The Increasing Complexity & Need for Validation

Introductory Keynote

NIST Physical Security Testing Workshop – Honolulu
Monday, Sept. 26, 2005

Paul Kocher

President & Chief Scientist

Cryptography Research, Inc.

www.cryptography.com

575 Market St., 21st Floor, San Francisco, CA 94105

CRYPTOGRAPHY RESEARCH

CRYPTOGRAPHY RESEARCH

About Cryptography Research

- Founded in 1995:
  - Goal: Help understand and solve important real-world security problems
  - Major applied focus: Products incorporating CRI technology secure over $100 billion in commerce annually
- Main industries served:
  - Financial Services
  - Wireless / Telecommunications
  - Pay Television
  - Internet
  - Entertainment
- Business areas:
  - DPA countermeasure licensing
  - Anti-piracy technology licensing (pay TV, optical disc formats)
  - Other areas include consulting services, DPA workstation, education

The absence of functionality

se·cu·ri·ty – (*n.*) Freedom from danger.
(Merriam-Webster Dictionary)

Security is a negative property…

- From an engineering perspective, security is usually the absence of classes of certain vulnerabilities
  - Confidence that undesirable property is absent
  - Example: Remote control via buffer overflow

Security vs. Functionality

Functionality → **Reliability** → **Security**

Correct operation in *typical* situations	Correct operation in *all naturally-occurring* situations	Correct operation in *maliciously-chosen* situations
Unexpected functionality is fine ("a bonus")	**Unexpected functionality makes testing harder**	**Unexpected functionality is the problem**

Historical view of cryptographic security

- Computation used to be very expensive
  - Manual & (electro)mechanical ciphers have tight limits on computational complexity – achieving functionality is difficult
  - This made cryptanalysis hard to defeat

Today: Algorithm design has beaten cryptanalysis

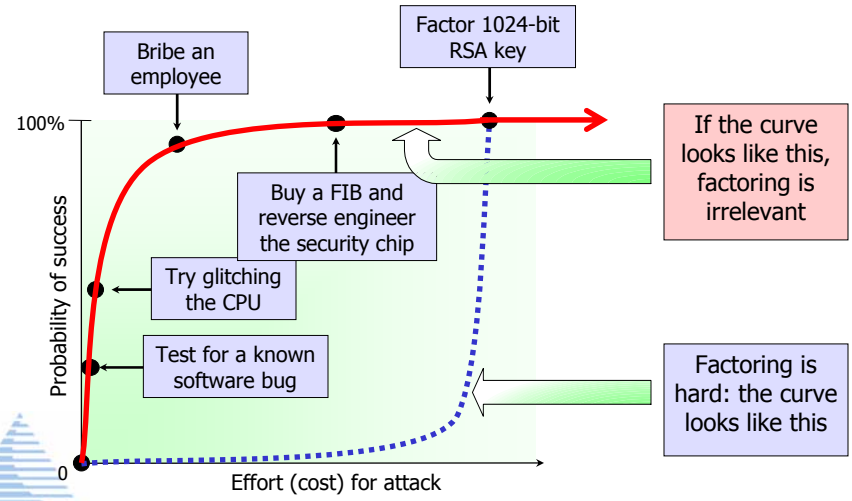
- Modern algorithms can infeasible to break with mathematical attacks
  - Brute force, factoring, adaptive chosen message attacks…

- Margin of victory is <u>increasing</u>
  - Moore's Law favors the cryptographer
  - 2-8X CPU power => 2-3X key length =>*square* the effort for cryptanalysis
    - Example: DES → triple DES takes 3X CPU but increases brute force effort by ~2^{56} times (~72,000,000,000,000,000X)

But something is wrong…

We have "unbreakable" cryptographic primitives, but cannot affirmatively categorize any useful real-world system as "secure".

