Obvious in Hindsight:

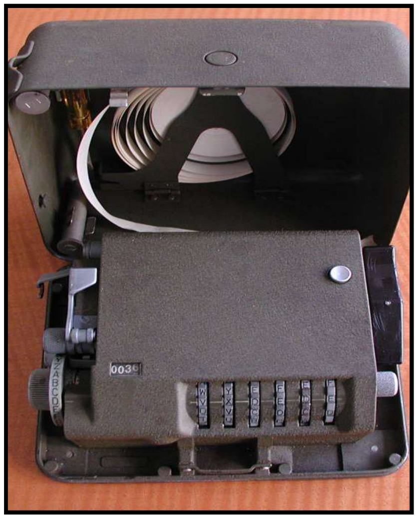
From Side Channel Attacks to the Security Challenges Ahead

Invited talk at CHES 2016 & CRYPTO 2016

Paul Kocher

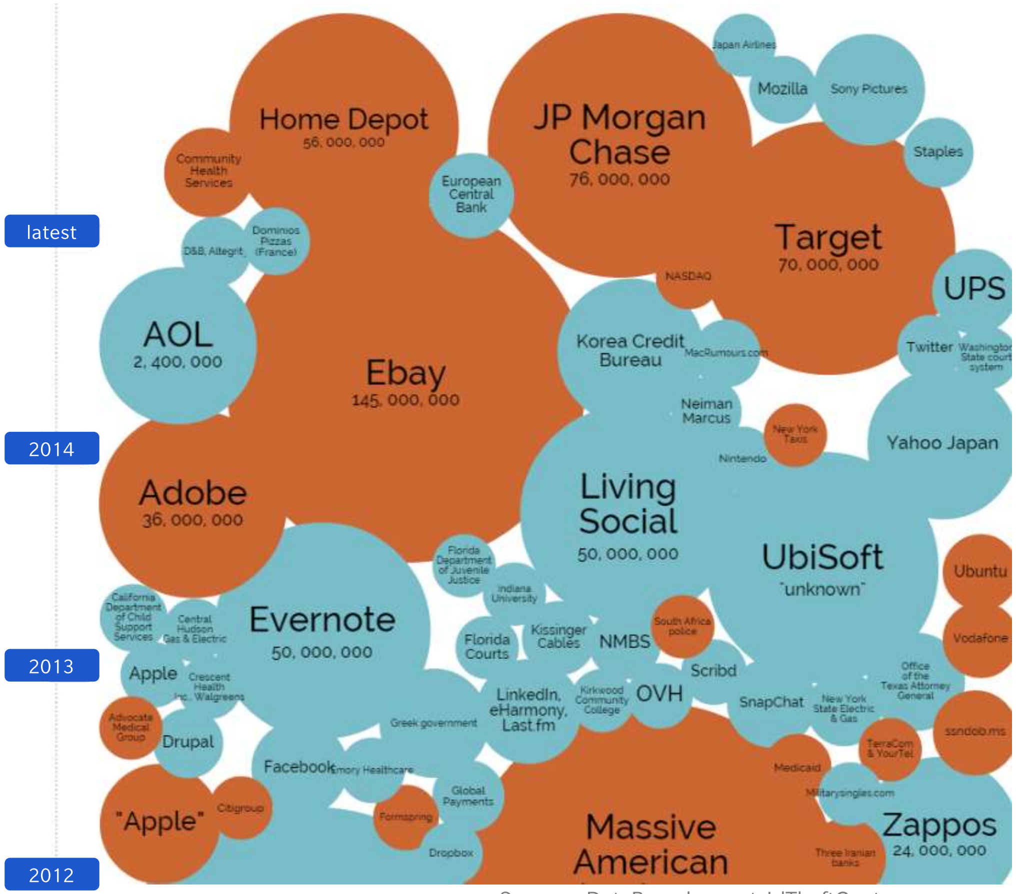
President & Chief Scientist
Cryptography Research Division of Rambus

August 17, 2016



- Scaling favors crypto strength (DES → 3DES: ~3X work = ~2^{56}X strength)
- Algorithms have now won, if we don't over-optimize
  - Prediction: No practical cryptanalysis of triple AES-256 – ever

… but security obviously isn't going well … incl. crypto



Sources: DataBreaches.net, IdTheftCentre

bitcoin

ethereum

…

- Inputs.io (2013: ~$1M)
- BIPS (2013: ~$1M)
- Mt. Gox (2014: ~$350M)
- Bitpay (2014: ~$2M)
- Flexcoin (2015, ~$650K)
- bitstamp (2015 ~$5M)
- BTER (2015: ~$2M)
- Cryptsy (2016: ~$6M)
- Bitfinex (2016: ~$60M)
- Gatecoin 2016 ~$2M
- Ethereum DAO (2016: ~$50M)
- (and more…)

https://magoo.github.io/Blockchain-Graveyard/

In the middle ages…

“Physicians tended to be academics, working in universities, and mostly dealt with patients as an observer or a consultant. They considered surgery to be beneath them.” [1]

… so surgery was done by barbers

[1] https://en.wikipedia.org/wiki/Barber_surgeon

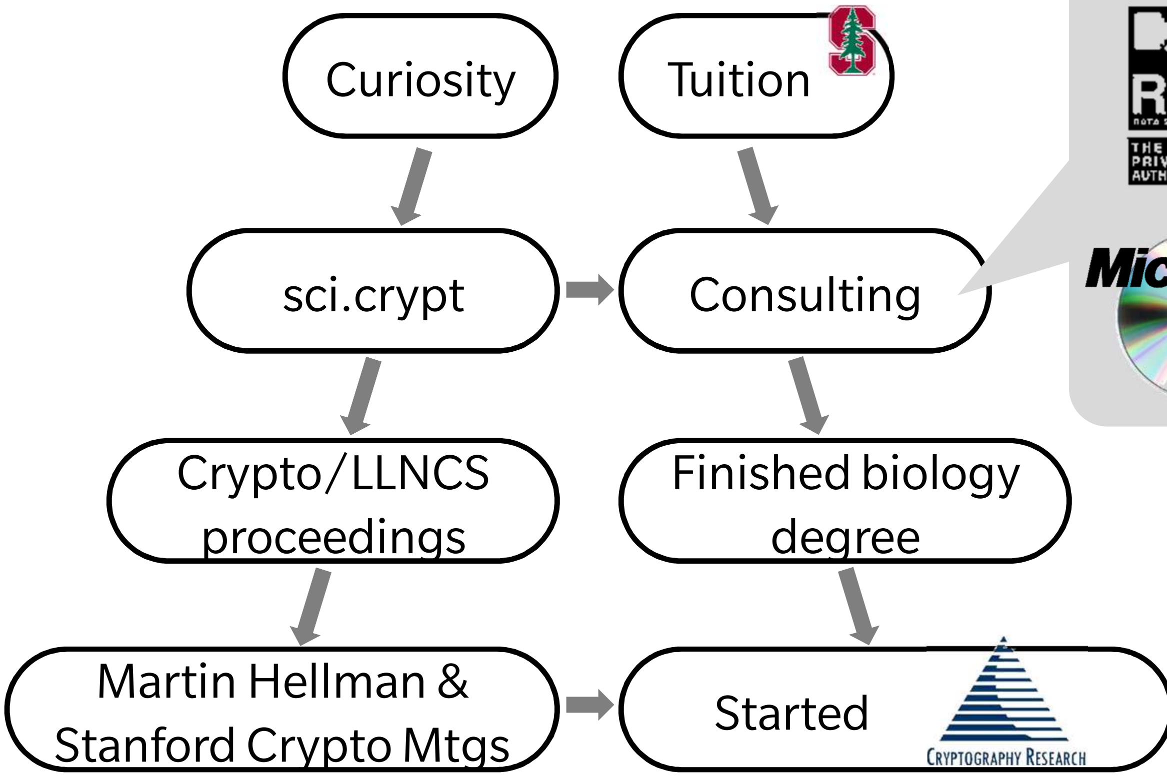
Crypto Theory

Our ‘barber surgeon’ era

- Practice yields many bad outcomes (and a few very good)
- Research too divorced from practice
  - Theory struggles with messy reality
  - Theory isn’t applicable
  - Practice ignores theory
- Dire needs: Practice goes on

Practice

Barbers doing surgery <-> pre-vet students doing crypto?



