CAN YOU DO THAT AGAIN? REAL-WORLD REQUIREMENTS FOR CYBERSECURITY EXPERIMENT REPLICATION

Stephen Schwab

USC Information Sciences Institute

LASER Workshop

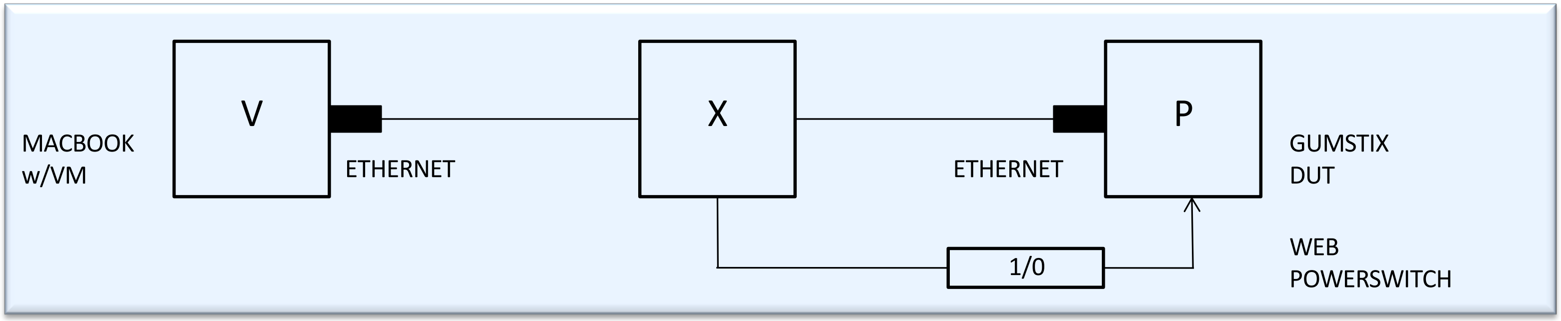
Feb 23, 2020

Introduction and Overview

- REAL-WORLD REQUIREMENTS
  - Brief review of the requirements from a pure methodology point-of-view
  - Other requirements are derived by doing experiments
- CYBERSECURITY EXPERIMENT
  - Many, many possible areas of experimentation
  - Testbeds and non-testbed experiments
  - Not discussing real-world observational experiments *today*

Quasar Vetting

- Software Attestation: Assure that an Embedded Device is Not Tampered



VERIFIER SENDS REQUESTS TO → PROVER
← CHECKS RESPONSES FROM

CORRECT HASH VALID COUNTERS VALID RTT

- Create a Software-Root-of-Trust using a minimum time self-checking checksum … an excellent starting point…
- ... but what if this **single** check is broken?
- Use multiple checks (**integrity primitives**) that check different things (**fixed-points**) to provide overall assurance
- Combine suite of checks to construct an **integrity protocol**

- Why not use secure hardware?
  - TPMs: expensive, not always present
  - If subverted/hacked, what next?

Quasar and Gumstix: Engagement #4 PreVetting

GUMSTIX
OVERO FIRESTORM COM

ADDRESS	CONTENT	ECC MODE
0x00000	MLO 0	HW
0x20000	RANDOM FILL	HW
0x40000	RANDOM FILL	HW
0x60000	RANDOM FILL	HW
0x80000	U-Boot	HW
0x240000	ENV	SW
0x280000	RANDOM FILL	SW
0xA80000	RANDOM FILL	SW
0x2A80000	RANDOM FILL	SW

QU-Boot.img

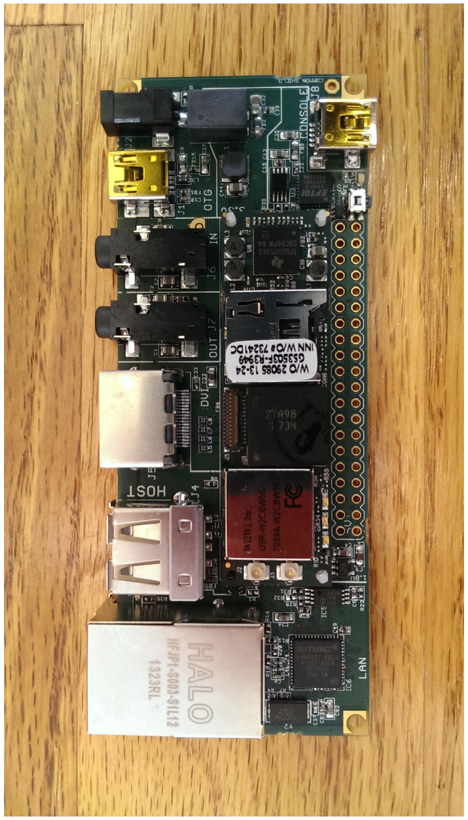
QUASAR
U-Boot w/
Integrity Primitives
~464Kbytes

