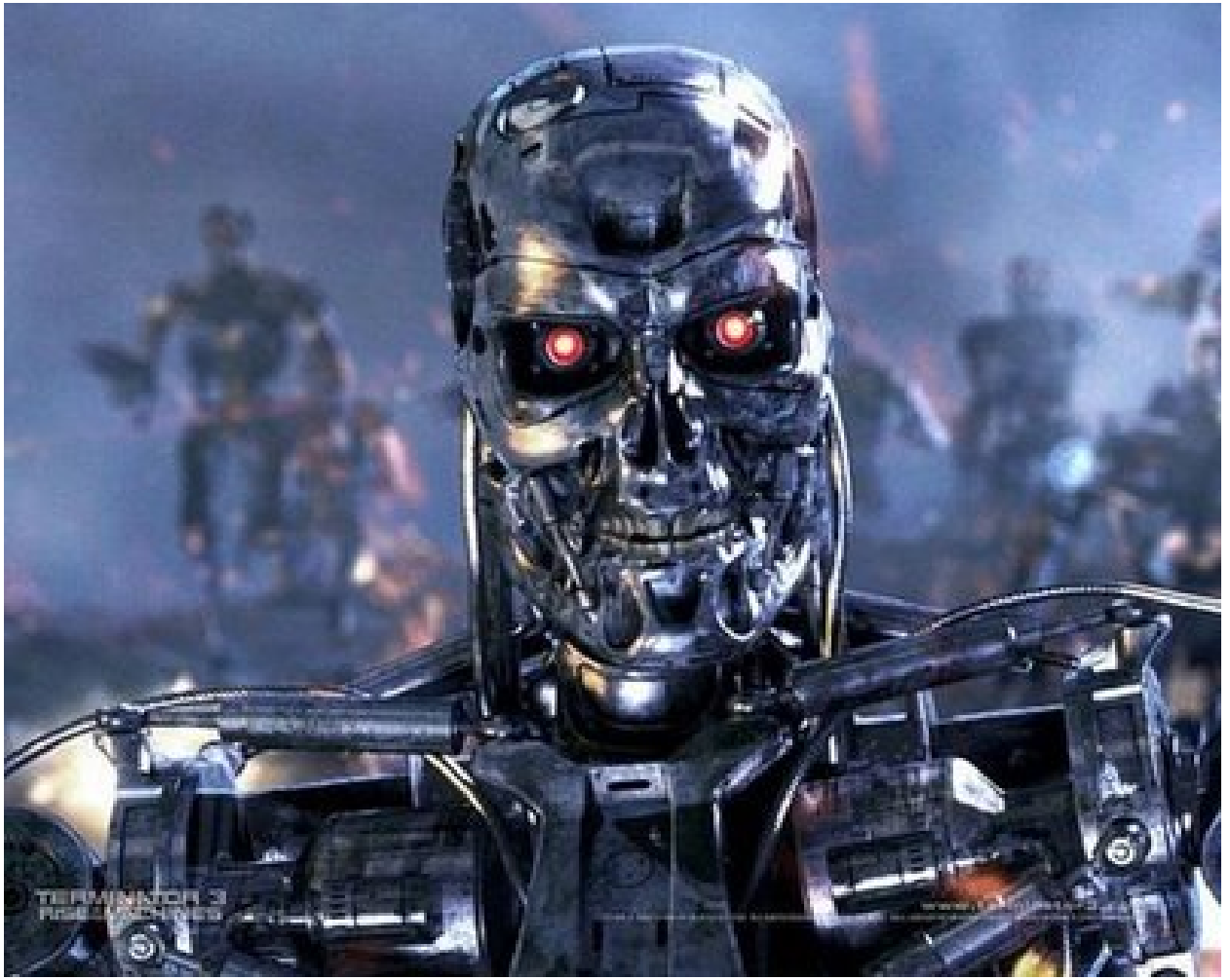
What are the most critical **Grand Challenges** in distributed systems security?

some ideas...