Implementing, finding issues (RC4 related keys), patents

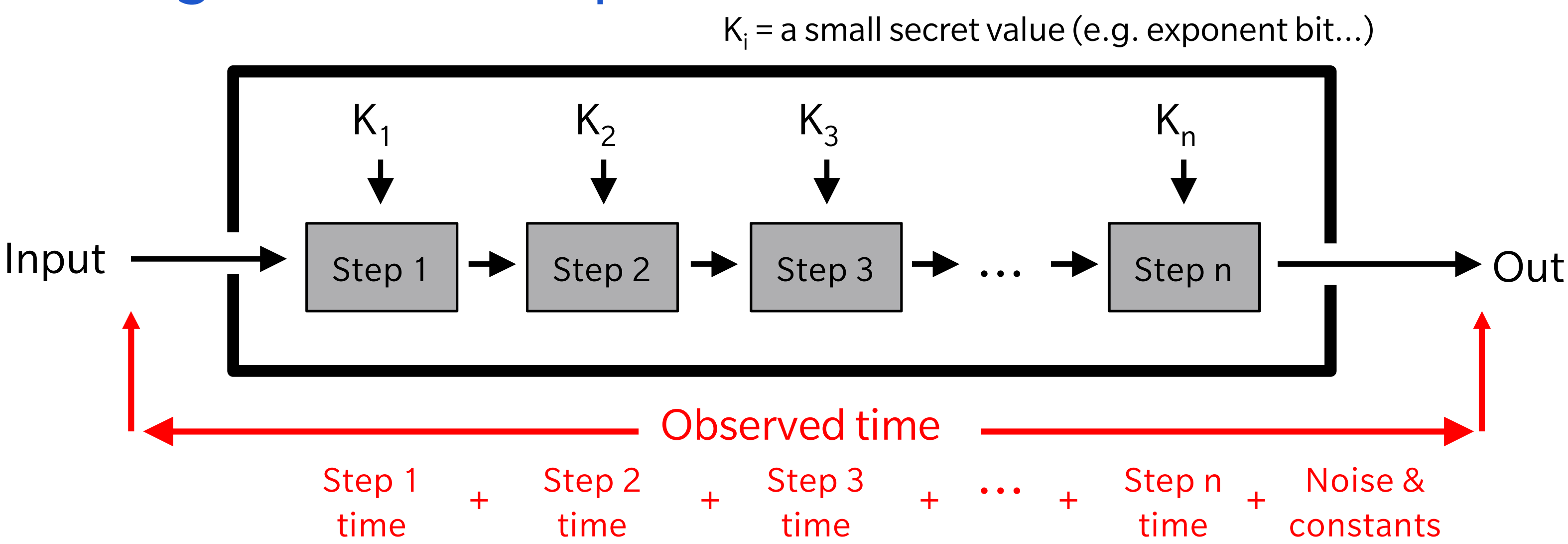
Breaking "try before you buy" schemes

- Presentation @ Stanford on Differential & Linear Cryptanalysis
  - Tried improving – failed
  - Frustratingly weak correlations
  - Knew timing was non-constant from profiling my own code

    MUL, caches, branches…

MUL	8-bit Reg	70 – 77	2
	16-bit Reg	118 – 133	2
	8-bit Mem	(76 – 83) + EA	2-4
	16-bit Mem	(124 – 139) + EA	2-4
IMUL	8-bit Reg	80-98	2
	16-bit Reg	128 – 154	2
	8-bit Mem	(86 – 104) + EA	2-4
	16-bit Mem	(134 – 160) + EA	2-4
DIV	8-bit Reg	80 - 90	2
	16-bit Reg	144 - 162	2
	8-bit Mem	(86 – 96) + EA	2-4
	16-bit Mem	(150 – 168) + EA	2-4
IDIV	8-bit Reg	101 – 112	2
	16-bit Reg	165 – 184	2
	8-bit Mem	(107 – 118) + EA	2-4
	16-bit Mem	(171 – 190) + EA	2-4

Timing Attack Example



Given a set of inputs and their observed transaction times:

- Can estimate time for each run of Step x given Input and all $K_{i<x}$
  - Estimates will correlate to observed time if $K_{i\leq x}$ correct – and no correlation if $K_{i<x}$ is wrong
  - Identify correct K_i, then iterate to find key

Implications

- Yielded the strong correlations I wanted
  - Modest data needs – implementable
  - More fun than linear & differential cryptanalysis ☺

- Obvious in hindsight…
  - Tiny side channels can expose keys
  - Real implementations aren't black boxes
    - Optimizations make things worse
  - Disconnect between algorithm requirements & implementation
    - Incorrect (often unwritten) assumptions
  - Crypto > mathematics

Smart Card Projects

- Clients were deploying smart cards
  - Suspiciously bold security claims
  - … but a "proper" testing lab required $$MM equipment
- Did protocol reviews
  - Consistently bad: Time-memory trade-offs, weak MACs, unpadded RSA, key reuse…
- Vendors disputed vulnerabilities
  - Got a smart card reader & implemented
- Checked for timing issues
  - Consistently bad: RSA attacks, MAC & PIN verify timing leaks, undocumented backdoors
  - Also: timed resets to reset counters, EEPROM exhaustion, faults…

Power Analysis

- Wanted better data than timing
  - Bought the cheapest analog oscilloscope at Fry's electronics
  - Resistor from Radio Shack "Science Fair 60 in One Electronic Project Lab"

- Instant SPA results, e.g.:
  - RSA (squares vs. multiplies, CRT timing…)
  - DES (with branching in C/D shift – really!)
    - At night only

Implementing DPA

- HP 54645 digital storage scope
  - 100MHz, 1MB memory (!) -- see one-time events
  - Josh Jaffe got data onto PC, visualization: SPA → DPA
- Major effort on countermeasures
  - Filed patents -- got too busy to submit to conferences 😐
- Breaking everything tested…
  - Eventually an Australian reporter found out
    - Mooted 'responsible disclosure' question
    - Initial white paper, academic paper

In retrospect…

- Obvious in hindsight
  - Changes in electron movements affect power & EM
  - Measurements correlated to secret intermediates
  - Cryptanalysis can leverage tiny correlations
    - Example: can break a tiny block cipher circuit in a big, nosy ASIC