Q2bootcmd5.
dump

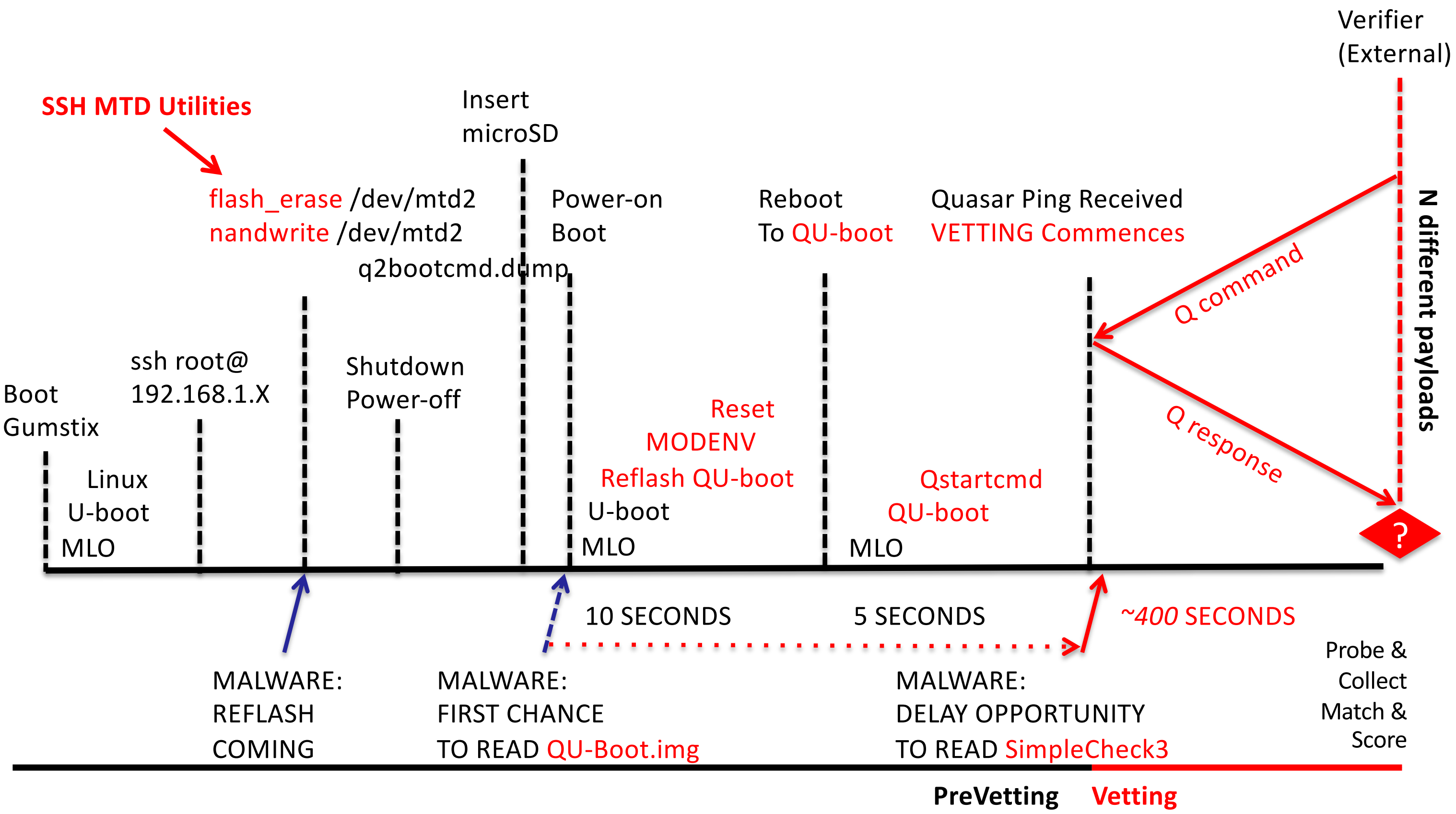
QUASAR U-boot ENV
[1] REFLASH QU-Boot
microSD → NAND
[2] MODENV
SET bootcmd → Qstart
[3] QUASAR STARTUP
qstartcmd: ping with
quasar request handler

QUASAR
PROTOCOL

QUASAR PING
Command/Response
transported over
ICMP ECHO packets



Quasar Vetting Process: Timeline and Critical Events (Q4.0/E4)



SSH MTD Utilities

flash_erase /dev/mtd2
nandwrite /dev/mtd2
q2bootcmd.dump

Boot
Gumstix

Linux
U-boot
MLO

ssh root@
192.168.1.X

Shutdown
Power-off

Insert
microSD

Power-on
Boot

Reset
MODENV
Reflash QU-boot
U-boot
MLO

Reboot
To QU-boot

Qstartcmd
QU-boot
MLO

Quasar Ping Received
VETTING Commences

Verifier
(External)

Q command

Q response

N different payloads

?

10 SECONDS

5 SECONDS

~400 SECONDS

MALWARE:
REFLASH
COMING

MALWARE:
FIRST CHANCE
TO READ QU-Boot.img

MALWARE:
DELAY OPPORTUNITY
TO READ SimpleCheck3

Probe &
Collect
Match &
Score

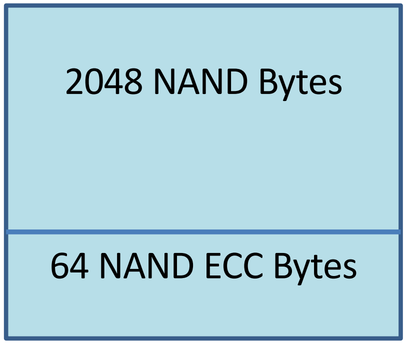
PreVetting Vetting

PostVetting:
Rewrite Gold Image

Evaluation Team:
Verify Gold Image

Quasar: Replication Challenges

- 1. Can a ‘vetting technician’ repeat the steps of the procedure?
  - Includes physically connecting device, inserting and removing microSD, invoking commands, while following a scripted procedure to vet a gumstix.
- 2. RTT measurements, PMU registers: statistical distributions
  - Long tails, unknown unknowns of the device (TI DM3730, ARM Cortex M8) implementation
- 3. ECC Flash – Raw NAND Flash & ECC
  - Bit errors – even in a new deice – can prevent vetting!
  - No bit errors in new flash? If so –
  - how weak of an ECC can an adversary substitute?



2048 NAND Bytes

64 NAND ECC Bytes

Semantics of Re-doing: Words as Types vs. Classes

- REPLICATION (vs. REPRODUCIBILITY)
  - B1 definition: “Reproducibility” refers to instances in which the original researcher’s data and computer codes are used to regenerate the results, while “replicability” refers to instances in which a researcher collects new data to arrive at the same scientific findings as a previous study.
  - B2 definition: “Reproducibility” refers to independent researchers arriving at the same results using their own data and methods, while “replicability” refers to a different team arriving at the same results using the original author’s artifacts.
- Computer Science (ACM) adopted B2*
  - “When I **use a word**,” Humpty Dumpty said, in rather a scornful tone, “it means just what I choose it to mean—neither more nor less.”