Interactions, complexity, & low-hanging fruit

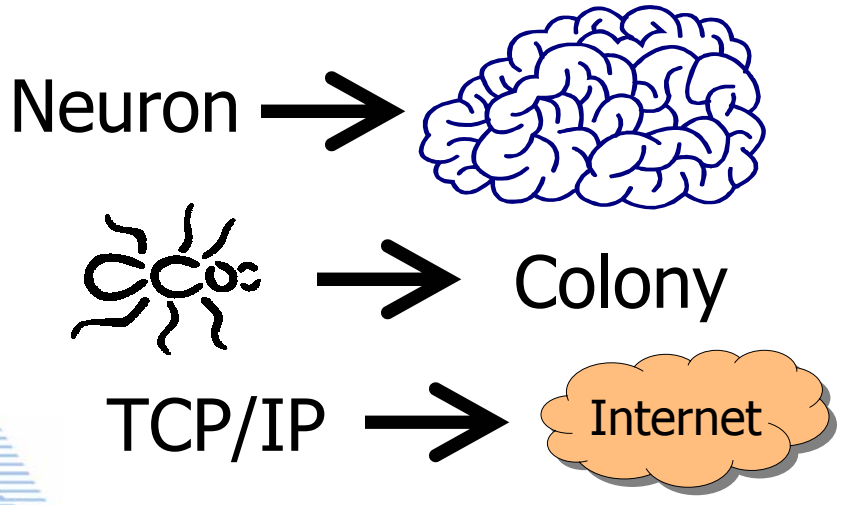


Modeling risks

- Common tool: Attacker risk matrix
  - Attacker characteristics (type, motivation, budget…)
  - Attacker strategies (specific known attacks)
  - Works for attacks we understand
    - If the model shows a device is inadequate, it probably is
- Problem: Attacker creativity
  - We can't reliably model what we don't know

… but we can look at where these risks come from.

Emergent properties



Emergent properties

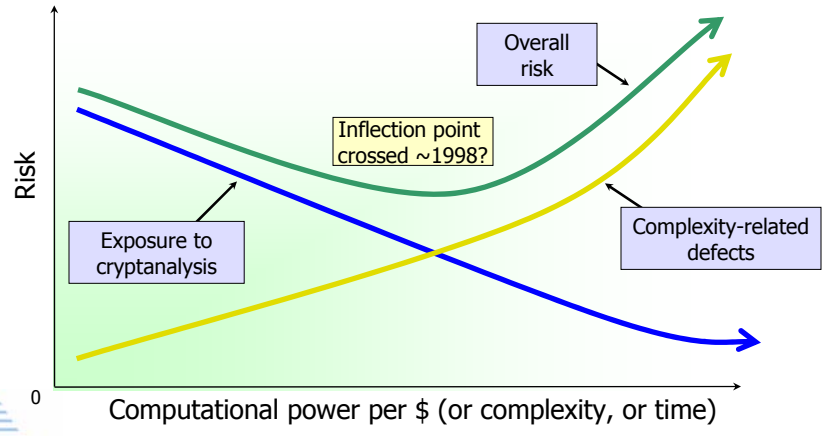
- Basic concept:
  - As elements are combined to form a system, properties can emerge that are not obvious given the elements in isolation
- For digital systems that manage data:
  - When combinations of desired capabilities are combined, unexpected characteristics emerge

Emergent insecurity

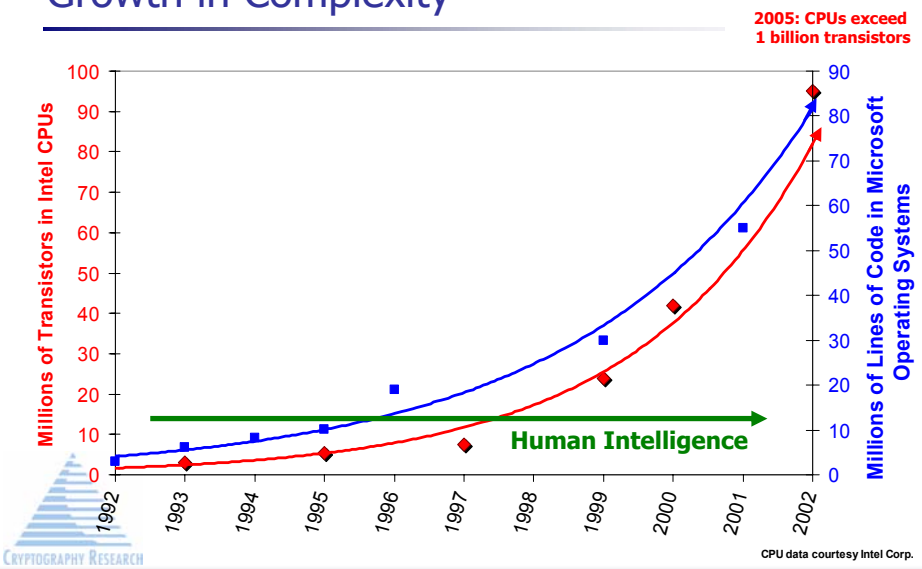
- Our ability to understand each of the individual pieces of complex systems *in isolation* creates a false impression that we can understand how they work together.