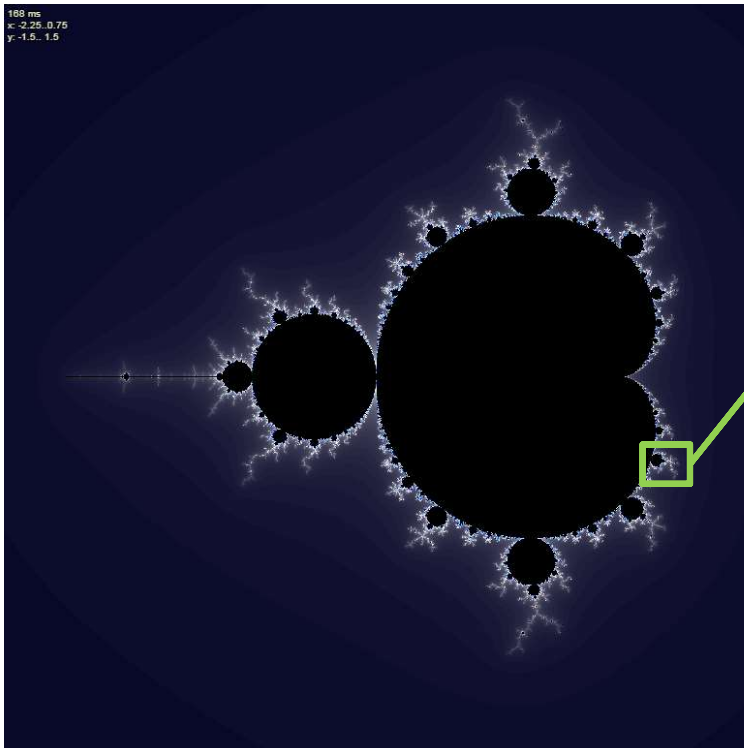
  - Strong algorithms are the beginning of crypto… not the end

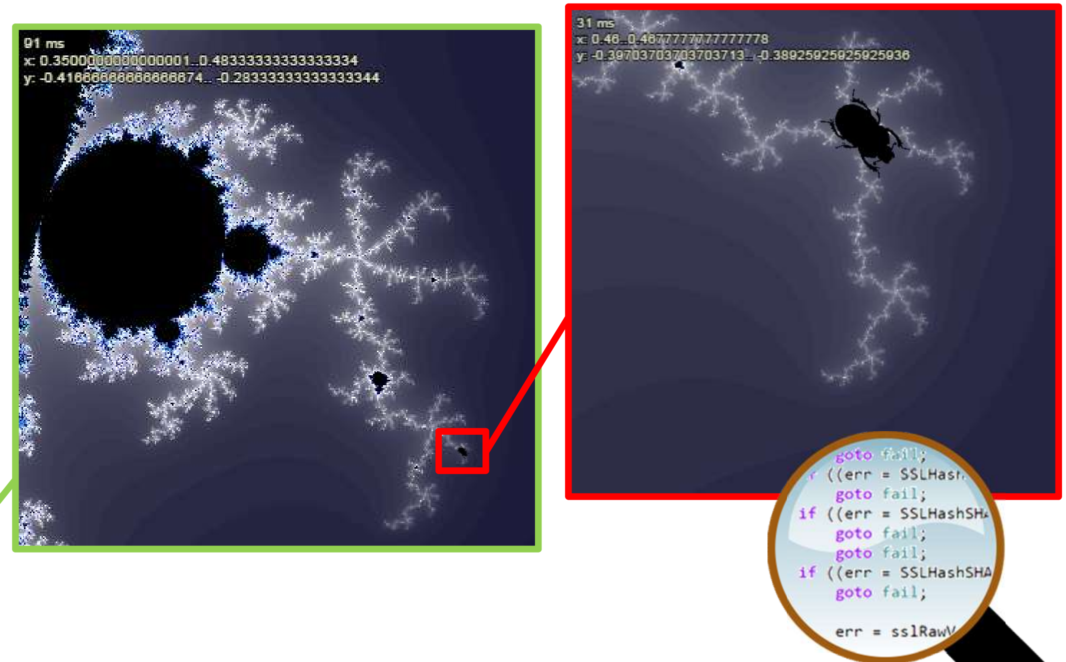
“Obvious in hindsight” != useful*

* Except for assigning blame ☹

Why aren’t problems obvious beforehand…?

Security & Fractals



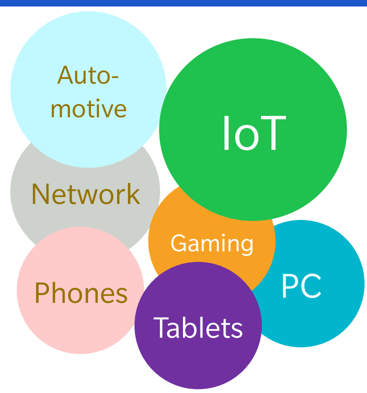


Individual vulnerabilities are “obvious”
– when we stare directly at minutiae

Overall risks are “obvious” too
– if we look broadly

Computing & Security Trends

More Devices



More Targets

More Valuable Data

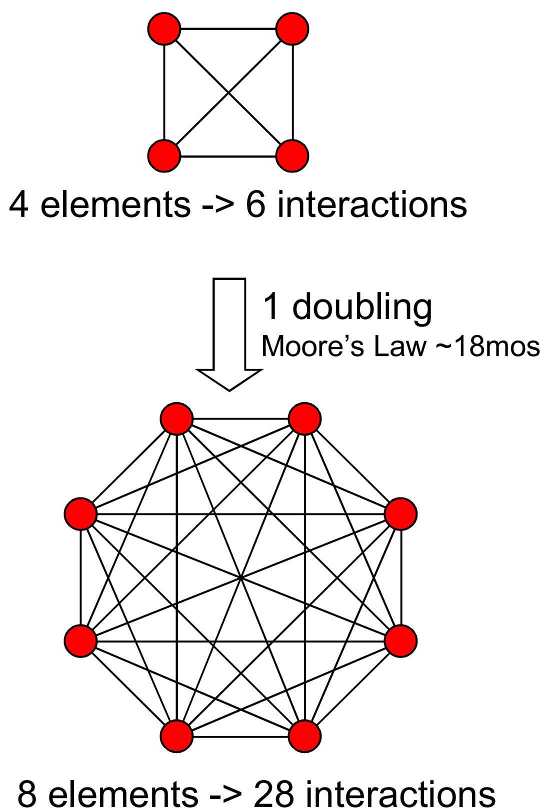
ID
cloud
passwords
Breaches
payments
DRM
IP