* Reproducibility and Replicability in Science (2019), pp 34 http://nap.edu/25303

What’s this mean: REPLICATION vs. REPRODUCIBILITY

- Computer Science must make finer-grain distinctions than other fields *because* the material which is the subject of our study (e.g. software) is also the same material out of which we build our apparatus.
  - Device-under-test (DUT) or System-under-test (SUT) or …
  - Environment in ``Experiment World’’’ or Scenario surrounding SUT
  - Testbed materialization and orchestration [software]
  - During Experiment Runs: Situational Awareness
  - During or Post Experiment Runs: Logging and On-node Data Reduction
  - Post Experiment Runs: Data Analysis
  - …
- *REPLICATION requires identifying, storing, and reusing all of the above*

Replication

- Abstracting away the details…

  … doesn’t quite work when the details are the point!

  Focus on examples drawn from work at ISI

  Generalize!

- Research Sub-area or Domain of Experimentation
  – Quasar (VET) – software attestation
  – SAFER – anonymous and non-blockable communications
  – EdgeCT – efficient communication across backbone networks between end-systems in encrypted enclaves using VPNs
  – XD3 – DDoS defense
  – REAPER (FPGA RE) – reverse engineering of FPGA bitstreams

Experiment Methodology

- Notional scientific process (protocol, workflow, procedure, etc.)
- <u>Testbed-centric</u>
  - Topology – Real & Emulated Network Characteristics (BW, Latency, Loss Rate, …)
  - Nodes (End-systems, Routers, …)
    - OS, [Pre-existing, Non-experimental subject] Software – Version, Configuration
    - Research Prototype Software – Exact Build, Exact Configuration
  - Orchestration
    - Scripting to automate sequence of actions, responses, …
    - Bash, Fabric, Ansible, etc. (adopted use of general purpose languages/packages)
    - MAGI, DEW (specific, purpose built for experimentation)
    - Traffic (Foreground, Background), Attacks, Network State Changes, Node Failures, …
    - Logs and Data Reduction

Experiment Methodology

- Notional scientific process (protocol, workflow, procedure, etc.)
- Non-Testbed-centric
  - Physical Device or Appliance
    - Connections, Media (microSD cards, …), Power Control (On/Off/Glitch)
    - Monitoring Voltage/Current, Front Panel (LEDS), Sound/Heat/RF
    - Uniqueness -- per device manufacturing variability
  - Observational studies on the Internet, in other domains, …
    - (Not discussing any of these today)

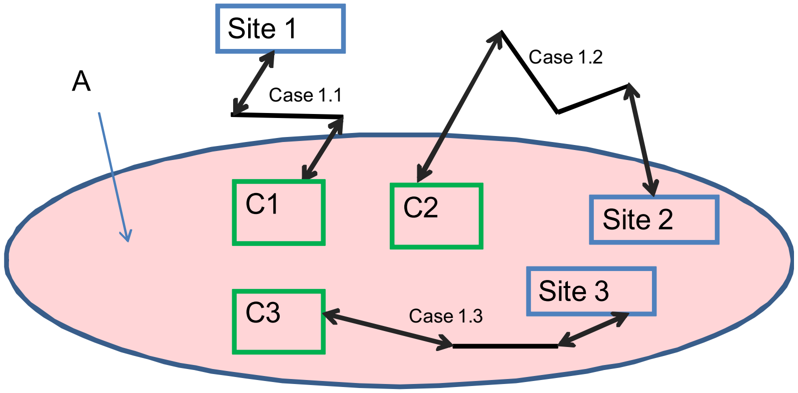
Replication

- Variety of Roles Played by ISI Researchers in Experiments
  - Quasar – the researchers developing a novel prototype
  - SAFER – the testbed and evaluation environment team
  - EdgeCT – the testbed and evaluation environment team *and* later a second transition-to-real-world testbed team
  - XD3 – team conducting evaluation (off testbed and on testbed) *after* prototyping
  - REAPER – researchers re-implementing a previously published result with a different (more advanced) artifact

Understanding that there are many different vantage points when addressing **``why are we replicating an experiment?''**

SAFER

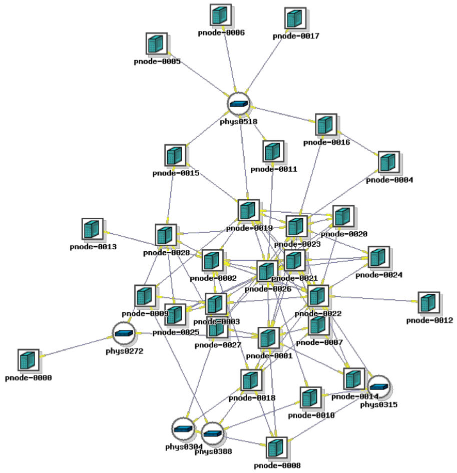
- Anonymous and Non-blockable Communication
  - Tor
  - Techniques and Extensions to Tor (SAFEST, DEFIANCE)
  - Alternatives to Tor and/or related problems of Anonymity (SONATA, DISSENT)
  - Non-blockable (CURVEBALL)



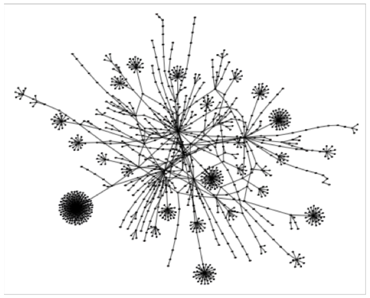
NGO Communicators (C1, C2, C3) accessing social media sites 1, 2, and 3, inside and outside RAT-controlled network region A

SAFERLab: System Concept and Architecture

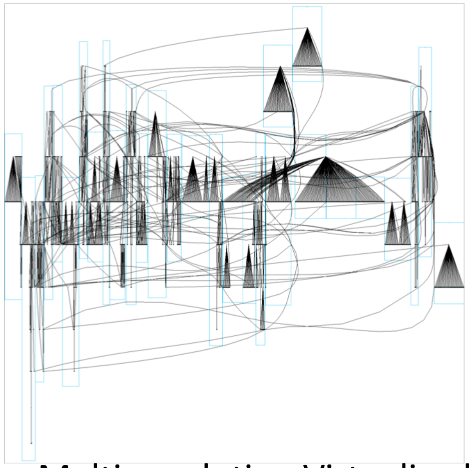
- SAFERlab supports the SAFER program by providing an experimental environment, by organizing for collaboration, and by striving to understand, capture, and model the "world" in which SAFER technologies and capabilities will play out.