Bryan Ford – NSF EDS Workshop
January 21, 2015

Which unsolved problems will turn our distributed systems into SkyNet?



A few directions for thought

- Decentralizing trust in distributed systems
- Securing the metadata as well as the data
- Software verification for dummies
- Securing the Internet of [Insecure] Things
- Remembering and forgetting distributed history
- Countering *shiny, insecure* approaches?

Decentralizing Trust

- How do we systematically split trust?
  - Many *tools* available: secret sharing, SMPC, …
  - How to deploy them securely and pervasively?
  - Avoiding “worst-of-all-worlds” security scenarios
    - e.g., TLS CA hierarchy: *everyone has a master key*
  - We *still* can't build a scalable, *decentralized* DHT!

Decentralizing Trust

- Most interesting recent near-miss: **BitCoin**
  - Highly decentralized by design/architecture
  - Incentives evolved it back into centralized oligarchy

Protecting Metadata

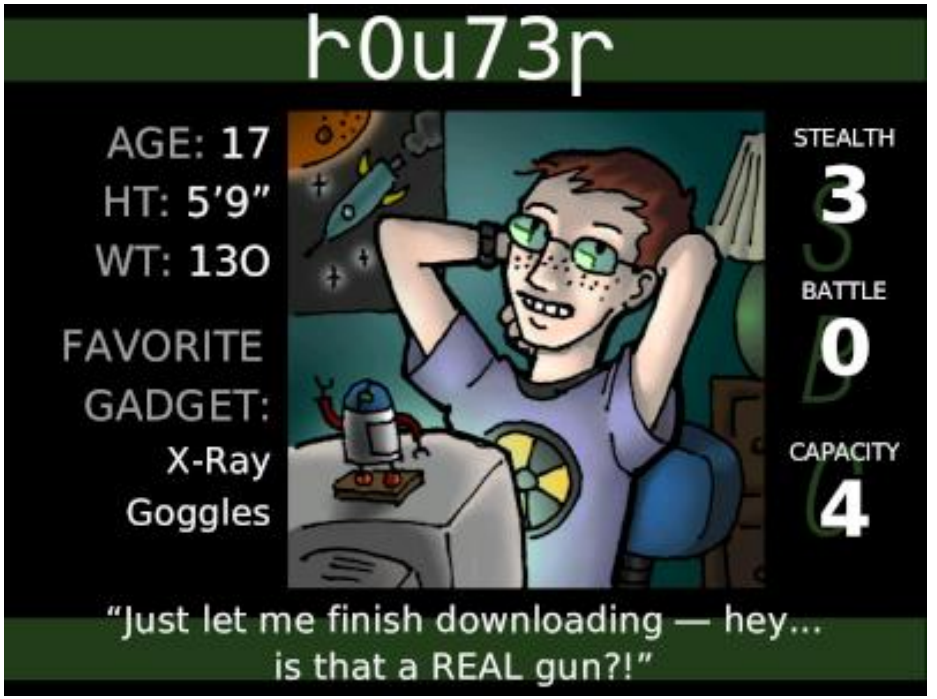
- Protecting access patterns (e.g., ORAM)
- Protecting identity (anonymity, pseudonymity)
  - Is “dataset anonymization” dead?
    What do we replace it with?
- Protecting systematically against side-channels
  - Key recent lesson: the most harmless-looking side-channels turn into attack vectors
- How do we build extensible distributed systems that *systematically protect metadata?*

“We kill people based on metadata” - Michael Hayden

Software Verification for Dummies

Bringing verification practice from here...