Insecurity is an emergent property that appears as complexity increases.

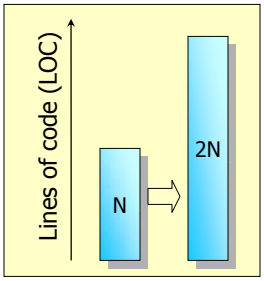
Cryptographic security vs. cost of computation

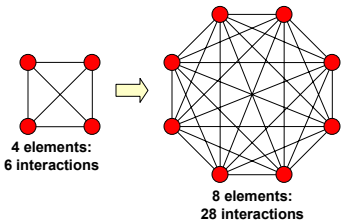


Growth in Complexity



What happens as complexity increases?





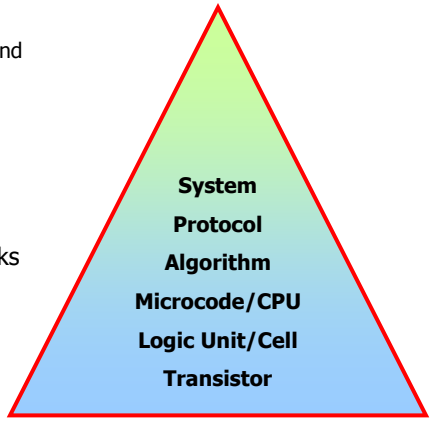
- What if the number of security-critical elements doubles?
  - 2X more opportunities for bugs
    - If defect density per element (e.g., LOC) is constant, the odds of having no catastrophic weaknesses is *squared* (e.g., 20% -> 4%)
  - Number of interactions increases as the *square* of complexity
    - If defect density per interaction is constant, the odds of having no catastrophic weaknesses goes up to the 4th power (e.g., 20% -> 0.016%)
  - This assumes that flaw densities do not also increase…

Not only a crypto module problem

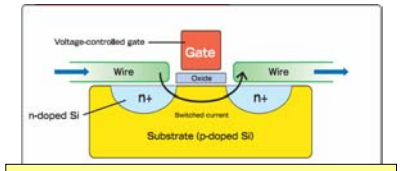
- Example: Complexity of computer networks
  - Individual nodes are getting exponentially more complex
  - Networks themselves are getting much more complex
- Abstraction & complexity are muddying everything
  - Example: Ports used to signify protocols (talk, IRC, FTP…)
  - … so people used them to enforce security policies
  - … which made them unreliable
  - … so developers found more reliable methods (e.g., port 80)
- Even the OSI model is breaking down
  - Example: Overlay networks like the one used by Skype
  - I'm not intending a value judgment (this is not necessarily bad) … but it does means our past experience is obsolete

Layers of abstraction

- Layers of abstraction help manage emergent properties
  - Concealing underlying details
    - Eliminates need to specify understand what is going on at lower layers
- Good for functionality
  - Essential for complex engineering
- Bad for security
  - Layers hide underlying security risks
  - People specialize in individual layers, creating unintended interactions that no single individual understands



Layers of abstraction: Example (SPA)



Power variations are a fundamental property of circuits' transistors & wiring

Power variations were an undocumented property of logic units, CPUs, but…

Algorithms didn't take this into account

Compromised system & protocol security

System

Protocol

Algorithm

Microcode/CPU

Logic Unit/Cell

Transistor

We can't go back

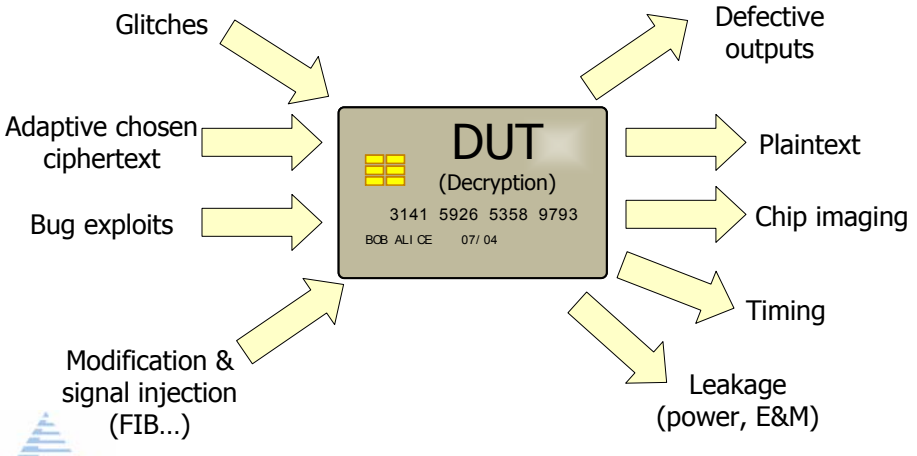
- Users won't accept the reduced functionality of older systems
  - Companies and governments have to use modern technology to keep up with their rivals
- Except for basic crypto algorithms and protocols, security defenses are crumbling
  - Example: All major OSes have catastrophic security bugs