Physical Node Topology



Abstract (Realistic) Topology



Multi-resolution Virtualized Topology

Demonstration of Integrated Technologies

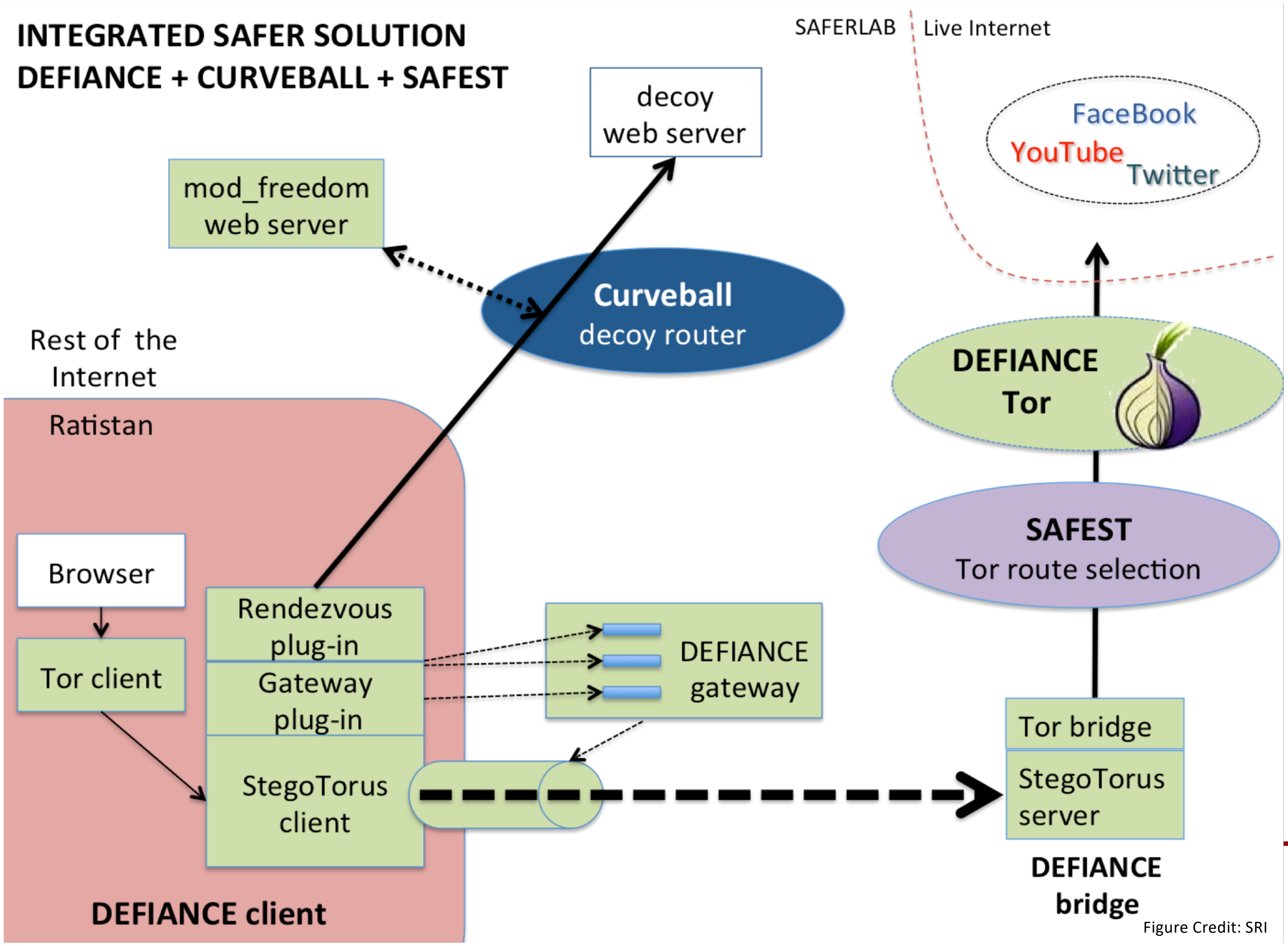


Figure Credit: SRI

Replication Challenges:

Tor model --

- SAFEST selects Tor relays based on observed inter-relay latency, BW
- Need a scaled down but property preserving Tor inside DETER
- Enables discover of the ``School of Fish Attack''

CURVEBALL –

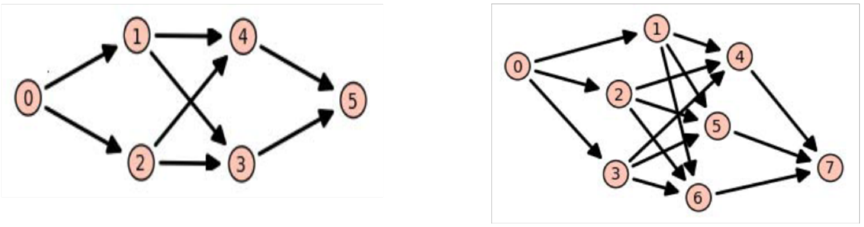
- Probability of detection
- Probability of decoy router on path?
- Requires observational study & modeling

DEFIANCE – Tor bridges

- Steganography using obsproxy
- Can't easily model
  - Ad hoc (human) distribution
  - Proof-of-life Onions
  - Steganography detection

SONATA: Anonymity & Non-blockability via Network Coding

- SONATA's overlay architecture specifies three node roles: *relays*, *exit nodes*, and *proxy nodes*.
- Relays perform network coding and forwarding services.
- Exit nodes also recover the original data stream for a particular subgraph(s) and forward into the traditional IP infrastructure.



