Critical Infrastructure, Software Engineering, & Complexity

From Dagstuhl

Randy Bush <randy@psg.com>

with the help of

Matt Roughan <matthew.roughan@adelaide.edu.au>

Critical Information Infrastructure

What the Heck is Critical Internet Infrastructure Anyway?

Framing from ENISA

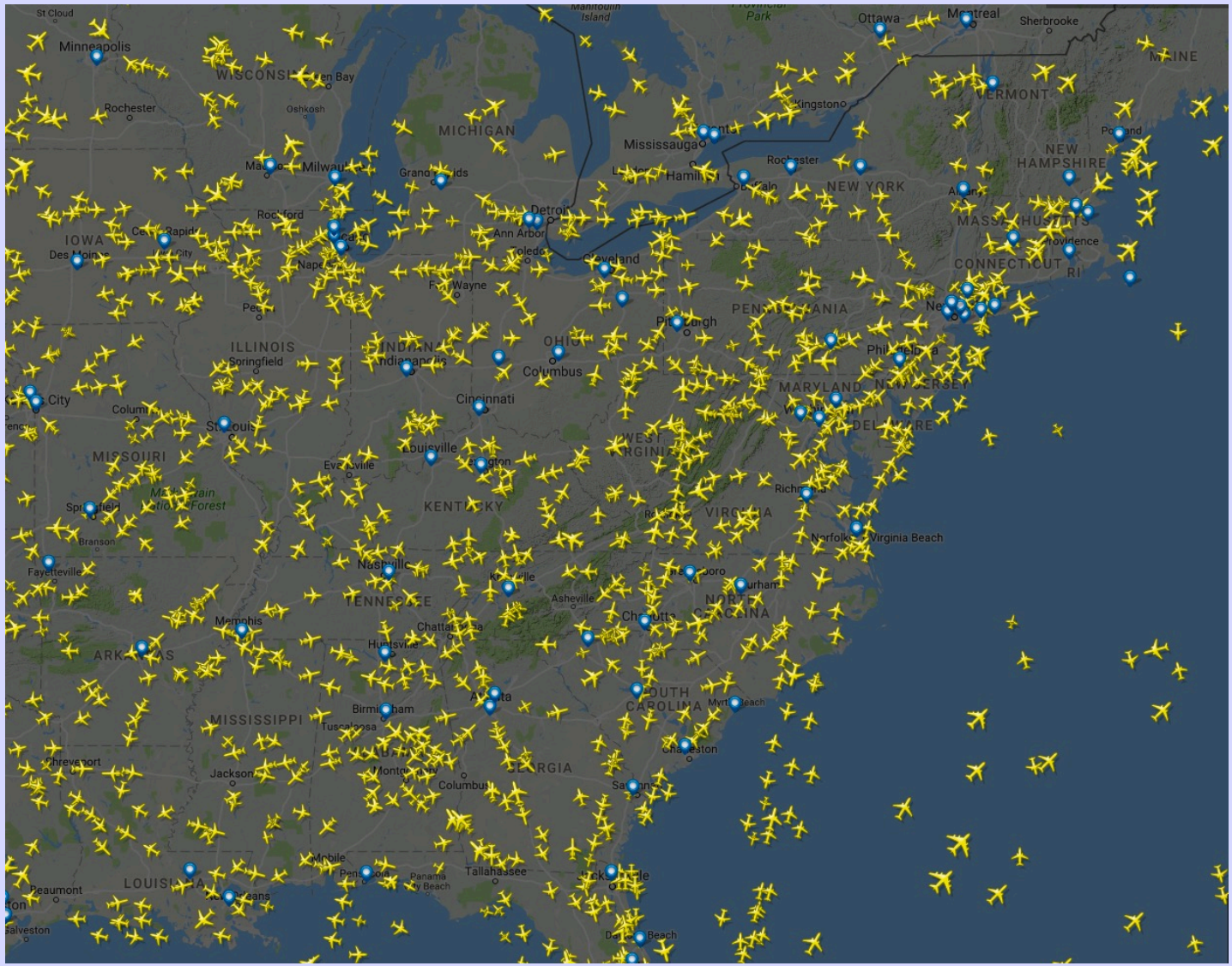
- **Critical Infrastructure**: an asset, system or part thereof located in Member States that is essential for the maintenance of vital societal functions, health, safety, security, economic or social well-being of people, and the disruption or destruction of which would have a significant impact on a Member State as a result of the failure to maintain those functions.
- **Critical Information Infrastructure**: Information infrastructure (like networks, hardware, software, etc.) that is critical to the functioning of a nation or country, like IT that supports health- or energy-sectors.