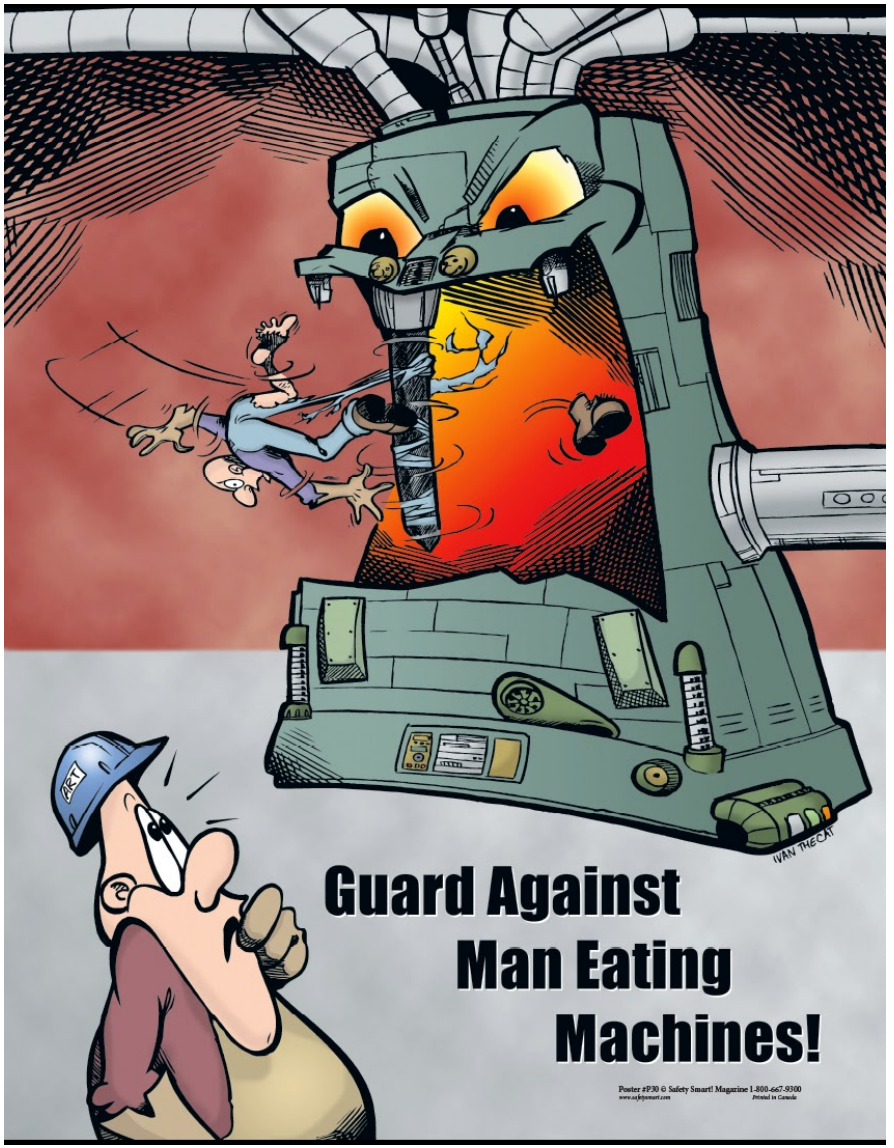
...to here



The Internet of Insecure Things

- Every“thing” will soon have Internet connection
- Business model for “things” is support-averse
  - Minimize post-sale customer support calls
  - Minimize post-sale development, maintenance
  - “Sell-and-forget”
- Can we rethink the architecture of the IoT to avoid a race-to-the-bottom in home security?
  - Mechanisms to separate design/sale responsibilities from responsibility for security maintenance?
  - Could “Home SDN” help secure the IoT?

The Internet of Insecure Things



Preservation in Distributed Systems

- Digital preservation vs “right to be forgotten”
  - Remembering in distributed systems
  - Forgetting in distributed systems
  - Deciding which to do
- Preservation is a distributed systems security problem, even a *national security* problem
  - “**We just downed a plane, an AN-26.**”
    - Igor Girkin, Ukrainian separatist, on Vkontakte
    (up for a few hours, now only on Internet Archive)