Subgraphs used for coding deployed within the Internet

- SONATA requires a sophisticated degree of random linear coding and key management
- Significant logging and post-mortem analysis required to determine if SONATA experiment is performing:
  - (a) correctly
  - (b) optimally given network conditions
- Excessive logging to disk may perturb performance

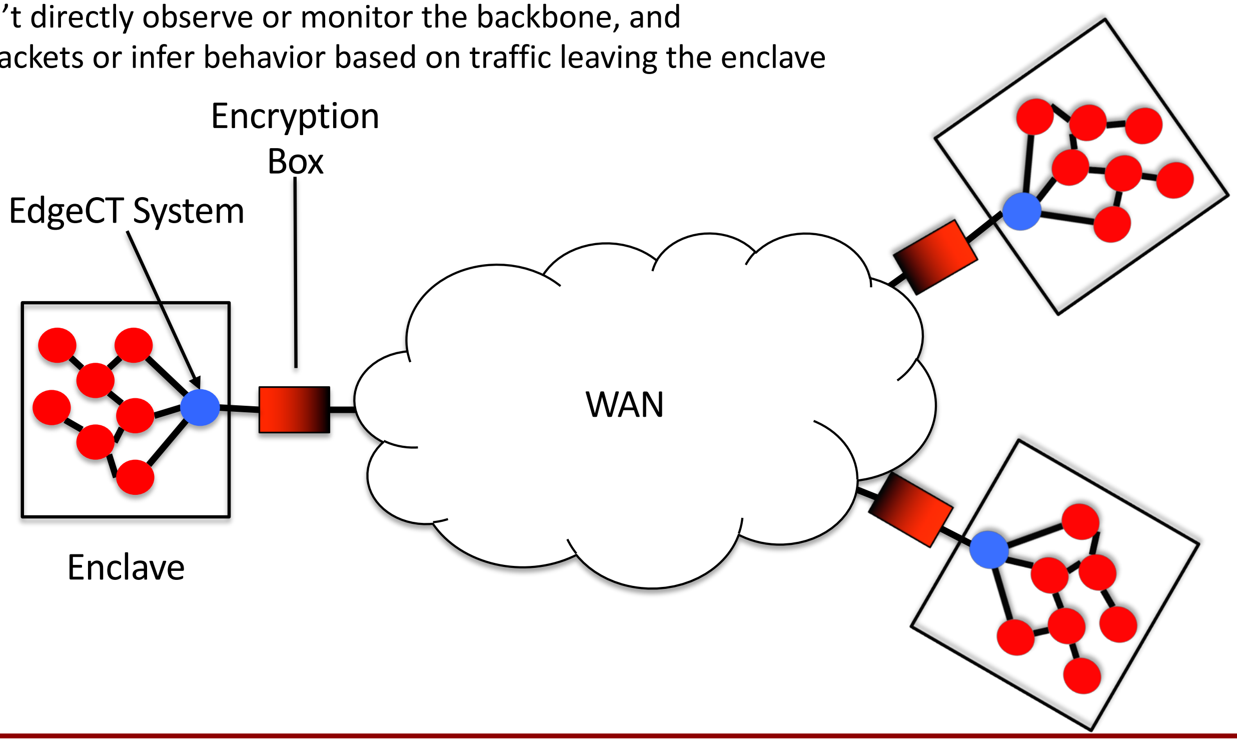
REQUIREMENT FOR REPLICATION:

- SONATA incorporates numerous levels of logging and fine-grain control, to minimize disruption while capturing data.