More Attacker Reward

More Complexity

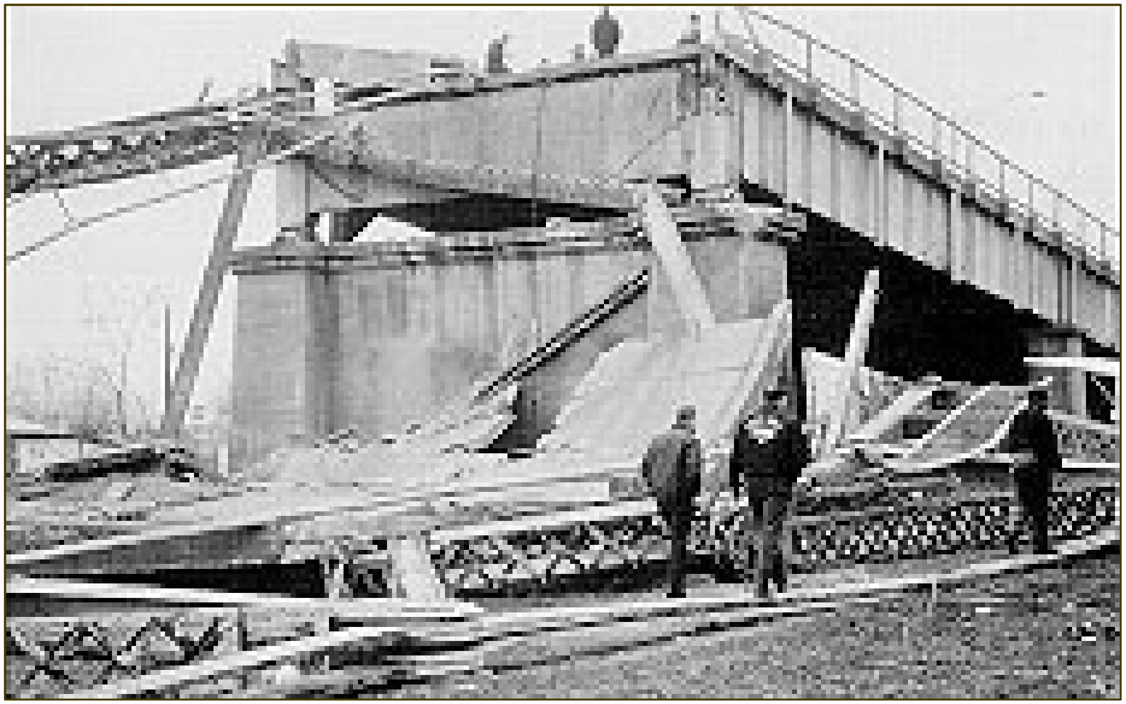
More Vulnerabilities

Complexity swamps security



- If defect density is constant per element, odds of zero flaws squares (20% → 4%)
- Reality is worse:
  - Defects reflect interactions: 4th power
  - Defect densities tend to increase

Silver Bridge on U.S. 35 in Ohio: Built 1924
Innovative optimization: High-strength steel ‘eyebars’ instead of cables



Collapsed in 1967, created awareness of “fracture critical components”

Image from model of bridge
(credit: NIST)

How many “fracture-critical” elements are in a typical connected device?

- CPU
- Additional logic
- Bits of DRAM (non-ECC)
- Bits of flash/storage
- Software instructions
- …

~10 billion (10^{10}) today…?

In 10 years ~1 trillion (10^{12})

Not counting compilers, infrastructure…

Defenses have failed to scale to today’s needs.

IoT security is much harder

	Traditional	Future (IoT…)
Product vendor security expertise	deep	limited
Secure product lifespan	5-10 years	20-50+ years
User attention to security per device	high-ish	low/none
User tolerance for security/reliability issues	high	low/none
Connected to physical world	no	yes
Number of software platforms	small	huge
On-device security tools	ubiquitous	usually none
Vendors can afford monitoring & patching	yes	no

What can we can do?

1. Focus on outcomes
2. Build better foundations

P(cryptanalysis) = small

P(mistake) = huge

Everyone wants to narrow the gap

Two approaches…

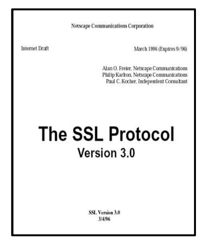
Make P(cryptanalysis) huge

Make P(mistake) small

Must think in probabilities – not certainties

- Proof != 100% confidence (mistakes, relevance, assumptions…)
  - Danger: Wrong assumptions → False confidence
- Gaps scale exponentially (fixed 75% of flaws → Gone in 2 doublings)

What P(desired outcome)?


The SSL Protocol
Version 3.0

History of massive over-confidence.

Our understanding of elements creates a false impression that we understand the complex system

What does crypto for fallible humans look like?

Goals = safety, assurance

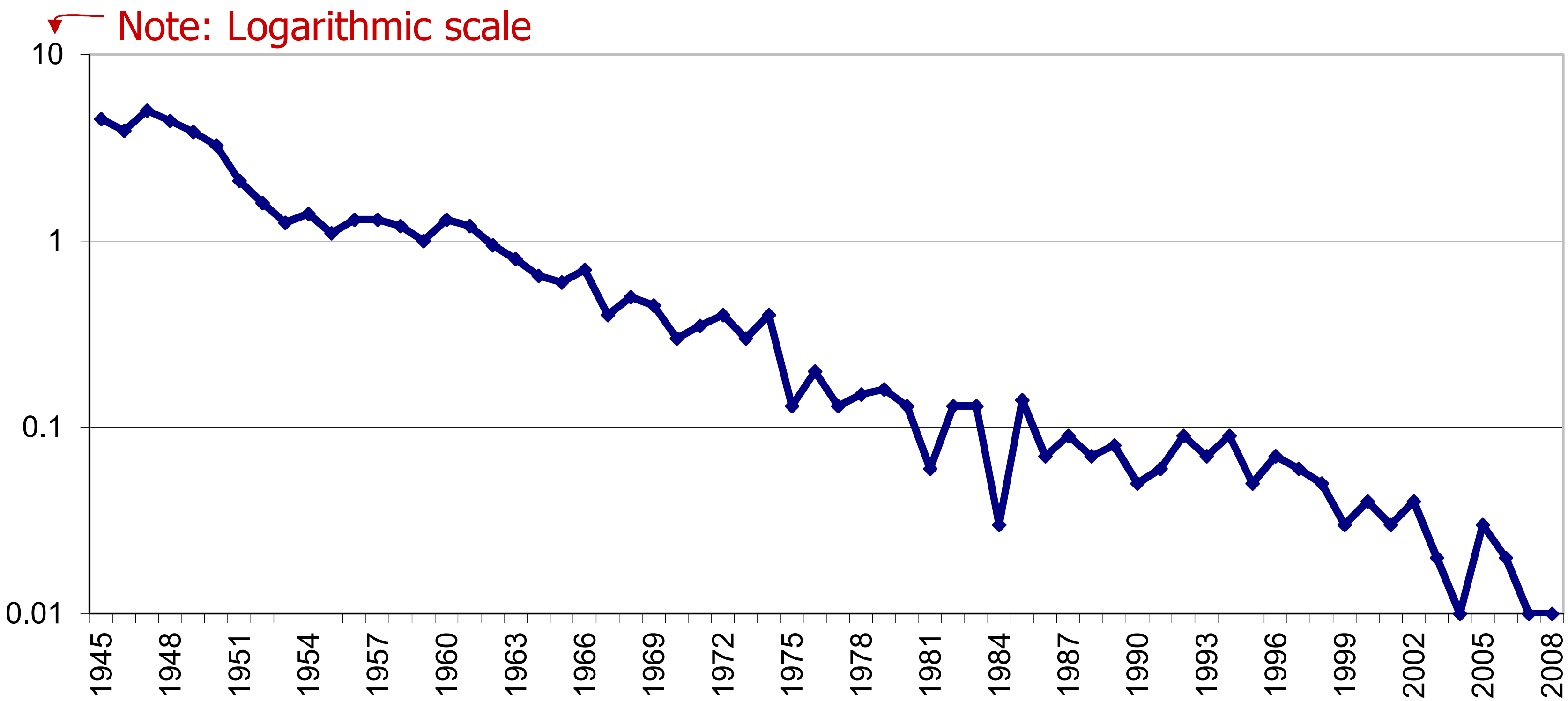
- 10X safer > 10X faster: Can 'mere mortal' practitioners usually succeed

What are the metrics, requirements, trade-offs?

