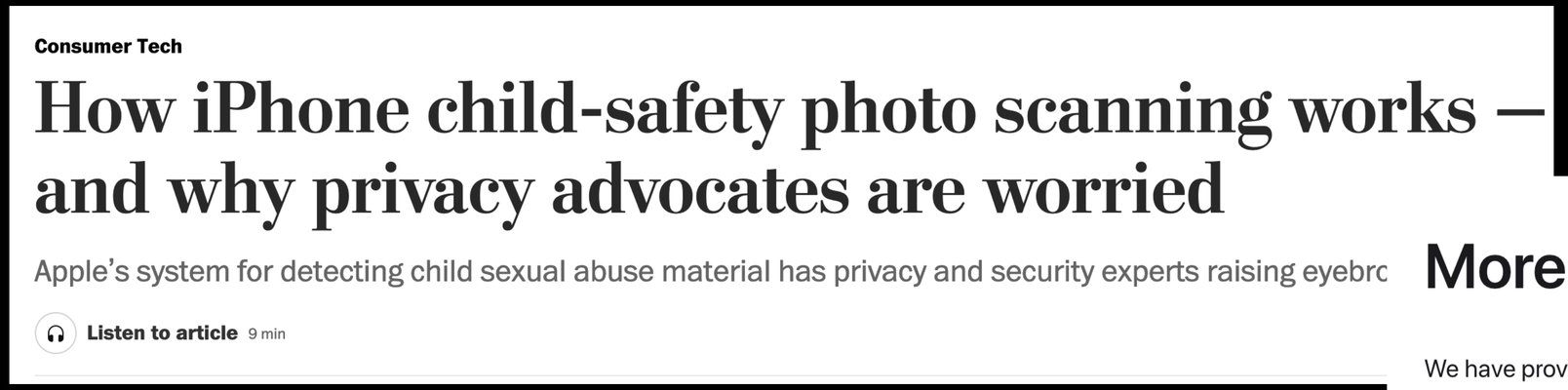
Kerckhoffs' Principle(s)

2. It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience;

"The enemy knows the System"
-- Claude Shannon's Maxim

Public Domain image from David Kahn's The Codebreakers, credited to "Eugen Drezen, Historio de la Mondo Lingvo (Leipzig, 1931), 102".

Kerckhoffs' Principle(s)



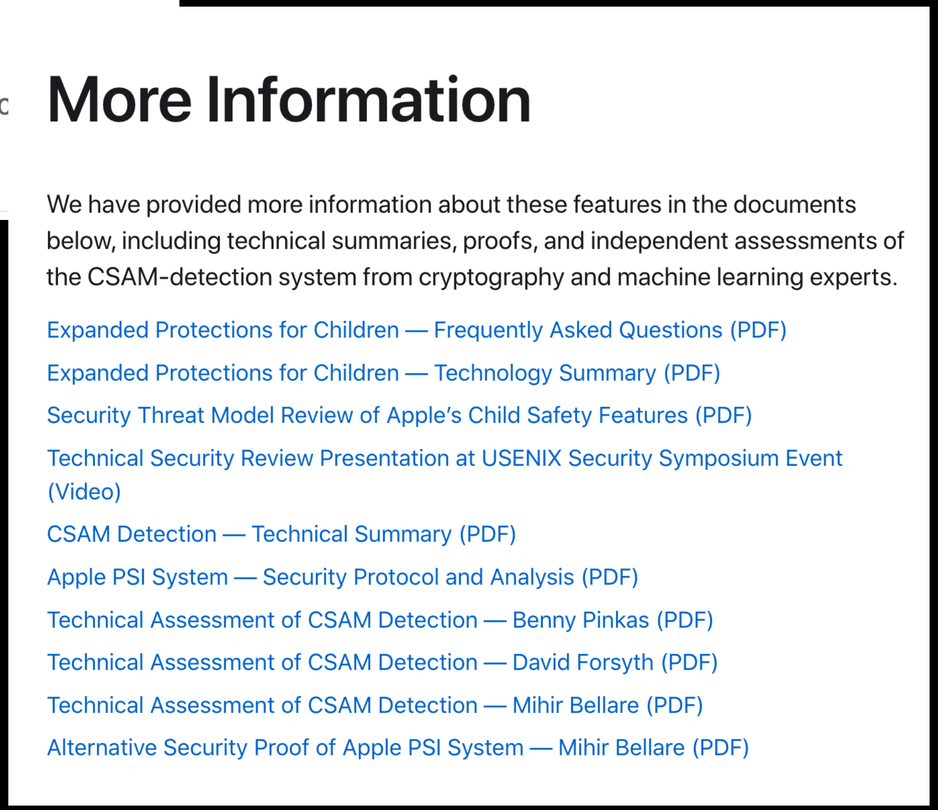
Consumer Tech

How iPhone child-safety photo scanning works — and why privacy advocates are worried

Apple's system for detecting child sexual abuse material has privacy and security experts raising eyebrc

Listen to article 9 min

August 5, 2021



More Information

We have provided more information about these features in the documents below, including technical summaries, proofs, and independent assessments of the CSAM-detection system from cryptography and machine learning experts.