THE NEW YORKER

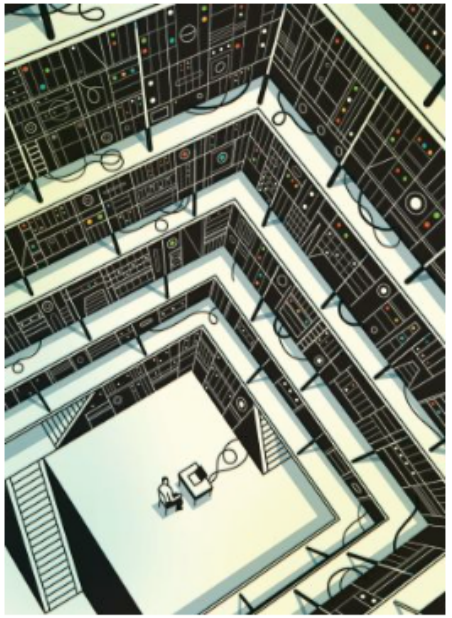
ANNALS OF TECHNOLOGY | JANUARY 26, 2015 ISSUE

THE COBWEB

Can the Internet be archived?

BY JILL LEPORE

Malaysia Airlines Flight 17 took off from Amsterdam at 10:31 A.M. G.M.T. on July 17, 2014, for a twelve-hour flight to Kuala Lumpur. Not much more than three hours later, the plane, a Boeing 777, crashed in a field outside Donetsk, Ukraine. All two hundred and ninety-eight people on board were killed. The plane's last radio contact was at 1:20 P.M. G.M.T. At 2:50 P.M. G.M.T., Igor Girkin, a Ukrainian



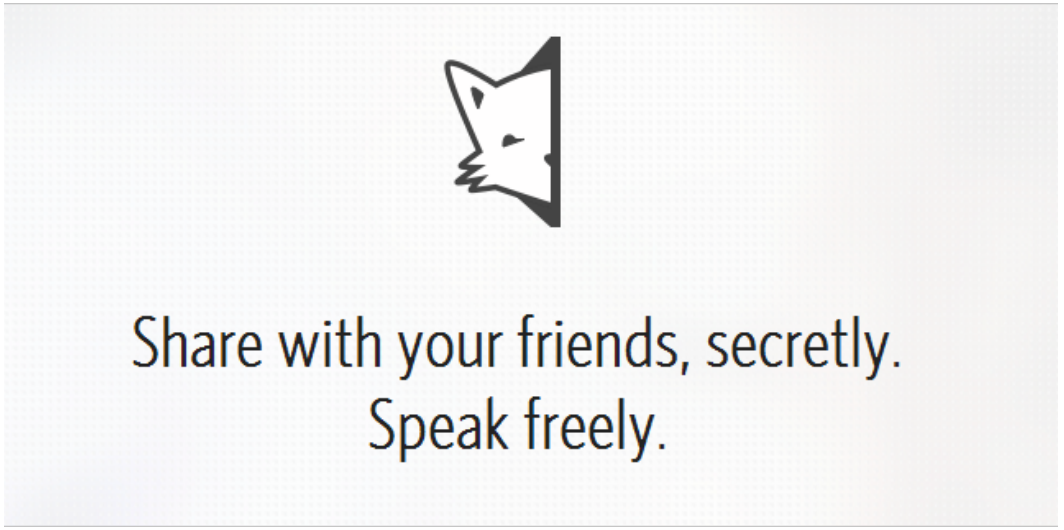
The Web wasn't built to preserve its past; the Wayback Machine aims to remedy that.

ILLUSTRATION BY HARRY CAMPBELL

The Next Library of Alexandria?

Shiny [In]security Solutions

Apps described as “100% Secure and Private”



Do we need ubiquitous *distributed reputation systems for software security?*

Are the Crypto Wars back?

“In extremis, it has been possible to read someone’s letter, to listen to someone’s call, to listen in on mobile communications. The question remains: are we going to allow a means of communications where it simply is not possible to do that? My answer to that question is: no, we must not.” - David Cameron, UK

A few directions for thought

- Decentralizing trust in distributed systems
- Securing the metadata as well as the data
- Software verification for dummies
- Securing the Internet of [Insecure] Things
- Remembering and forgetting distributed history
- Countering *shiny, insecure* approaches?