- Implementation risk (few LOC, no special cases, high test coverage…)?
- Safety margins (implementation redundancy, algorithm margins…)?
- Clarity (terminology, understandability to other stakeholders, bits'n' bytes…)?
- Precision (internal state, messages, computations, assumptions…)?
- Best practices (standards, 'building codes', APIs, guidelines…)
- Resilience (attack detectability, recoverability…)?

Culture of Safety: Aviation > Aerodynamics

Fatalities per 100M passenger miles for scheduled service; excl. "unlawful interference" and USSR



What can we can do?

1. Focus on outcomes
2. Build better foundations

Can we make foundations that can bear the security “pressure”?

Lowest layer = Crypto Algorithms

Well-understood – hopefully boring*

- Cipher
- Hash/MAC
- Sign/verify
- Key agreement
- Secret sharing/threshold

* Quantum resistance = not as boring as I'd like [...though no sign of qubit scaling]

Basic Crypto Algorithms

✓ Solved

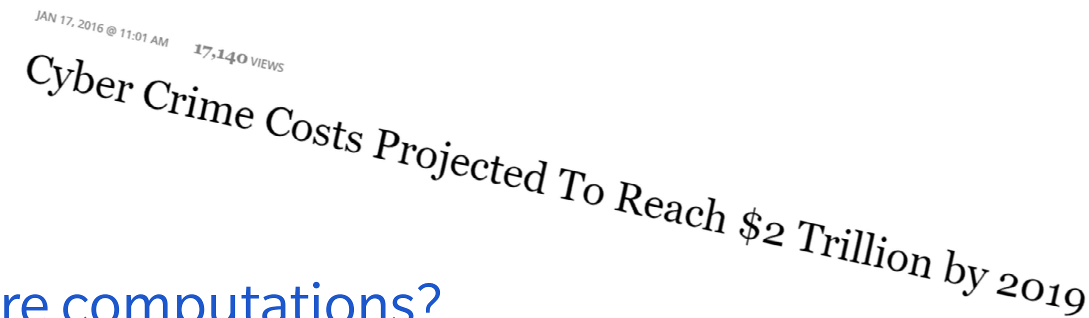
Protocols are well understood – in theory

- Real-world is messy
  - Interoperability between versions, implementations, algorithms (ECC curve proliferation is a mess)…
  - Export rules, regulations, standards process politics, “pride” algorithms…
  - Certificate syntax (X.509 is a mess), contents, parsing, revocation…
  - Performance optimizations for round trips, specific hardware capabilities
  - Certification authority economics & capabilities, manufacturing systems…
  - Denial of service, side channels, fault attacks, implementation complexity, attack surface area…
- 20+ years: Do we understand the SSL/TLS protocol family yet?

Protocols & Constructions — ! Big progress

Basic Crypto Algorithms — ✓ Solved

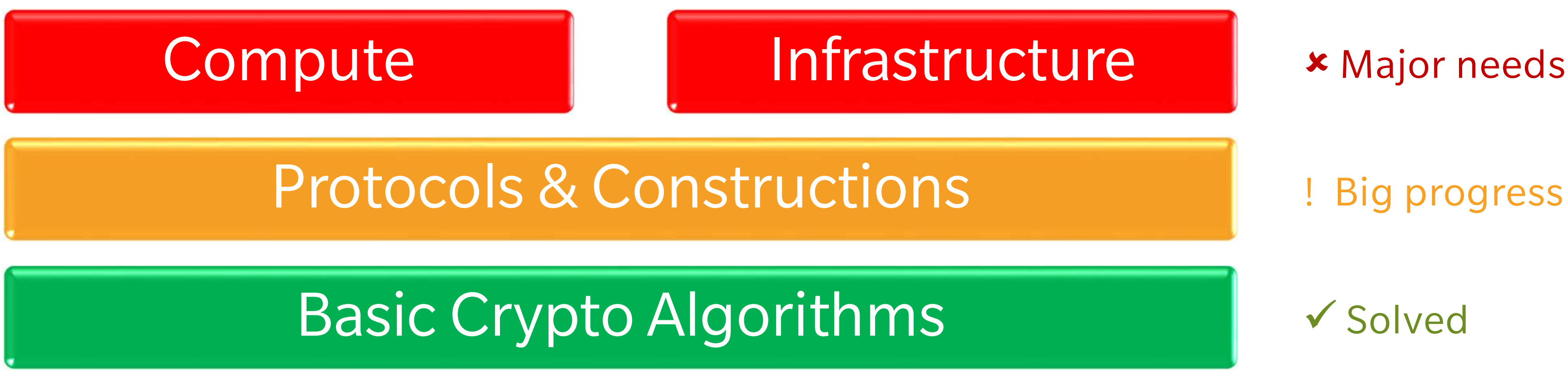
The $2T Question



JAN 17, 2016 @ 11:01 AM 17,140 VIEWS

Cyber Crime Costs Projected To Reach $2 Trillion by 2019

- How can we enable secure computations?
  - Pre-requisite for applications of crypto
  - Massive failures for even simple use cases (e.g. bitcoin wallets)



Compute – Miracle solutions?

Miracle primitives
(fast FHE, obfuscation…)

⇩

Still need secure compute
+ Lots more buggy code

Miracle:
People find the last bug in Windows, Linux, Intel Core i7, OpenSSL

⇩

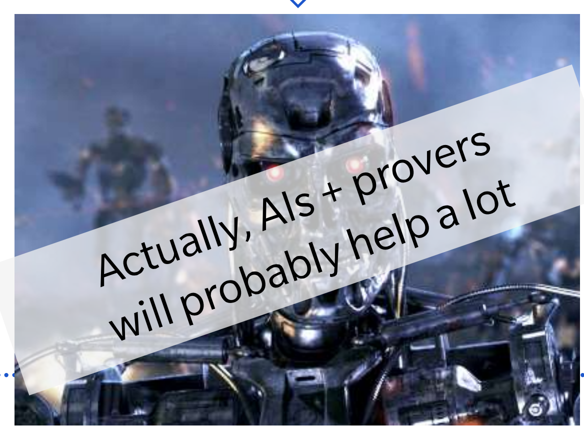
Product is obsolete
New bugs get added

Miracle:
Artificial Intelligence that can find all bugs

⇩

Singularity?