Why Systems Fail

- Vulnerabilities vs Operational Gaps

- If it was part of the "plan" it's an event, if it is not then it's a "disaster"

Normal US East Coast



A Poorly Tested Patch to the US Air Control

Flypocalypse

Just a Quick Patch

Device and System Vulnerabilities

- Protection of routers and switches from direct attack are common practice, but less commonly practiced than it should be
- Where is fiber? The PoPs? DataCenters?
- Wireless devices and infrastructure have some unique vulnerabilities.
- Human error and weak operational practice are major issues

Why Systems Fail

- Most security holes are due to buggy software.
  - As of 1998, 85% of CERT advisories described problems not fixable with cryptography.
  - About half of all new holes are due to buffer overflows.
- "Patch and pray" is no way to run an Internet.
- Patches are often hard to install, and can cause their own problems (and holes).
  - *No* responsible administrator of a production machine will install *any* patch without extensive testing.

Defense Strategies

- We're not going to get rid of buggy code.
  - We've been trying for far too long to have any realistic hope of success now.
- We can't do much about lack of cyber diversity – the "network effect" is too strong.
- We can try to reduce central points of failure.
- We must learn how to compose secure components, and how to build secure distributed systems out of insecure pieces.

Major Control Systems

Non-linear interactions of these can produce seriously disastrous results

- Routing
- MPLS Control Plane: LDP and RSVP
- DNS
- RPKI (and NTP)
- CDNs shifting traffic
- Peer to Peer Traffic Engineering
- Automated Traffic Engineerng
- Reactive Configuraton of Network
- OpenFlow
- TCP Congestion Mechanisms
- Interaction with Human Behavior (e.g. redialing)

Fat Finger Friday

The first Friday of each month, we take out one control system

- We will learn how to take it out, i.e. Vulnerability Analysis
- Minimal interventions to cause maximal affect
- We will pre-announce, so the world will think about defenses
- We can measure and analyse

Resilience Mechanisms

- Metric: correlation in spikes in help desk calls
  - AMAZON: rate of sales drop
  - Airlines – almost accidents investigated
- Airlines investigate all non-critical events
  - Maybe partial causes
  - Correlated events are perhaps the problem – interactions
    - Investigate subcritical because want to avoid the correlated cases
- What other systems should we look at
  - Power?
  - Old telcos?

Resilience Mechanisms Can Be Our Enemy

- Resilience mechanisms are designed with particular failure modes in mind. When circumstances fall outside those boundaries, their [re]actions can interact with control systems in unanticipated ways.

- E.g. SONET restoration under Layer Three healing under CDN traffic shifting.

Data Applicability

- Ambiguity
- Accuracy = Resolution, Precision, and Validity
- Inconsistency
- Missing Data and Bias
- Missing Meta-Data
- Issues of duration of measurement and shifts in what one is measuring.
- Issues with resolution of results over sample size/rate.
- Issues of representation of sample set, e.g. Route Views sees the 'clue core'.
- Be sure to understand how the data were collected

Software Engineering

- Formal Methods would be useful
- In devices
- In protocols
- Software Engineering is rare in the hardware vendor culture
- It is starting to be exercised in the OS and Applications Vendors

Complexity - the Enemy of Analysis, Reliability, Repairability, Scalability

Everything, Unless You are Paid by the Hour

But Where is CII?

- Let's assume that public agencies such as ENISA can easily identify Critical Infrastructure
- How do they figure out how it connects to the Internet so they can identify Critical Internet Infrastructure?
- And how do they discover inter-ISP connectivity?

Topology is Hard

- Maybe Critical Infrastructure does not want to disclose connectivity as it may make them more vulnerable
- Providers view interconnection as NDA
- Research into Internet topology is primitive and error prone
- Public data are weak despite braggadocio

"10 Lessons from 10 Years of Measuring and Modeling the Internet's Autonomous Systems"

Matthew Roughan, Walter Willinger, Olaf Maennel, Debbie Perouli, and Randy Bush

IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 29, NO. 9, OCTOBER 2011

And Then What?

- Are we going to regulate how Critical Infrastructure connects to the Internet?
- Are we going to regulate how Internet providers inter-connect?
- Are we going to regulate a provider's infrastructure?