

Reflections on Trusting Trust

To what extent should one trust a statement that a program is free of Trojan horses? Perhaps it is more important to trust the people who wrote the software.

KEN THOMPSON







INTRODUCTION

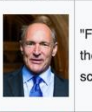


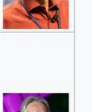




I thank the ACM for this award. I can't help but feel that I am receiving this honor for timing and serendipity as much as technical merit. UNIX¹ swept into popularity with an industry-wide change from central mainframes to autonomous minis. I suspect that Daniel Boh-

programs. I would like to present to you the cutest program I ever wrote. I will do this in three stages and try to bring it together at the end.

STAGE I

https://en.wikipedia.org/wiki/Turing_Award

1983	Ken Thompson		"For their development of generic operating systems theory and specifically for the implementation of the UNIX operating system" ^{[53][54]}
	Dennis Ritchie		
1984	Niklaus Wirth		"For developing a sequence of innovative computer languages, EULER , ALGOL-W , Pascal , MODULA and Oberon " ^[55]
1985	Richard M. Karp		"For his continuing contributions to the theory of algorithms including the development of efficient algorithms for network flow and other combinatorial optimization problems, the identification of polynomial-time computability with the intuitive notion of algorithmic efficiency, and, most notably, contributions to the theory of NP-completeness " ^[56]
1986	John Hopcroft		"For fundamental achievements in the design and analysis of algorithms and data structures" ^{[57][58]}
	Robert Tarjan		
1987	John Cocke	—	"For significant contributions in the design and theory of compilers , the architecture of large systems and the development of reduced instruction set computers (RISC)" ^[59]

2016	Tim Berners-Lee		"For inventing the World Wide Web , the first web browser , and the fundamental protocols and algorithms allowing the Web to scale" ^[107]
2017	John L. Hennessy		"For pioneering a systematic, quantitative approach to the design and evaluation of computer architectures with enduring impact on the microprocessor industry" ^{[108][109][110]}
	David Patterson		
2018	Yoshua Bengio		"For conceptual and engineering breakthroughs that have made deep neural networks a critical component of computing" ^{[111][112][113][114]}
	Geoffrey Hinton		
	Yann LeCun		
2019	Edwin Catmull		"For fundamental contributions to 3-D computer graphics , and the revolutionary impact of these techniques on computer-generated imagery (CGI) in filmmaking and other applications" ^{[115][116][117]}
	Pat Hanrahan		

Take-Away Message:

Attack Surface has to be small enough to audit

Ken's Example

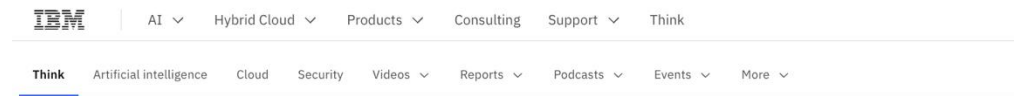
- Figure 3.2 shows a simple modification to the compiler
 - that will deliberately miscompile source
 - whenever a particular pattern is matched.
- If this were not deliberate,
 - it would be called a compiler "bug."
- Since it is deliberate,
 - it should be called a "Trojan horse."

MORAL

- You can't trust code that you did not totally create yourself.
- (Especially code from companies that employ people like me.)
- No amount of source-level verification or scrutiny
 - will protect you from
 - using untrusted code

Purpose of this talk

- Reading Group
- Summarize main point of paper
- As well as implications for contemporary audience
 - Current Reality:
 - Attack surface on phones is too large to audit
 - (and it will become larger in future)



What is an attack surface?

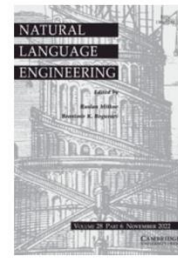


What is an attack surface?

Digital attack surface

What is an attack surface?

State of the Art in 1983



Emerging trends: Ethics, intimidation, and the Cold War

Published online by Cambridge University Press: 31 May 2021

Kenneth Ward Church and Valia Kordoni

Show author details

Article

Metrics

Save PDF

Share

Cite

Rights & Permissions

- Security was less of a topic in 1983 than these days
- When I was a grad student at MIT in 1970s
 - Computers in the MIT AI Lab had no security, by design.
 - Anyone could watch anyone do anything.
 - This was considered a feature, not a bug.
 - We were proud of the small town atmosphere.
 - Everyone knew everything about everyone.
 - When we were forced to add passwords,
 - Stallman (GNU) inserted a ``bug''
 - that he just had to show me one night
 - His bug printed passwords on system terminal (hardcopy)
 - So everyone could see everyone's passwords

Spying on Rivest (R in RSA)

- While the field was still small,
 - Lack of security → interesting consequences for cryptography
- A few years after RSA,
 - we crowded around a computer screen,
 - spying on Rivest as he was working on a subsequent paper,
 - suggesting that NSA had designed key lengths
 - for a proposed standard to be just short enough
 - so they could break the code and no one else could.

A Conspiracy Theory

- Some of the others in the crowd probably
 - knew more about the key length controversy than we did.
- Normally, these people did not spy on others,
 - but they encouraged us to spy on Rivest at just the right time.
- They may have been working with Rivest
 - to leak the story with plausible deniability.