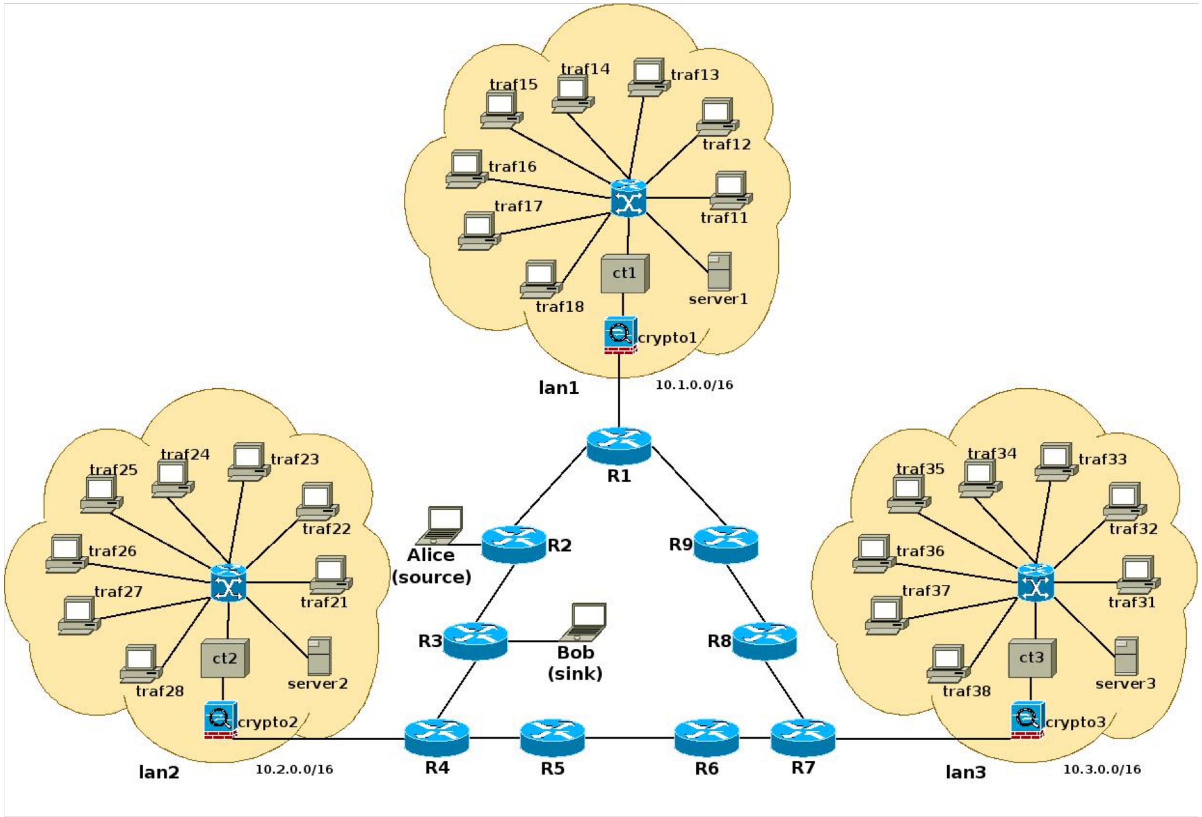
Figure Credit: BAE Systems

EdgeCT

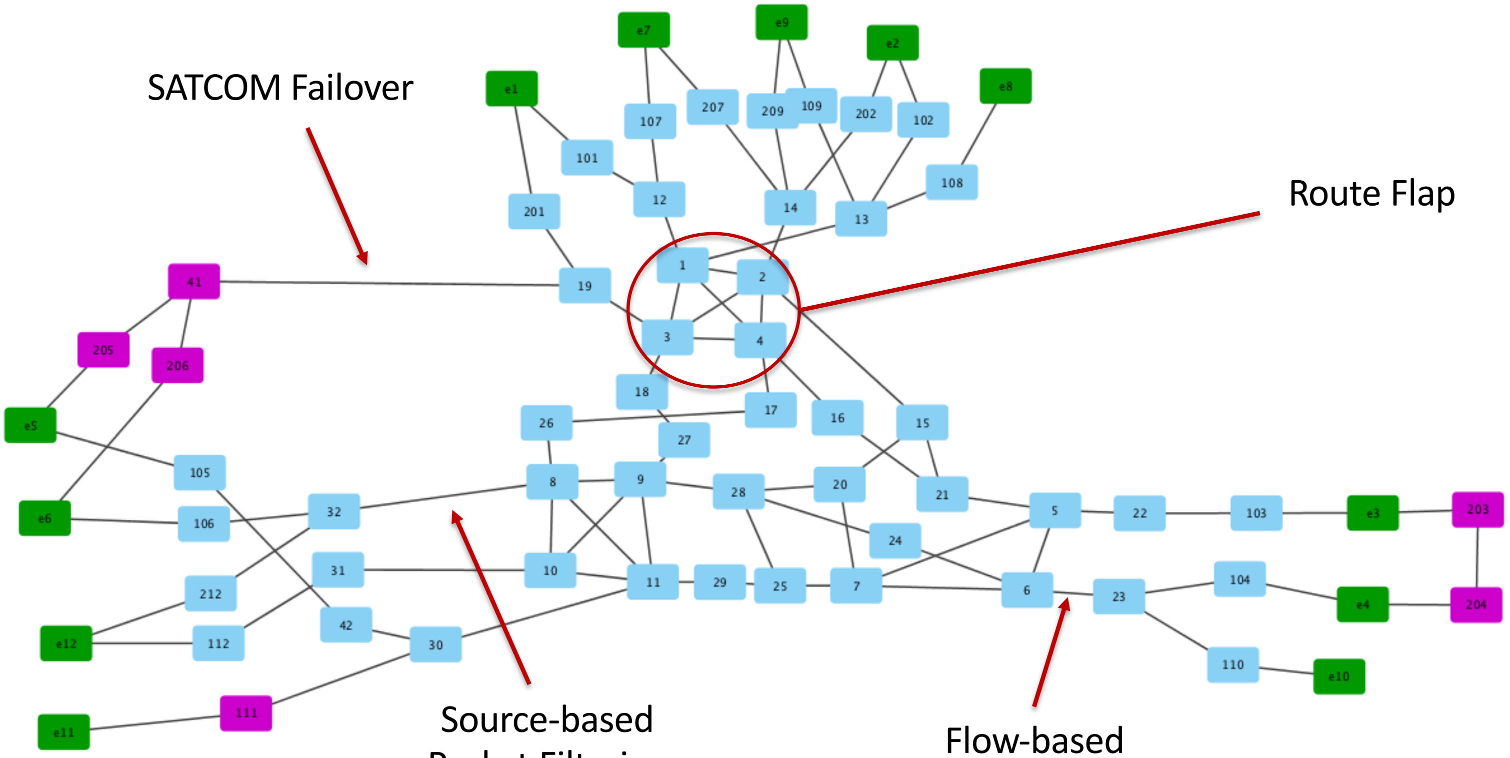
- Optimizing communications across a WAN backbone …
  - From behind a VPN encrypting gateway
  - Even when you can't directly observe or monitor the backbone, and
  - Must send probe packets or infer behavior based on traffic leaving the enclave