⇩



Compute – Approaches

Grow in a single security perimeter

Traditional approach for security enhancements in CPUs, OSes…

Failure is likely + catastrophic

Add additional partitions

Many small security perimeters, e.g. for each use case

Small, survivable failures

Little bits of security

Legacy platforms (CPUs, OSes, TEEs…) are

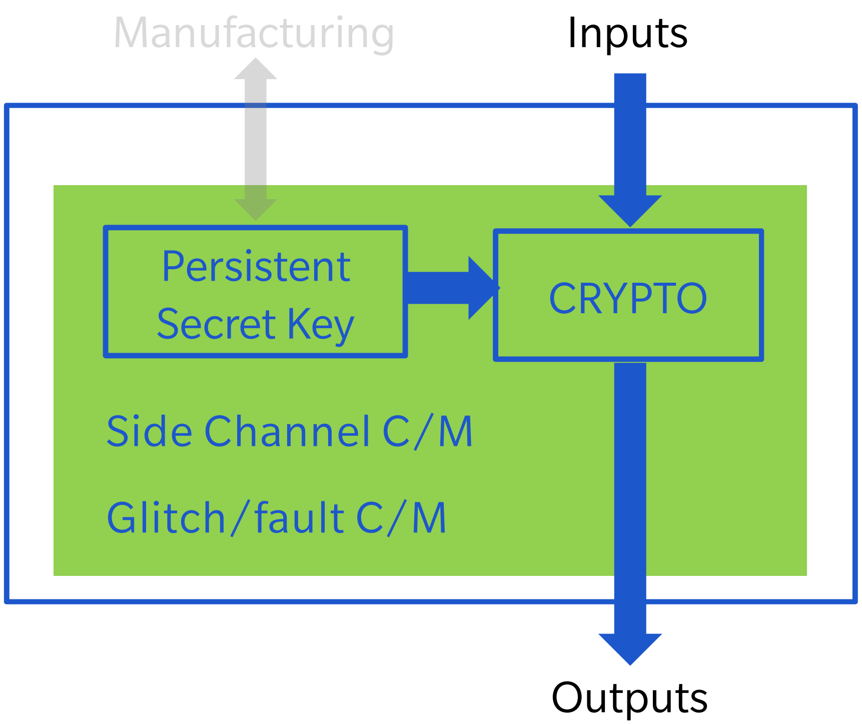
too complex to debug

too valuable to abandon

(Only?) solution:

- On-chip hardware that doesn't trust main CPU/OS/software
  - Intra-chip security perimeter
  - Hardware is unique: Security won't be ruined by a lower layer
  - Moore's Law helps (cheap transistors)
- Separate scaling: security complexity <<< system complexity

Minimal crypto core



How to best build circuits like this?

- What goes in "CRYPTO"?
- Redundancy?
- Algorithm-level SCA?
- Canary/anti-glitch?

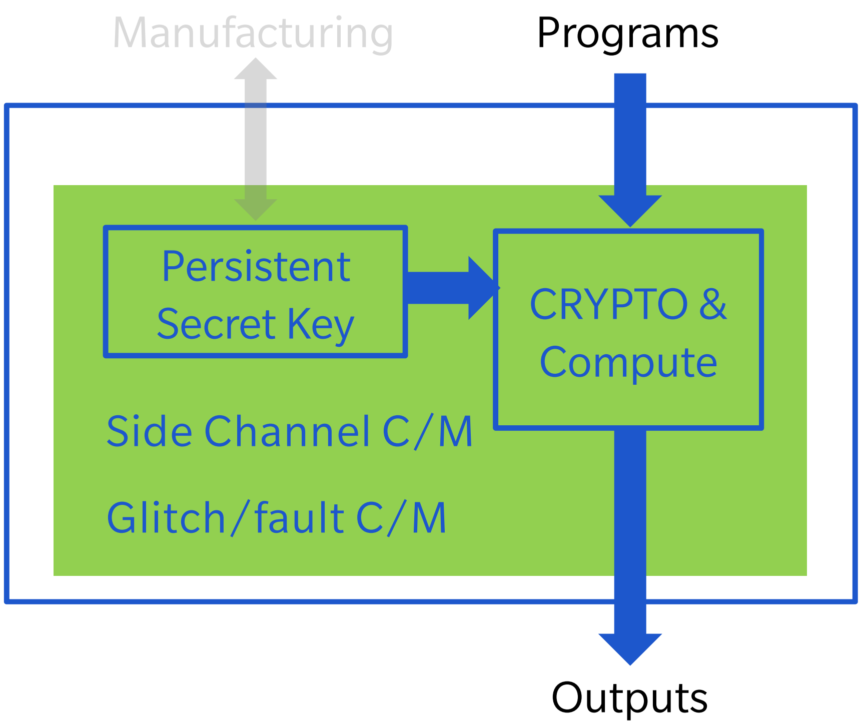
P(fail vs. noninvasive attack) = ?

P(fail vs. invasive attack) = ???????

In-field results seem mostly good...

- My team's CryptoFirewall & CryptoManager cores, DPA-resistant cores/libraries

Crypto-based secure execution



What should this look like?

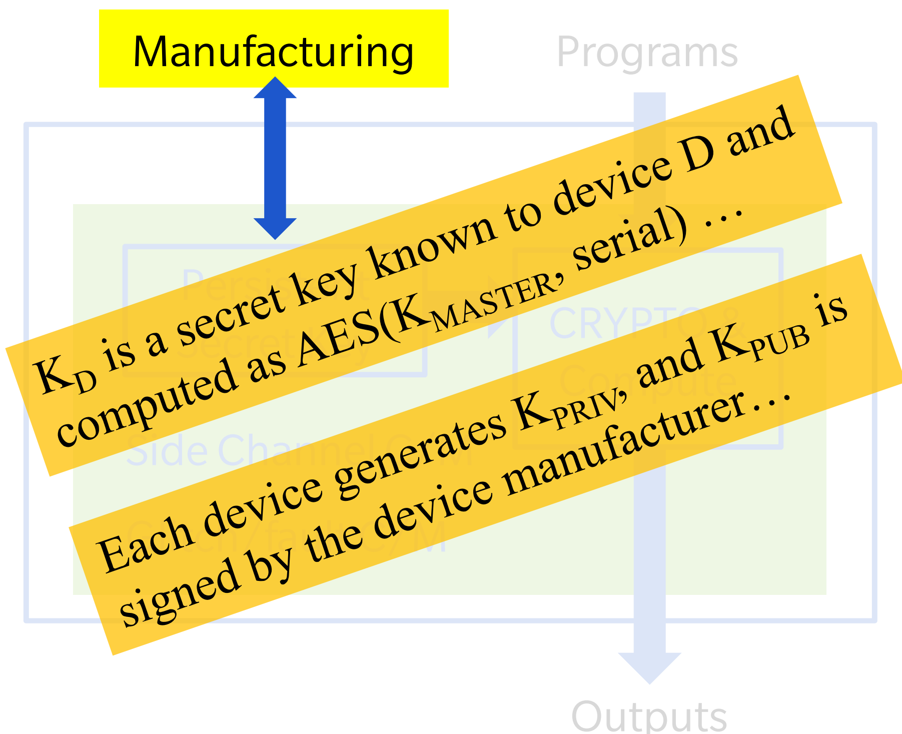
- CPU? FPGA? FSM? SGX-like mode? Something new?
- Include RAM, storage, UI, network…?
- Non-hierarchical trust models?

Lots of crypto problems to solve

P(fail) = ?