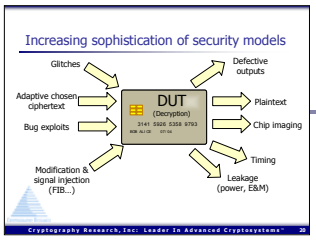
What does this mean for cryptographic modules?

Key trend:

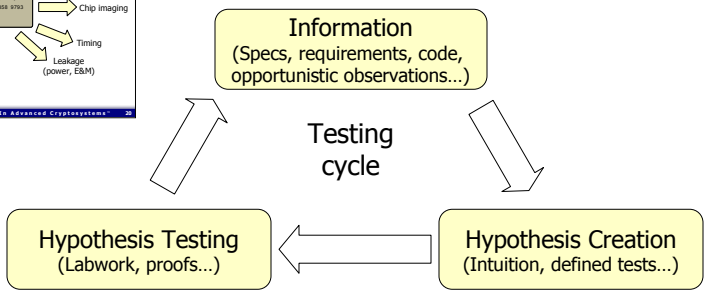
Validation is becoming more difficult and more important.

Increasing sophistication of security models





More to consider in the validation cycle



- More factors to consider when assessing whether a product design has correct, verifiable evidence of its security
  - Harder for relying parties to define their requirements
  - Harder for designers to provide this evidence
  - Harder for evaluators to verify this evidence
    - Many ways that the security 'warranty' can fail

Voiding the warranty

Look for anything that might void a design assumption.
Few initial security observations are exploitable attacks.

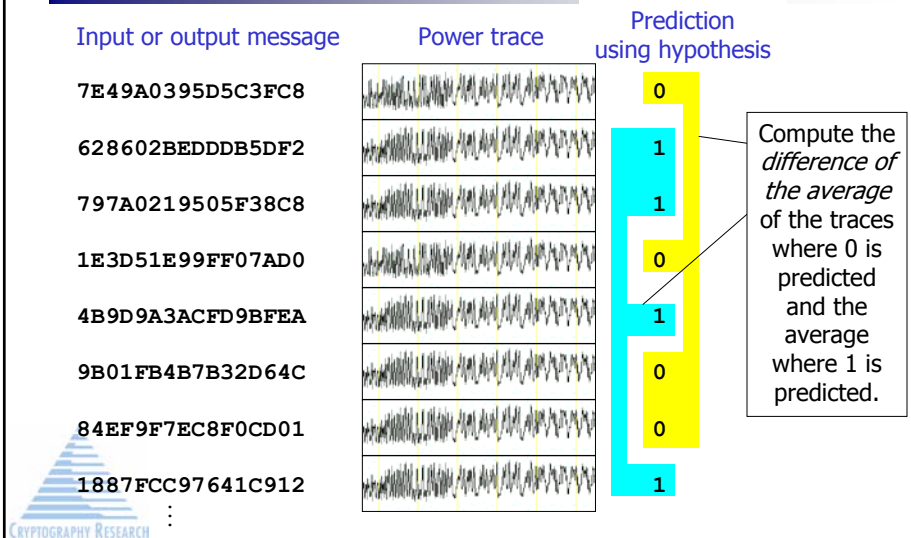
- "The amount of time consumed depends on the input value."
- "The first bytes out of RC4 aren't quite random for related keys."
- "The device crashes for a particular malformed input."
- "If the secure line is down, users will switch to an insecure one."
- "Error messages can convey information about the key."
- "Configuration files can be set to be read-only."
- "The device's ID is transmitted twice during the protocol."
- "DRAM errors become common above 120°C."
- "On-chip computations create electromagnetic radiation."
- "Cosmic rays cause random bit errors in DRAMs."