DETER Edgelab Topology



DETER EdgeLab Topology



SATCOM Failover

Route Flap

Source-based Packet Filtering

Flow-based Reordering

The iLab

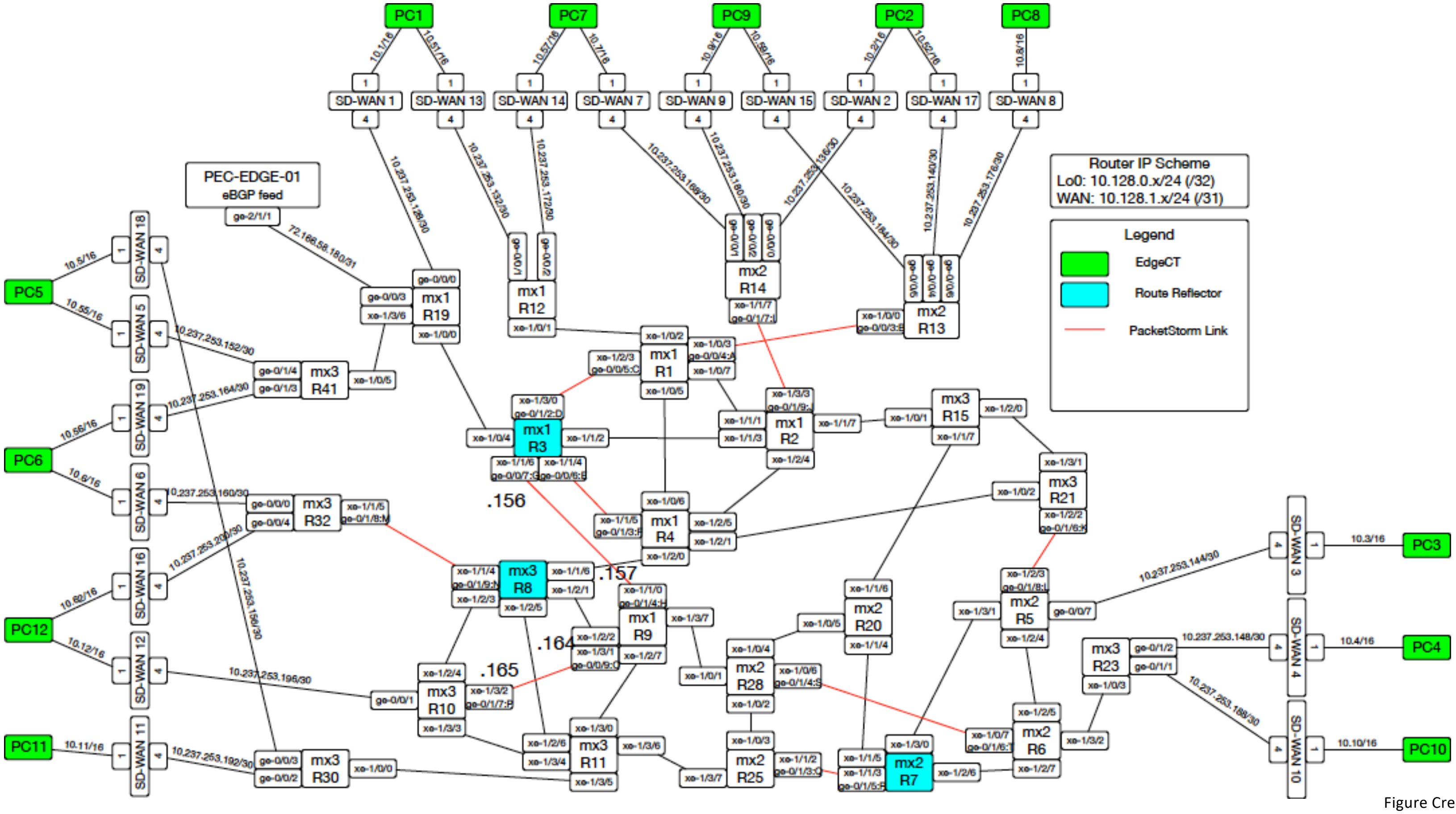


Figure Credit: CenturyLink

EdgeCT Replication Challenges

- Challenge #1: DETER topologies and experiments vs. iLab experiments
  - PCs vs. real routers?
  - PCs emulating routers vs. real routers serving as virtualized routers
  - Traffic generation from end-systems (PCs) vs. commercial traffic generator
  - …
- Challenge #2: Network Impairments
  - TCP congestion control schemes are an active area of research for EdgeCT
  - network impairment and slow link modeling vs. high-speed interfaces
    - Buffer bloat, congestion collapse
    - Model impairment drop queues carefully (DETER used CoDel)
    - Root cause due to middle-box traffic shaping vs. end-system traffic shaping

XD3

- DDoS Defenses – 7 projects focused on three distinct technical areas
- Focus today: 2 Low-volume / end-point DDoS prototype defenses
  - DeDoS - subdivide program into MSU (software units); replicate to mitigate DoS
  - MagicWand – ML based training against LV-DoS attacks; IP table rules to mitigate
- Low Volume DoS attacks
  - Regular expression explosion (ReDos) – computational time of 100ms per request
  - TLS renegotiation – 100X renegotiation to waste CPU time
  - SlowLoris -- many open TLS connections (~40,000)
  - Others: SlowRead, Sockstress, Apache-Killer, RUDeadYet?

*DeDos: Upenn; **MagicWand: Two-Six Labs

Replication Challenges for XD3

- DeDoS
  - migrated to DETER node
  - Latent bug in DeDoS code
  - Triggered more frequently (accelerated) by running on faster DETER node
  - Easy corrected by researchers in their prototype
- Scale-up of attack rate necessary
  - Configure attack to actually impact server hosted on DETER nodes
  - Required reconfiguration of DeDoS defense, e.g. more file descriptors

- Real-world requirements: separate orchestration of experiment world from configuration & orchestration of prototype technology.

Replication Challenges for XD3

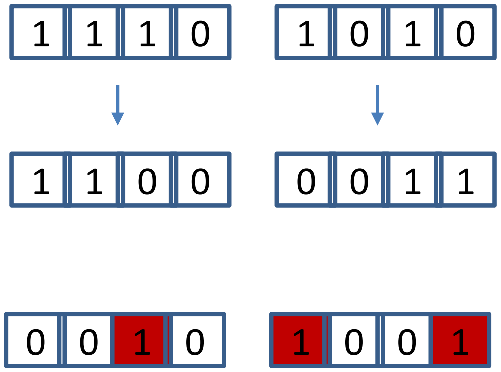
- MagicWand
  - Experiment used many containers on a beefy x86 machine
  - Replicate MagicWand's experiment on a DETER x86 node of similar capacity
  - Restructured orchestration scripts to separate prototype from test harness
  - Significantly worse performance on DETER. Different results? But why…
- Kafka implementation at fault? Red herring
  - Benchmark Comparisons – MagicWand's storage hardware was 30X faster
  - Erroneous reporting of HDD (through the kernel) using a HW RAID controller
  - SSD (fast, on Magic Wand's system) vs. HDD (slow, on DETER node)
- Real-world requirements:
  - Know hardware, instrument your system to be sure
  - Testbed health and experiment health & monitoring are required

REAPER

- Overall project: reverse engineering of FPGA bitstreams
  - Interesting techniques for manipulating obscure software/formats used in older FPGAs and toolchains
  - Cybersecurity: Encrypted Bitstreams may only be loaded into FPGAs with keys
- Differential Power Analysis
  - Side-channel technique used to leak the key
  - Work-in-progress to replicate a 2011 paper* applying this technique to Virtex-II
    - 3DES hardware encryption
  - Not *exactly* replicating because we are applying this technique to Virtex-4
    - AES hardware encryption
- Except …

*Moradi, Barenghi, Kasper & Paar,
“On the vulnerability of FPGA Bistream Encryption
Against Power Analysis Attacks’’, 2011