- P(bitcoins stolen)?
- P(SSL private key exposed)?
- …

Plumbing (manufacturing, programming, test…)



Historically neglected critical 'plumbing'

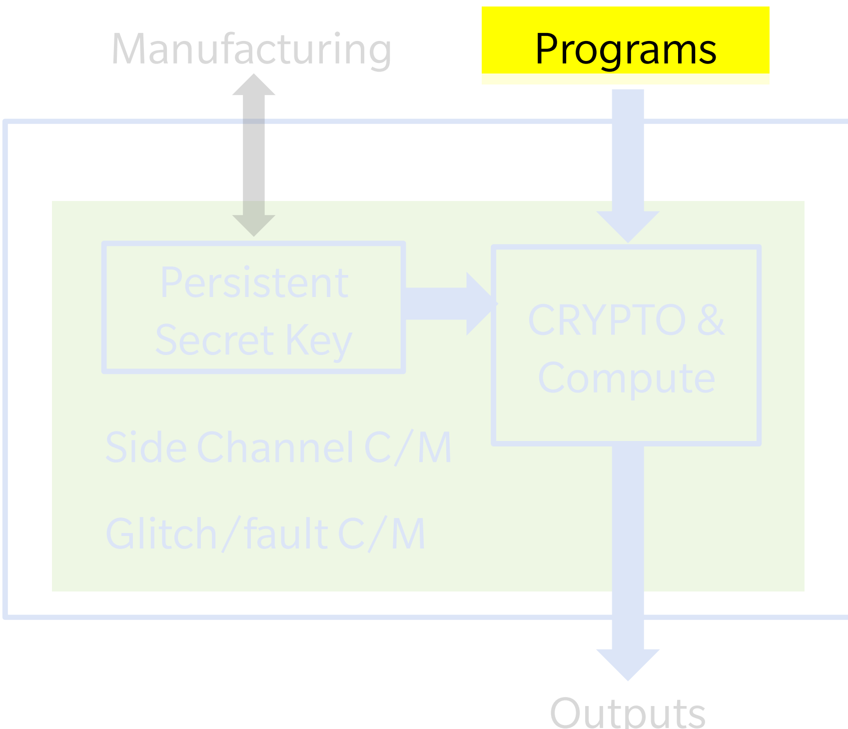
- many keys
- many product types
- many component vendors
- many protocols & use cases
- many security requirements

Cannot grow factory costs, downtime

Back-end is lots of work

- Factory, data center…
- Largest area of R&D spend for our CryptoManager business