DPA is a classic example

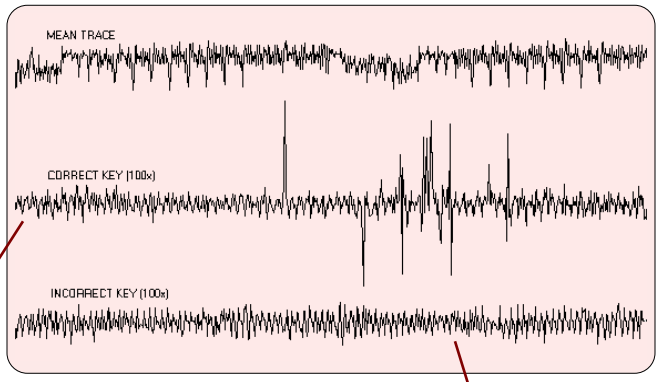
- Gives attackers a tiny bit of extra information
  - Ciphers are designed with the assumption that no information about intermediates is revealed
  - Power consumption measurements show that this assumption is often wrong
  - DPA is an attack that takes advantage of this mismatch

DPA: Hypothesis testing using statistics to exploit tiny leaks buried in noise

Live AES example: dpa_aes.bat
(see selection function results, several wrong and one right that solves for 8 bits of the key)



A typical DPA result



If the hypothesis is right

Predictions will have some (perhaps tiny) correlation to what the device did, and difference of the averages will approach a nonzero value in these places

If the hypothesis is wrong

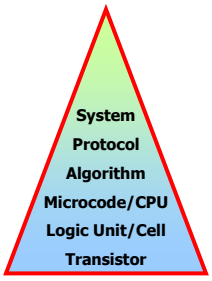
Predictions have no correlation to what the device actually did, so the difference of the averages will approach 0 (flat) everywhere

"Difference of the averages"

- The statistics automatically pull the key from the noise
  - Enables testing of arbitrary hypotheses
  - Noise, measurement errors, etc. all vanish as the number of measurements increases

What happened?

A characteristic of transistors (the lowest layer) compromised each layer above, ultimately compromising the system & business objectives.



[[Live Demo]]

0_cleanup.bat
1_show_data_1024.bat
1_show_data_800.bat
2_generate_selection_functions.bat
3_compute_averages.bat
4_show_results_1024.bat
4_show_results_800.bat (67, 82…)

In conclusion…

A look backward then a look forward

Power analysis: A brief retrospective

- Evaluation customers wanted results
  - Had to get keys out of smartcards & other devices quickly & cheaply
  - Invasive attacks required nasty chemicals
- Started with a cheap analog oscilloscope…
  - Then quickly built advanced digital tools + iterated
  - Also filed for countermeasure patents
- Kept problem confidential until AFR article "outed" DPA
  - Published academic paper & released more info
- Area of exciting research (though not without controversy)
  - Reinvigorated crypto implementation research (CHES…)
  - Academic & industry: Hundreds of papers & researchers
  - Products: Dramatic improvement in smartcards (other products have a long way to catch up)

Six directions in physical security

- High-assurance design
  - Goal: Realistic, testable assumptions about implementation
  - Approach: Limit complexity & make elements reinforce
- Emphasis on noninvasive attacks
  - Goal: Make devices that cannot be broken cheaply (attacker goal: keys)
  - Approach: Ongoing area of interdisciplinary research (math, hw, sw, physics…)
- Risk management & renewable (reactive) security
  - Goal: Ability to recover from unexpected failures
  - Approach: Ensure knowledge of attacks & ability to respond effectively
- Physical features to improve tamper resistance
  - Goal: Build effective defenses against specific attacks
  - Approaches: Shield layers, light sensors, power flattening…
- Software, algorithms and protocols
  - Goal: Make implementations secure for real-world hardware + usage scenarios
  - Approach: Mitigate risks (Blinding, randomization, obfuscation, better UI's…)
- Testing standards
  - Goal: Define processes that challenge vendors to make better products
  - Approach: Continuously update standards & educate relying parties to keep pace with risks

Questions?

Paul Kocher
paul@cryptography.com

Cryptography Research, Inc.
575 Market St., 21st Floor
San Francisco, CA 94105 USA

www.cryptography.com

Tel: +1 (415) 397-0123
Fax: +1 (415) 397-0127

p.s. we're seeking strong technical folks who want to join our team in San Francisco!