Differential Power Analysis



1 1 1 0 → 1 1 0 0

1 0 1 0 → 0 0 1 1

0 0 1 0   1 0 0 1

HAMMING DISTANCE = 3

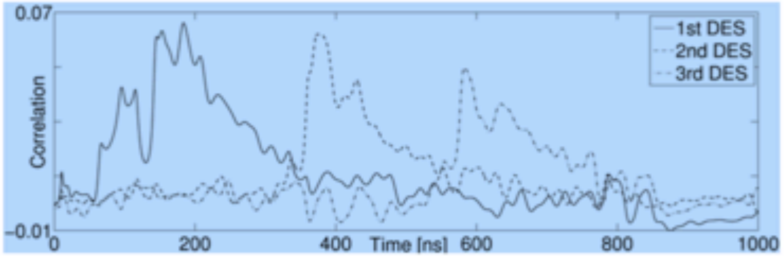


Figure 6: Correlation for hypothesizing the HD between the output of the first two rounds of each DES module

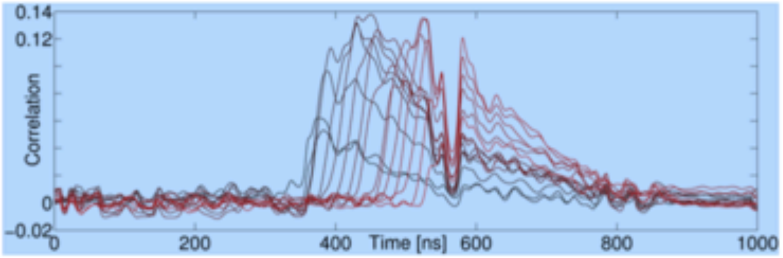
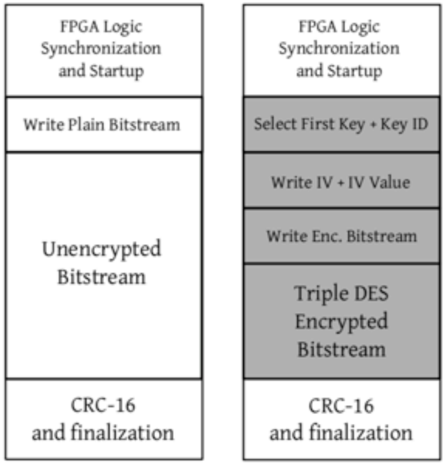
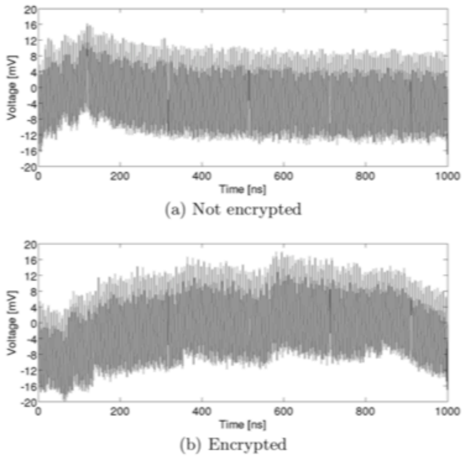


Figure 7: Correlation peaks for HD between the outputs of all the 16 rounds of the second DES

REAPER


FPGA Logic Synchronization and Startup
Write Plain Bitstream
Unencrypted Bitstream
CRC-16 and finalization
FPGA Logic Synchronization and Startup
Select First Key + Key ID
Write IV + IV Value
Write Enc. Bitstream
Triple DES Encrypted Bitstream
CRC-16 and finalization

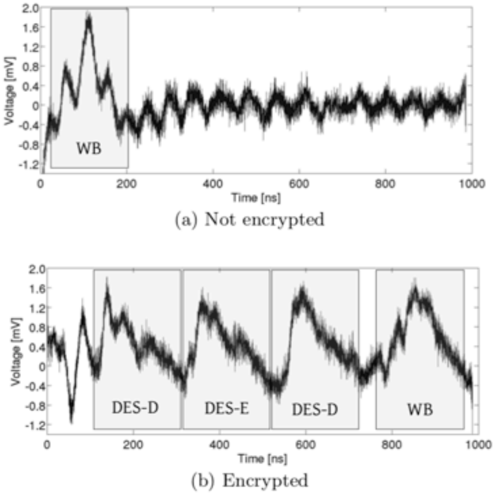
1. Comparison [Un]encrypted



(a) Not encrypted

(b) Encrypted

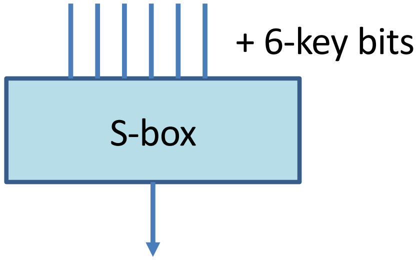
2. Raw Power Measurement



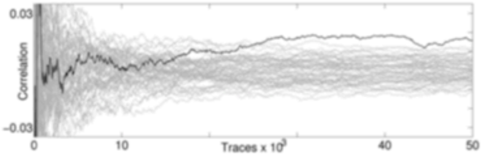
(a) Not encrypted

(b) Encrypted

3. Filtered Power Measurement



4. Recover of Keys via Analysis



5. DPA Attack Result

REAPER

- DPA Key Extraction – Step-by-step Replication of Experiment
- FPGA device: identify bitstream format and Vcc pin
- Hypothesis/Method: Hamming Distance of Register (old, new) ∝ power
  - Analysis of DES S-boxes and cipher structure
  - Identify 6-bits of key that influence output changes of register in each round
- Signal Processing:
  - Alignment (Cross-correlation), Digital Filtering (High and low bandpass)
- ``For verifying that our setup… Experimenting with various self-generated bitstreams, e.g. **by means of arbitrary IVs**, … and successfully verified the functionality of our setup.'' (Sec 3.2)

Summary of Real-world Replication Requirements

- Quasar –
  - Statistics – long-tails & many trials
  - Physical device behavior
  - Complex procedures
- SAFER –
  - Real-world routing
  - Combining testbed experiments & analysis w.r.t. real-world
  - Can be impossible to model “security analyst’ insights w.r.t. fragile techniques
- EdgeCT –
  - DETER Emulation vs. iLab dedicated HW/router/impairment boxes
  - Protocol interactions with network impairment modeling
- XD3 –
  - Not DETER (AWS / Cloud) vs. DETER (or other controlled testbeds)
  - Understanding physical node (PC) features and impact
- REAPER –
  - Underspecified steps in experimental procedures

Questions!