Flame Wars (circa late 1970s)

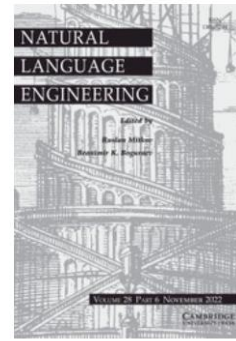
- While no one knows
 - whether Rivest actually took advantage of
 - the lack of security to leak the story,
 - there is no doubt that others have used
 - the lack of security in creative ways.
- For example, Minsky knew that a friend of his,
 - the head of the MIT AI lab,
 - was the only one in the lab
 - that did not read email sent to the head of the lab.
- Minsky used this feature to send a flame to his friend,
 - knowing that his friend would not see the flame,
 - but others would (and did).

Security Today

- Zero click exploits
- Cheaper than divorce lawyer
- Weapon of Math Destruction



- Nobody but US → Bug Market



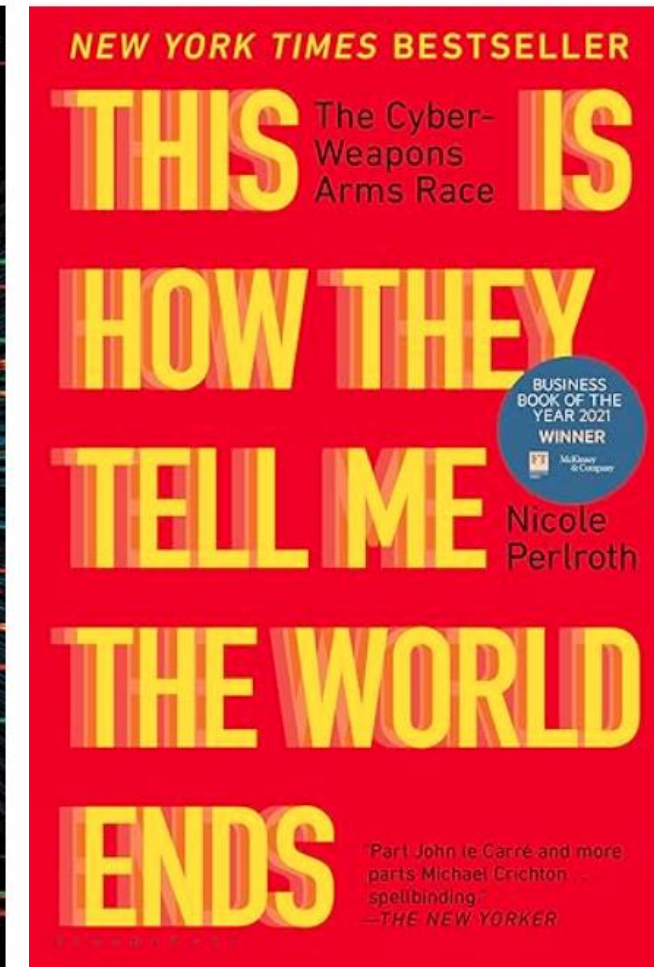
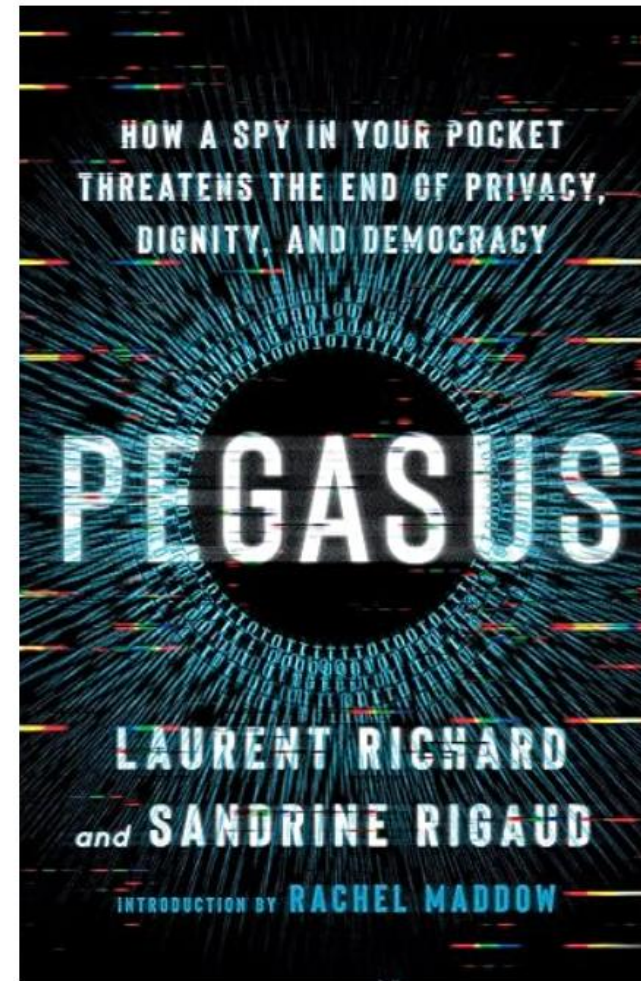
Emerging trends: Risks 3.0 and proliferation of spyware to 50,000 cell phones

Published online by Cambridge University Press: 19 May 2023

[Kenneth Ward Church](#)  and [Raman Chandrasekar](#) 

[Show author details](#) ▾

Article Metrics



https://en.wikipedia.org/wiki/Zero_Days

- Stuxnet
- NOBUS: Nobody but us
 - <https://en.wikipedia.org/wiki/NOBUS>
 - There was a time
 - when cyber warfare favored us
- But now that bugs are cheap
 - and attack surfaces are big (and growing)
- Cyber favors small countries
 - Terrorists, mobsters, anyone with a beef
 - Cheaper than a fighter jet
 - Cheaper than a divorce lawyer
- Big countries have more targets