Crypto-based secure execution

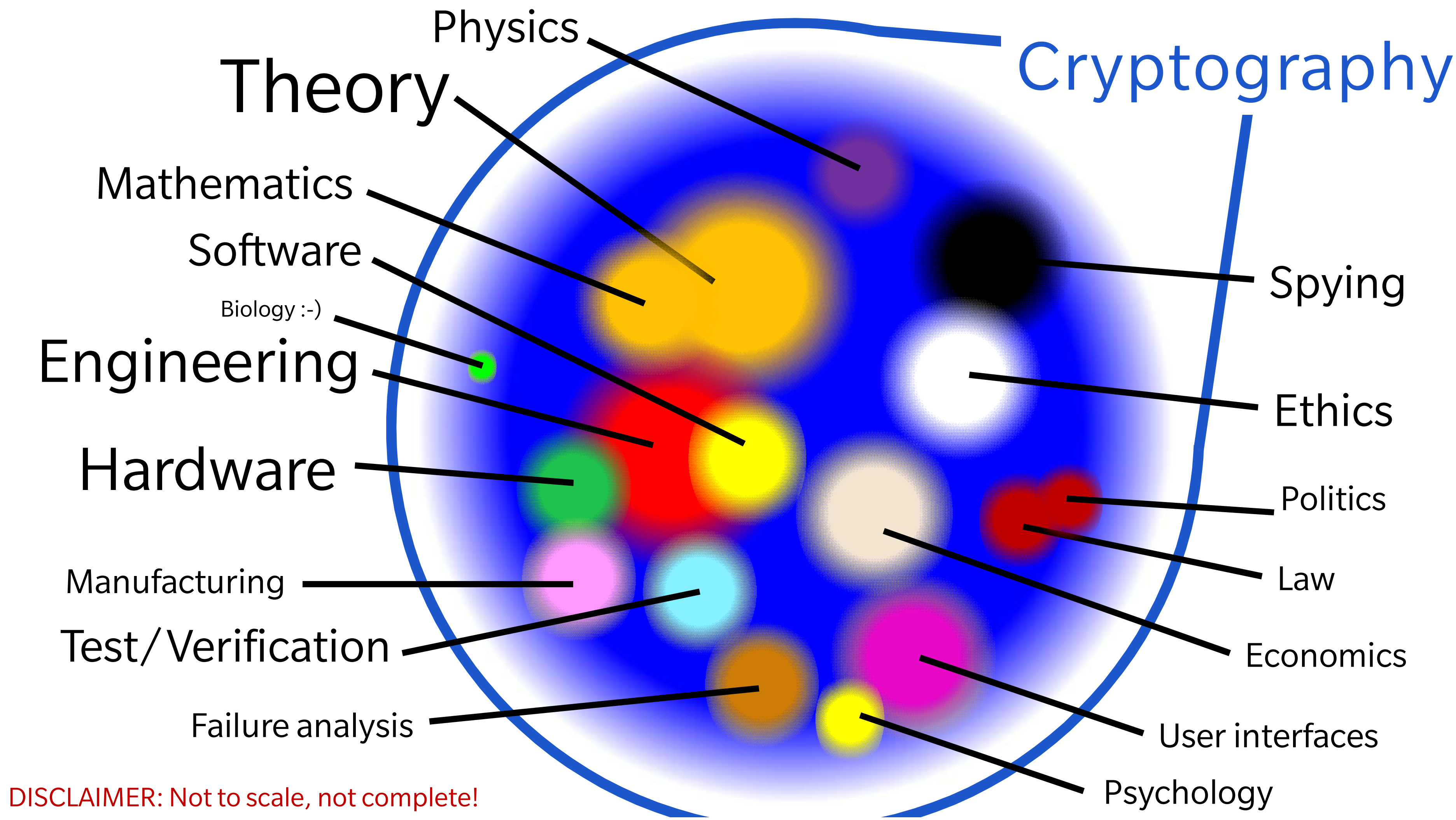


Good buildings > strong foundations

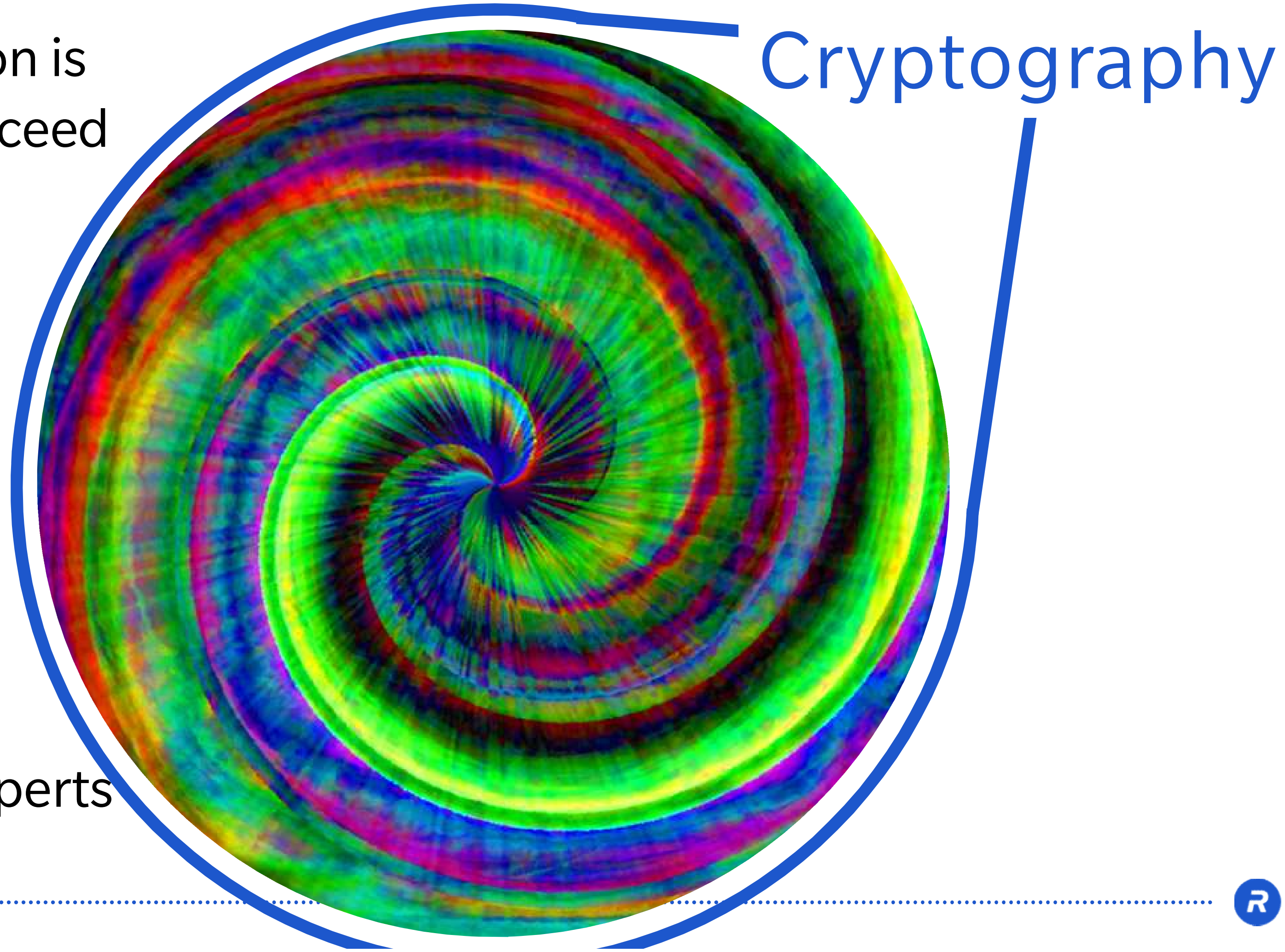
Dreaming…

- What programs will we write?
- What new problems will arise?

Dreams of advanced surgeries are irrelevant without basic sanitation

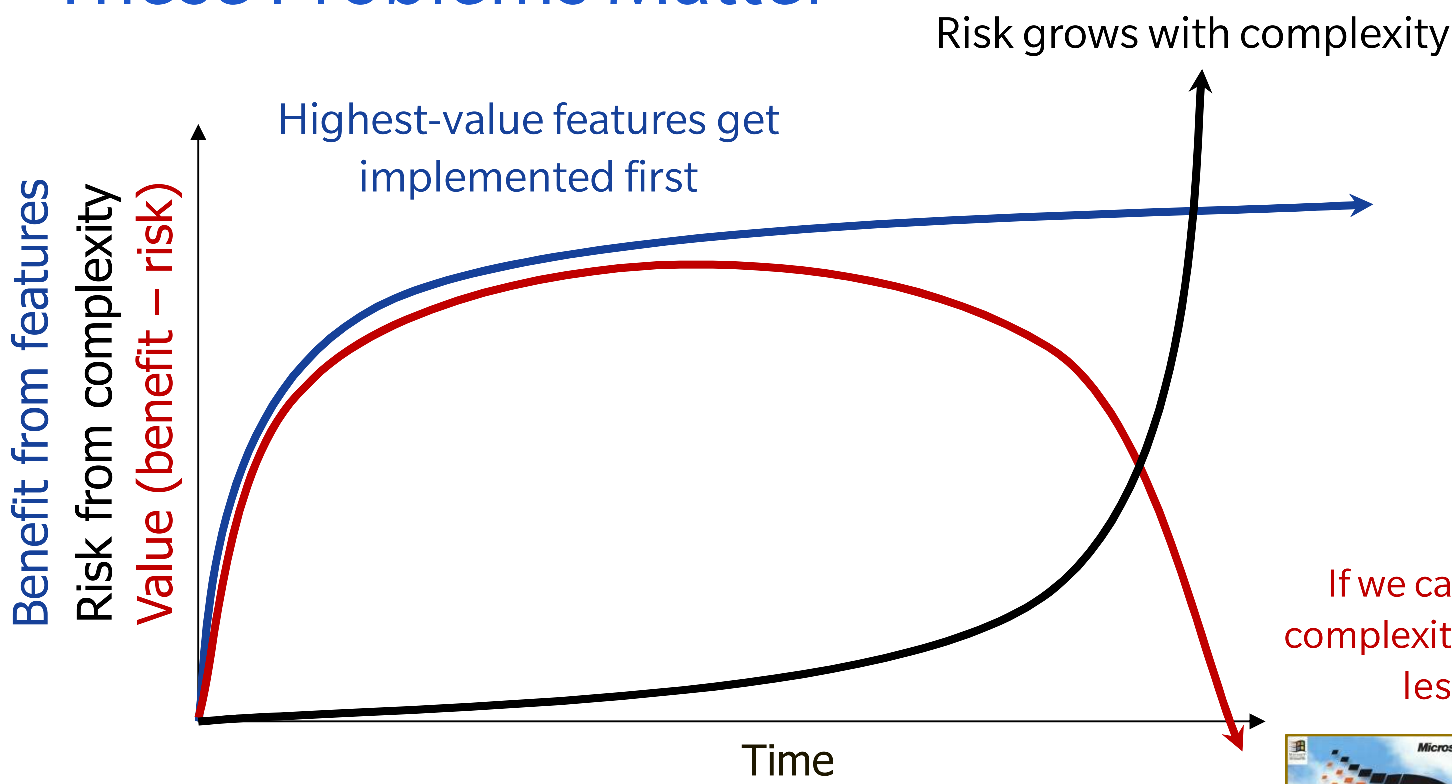

Cryptography
Physics
Theory
Mathematics
Software
Biology :-)
Engineering
Hardware
Manufacturing
Test/Verification
Failure analysis
Spying
Ethics
Politics
Law
Economics
User interfaces
Psychology
DISCLAIMER: Not to scale, not complete!

Crypto's expansion is more likely to succeed than other fields subsuming crypto.



Call to action
Discuss our problems with experts from other fields

These Problems Matter



If we can't control risk, complexity makes products less valuable

+

Looking Ahead

- Macro trend of worsening will continue for 3-5 years minimum
  - Individual designs may fare much better/worse
- Technology industry's future depends on finding solutions
  - Otherwise, security risks will erase society's benefits from new technology
- Cryptography = a very broad & wonderful set of problems

Thank You

For slides, questions, or thoughts:
paul@cryptography.com

Rambus CRYPTOGRAPHY RESEARCH