Expanded Protections for Children — Frequently Asked Questions (PDF)

Expanded Protections for Children — Technology Summary (PDF)

Security Threat Model Review of Apple's Child Safety Features (PDF)

Technical Security Review Presentation at USENIX Security Symposium Event (Video)

CSAM Detection — Technical Summary (PDF)

Apple PSI System — Security Protocol and Analysis (PDF)

Technical Assessment of CSAM Detection — Benny Pinkas (PDF)

Technical Assessment of CSAM Detection — David Forsyth (PDF)

Technical Assessment of CSAM Detection — Mihir Bellare (PDF)

Alternative Security Proof of Apple PSI System — Mihir Bellare (PDF)

Kerckhoffs' Principle(s)

Another worry is that the new technology has not been sufficiently tested. The tool relies on a new algorithm designed to recognize known child sexual abuse images, even if they have been slightly altered. Apple says this algorithm is extremely unlikely to accidentally flag legitimate content, and it has added some safeguards, including having Apple employees review images before forwarding them to the National Center for Missing and Exploited Children. But Apple has allowed few if any independent computer scientists to test its algorithm.

August 11, 2021

Kerckhoffs' Principle(s)

Kerckhoffs’ Principle(s)

August 18, 2021

Kerckhoffs’ Principle(s)

August 19, 2021

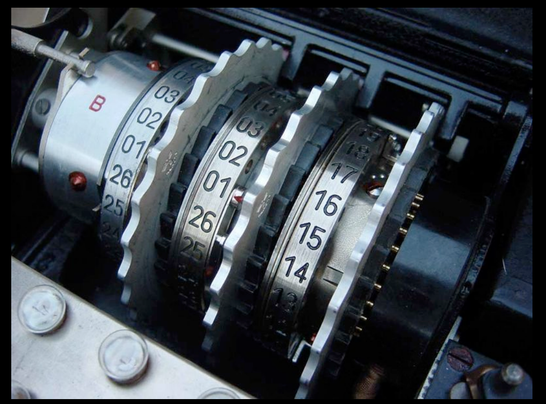
Don’t worry!

- I’m not all doom & gloom
  - We can do some things very well
  - Other things fairly well
  - Still others... well-ish
- We can certainly do better than most

Going forward:
The next few weeks

Part 1

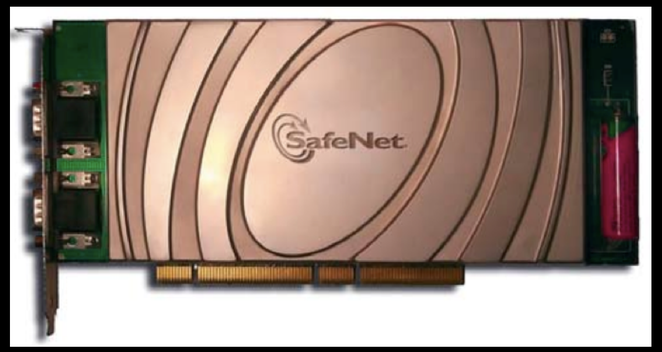
- (Re-)introduction to Crypto... at high speed:
  - Classical cryptography
  - Symmetric-key encryption & block ciphers
  - AES, ChaCha
  - Public-key cryptography
  - Diffie-Hellman, RSA



Enigma image from Wikipedia, used under GFDL.

Part 2

- Exploiting Software:
  - Crypto vulnerabilities and where they turn up
- Protocols
  - Signal
  - TLS



Part 3

- Reductionist security & protocols
  - Proving the security of a construction
  - Analyzing protocols that fail
- Random number generation
- Security evaluation
  - What a security evaluation process looks like
  - The FIPS standards

A Note on Ethics

- We'll be discussing vulnerabilities in many systems
  - Some have been fixed
  - You might find more
  - It goes without saying: exploiting systems is often a crime. Be careful.
  - Important to disclose vulnerabilities responsibly

END