Increasing Cyber-Resilience of Large-Scale and Long-Lived Energy Delivery Infrastructure (EDI)

Prashant Anantharaman, J. Pete Brady, Sergey Bratus, I. Ray Jenkins, Michael Millian, David Nicol (Illinois), Kartik Palani (Illinois), Elizabeth Reed (Illinois), Sean Smith

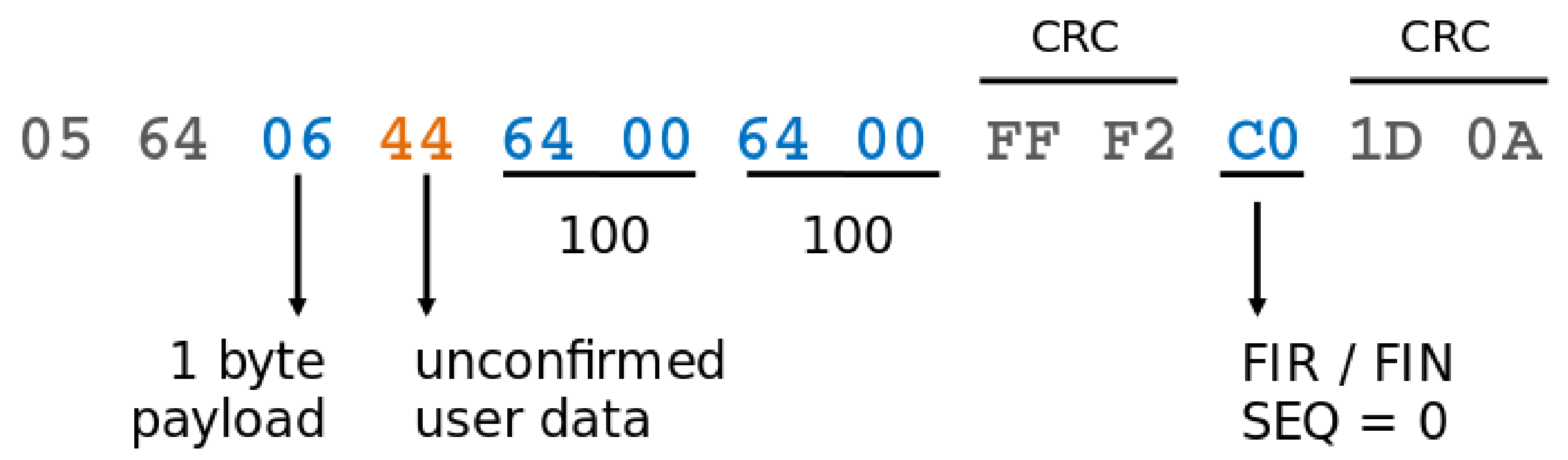
GOALS

Embedded systems are rife with security holes, with 0-days and forever-days. At larger scales in space, pushing patches to these boxes will be complicated. To address these problems we are working on the following.

- *Prevention:* Building tools to help prevent 0-days and forever-days in the first place (e.g., hardened parsers for DNP3, Modbus and SSP21.)
- *Mitigation:* Building tools to help mitigate 0-days and forever-days discovered later (e.g., verifiable protocol filters and interface snap-ins)
- *Evaluation:* Building simulation tools to evaluate how effective such tools will be when scaled up to long-lived EDI. (E.g., what approach makes the biggest improvement? For security, can N firewalls do almost as well as 100N verifiable devices?)

FUNDAMENTAL QUESTIONS/CHALLENGES

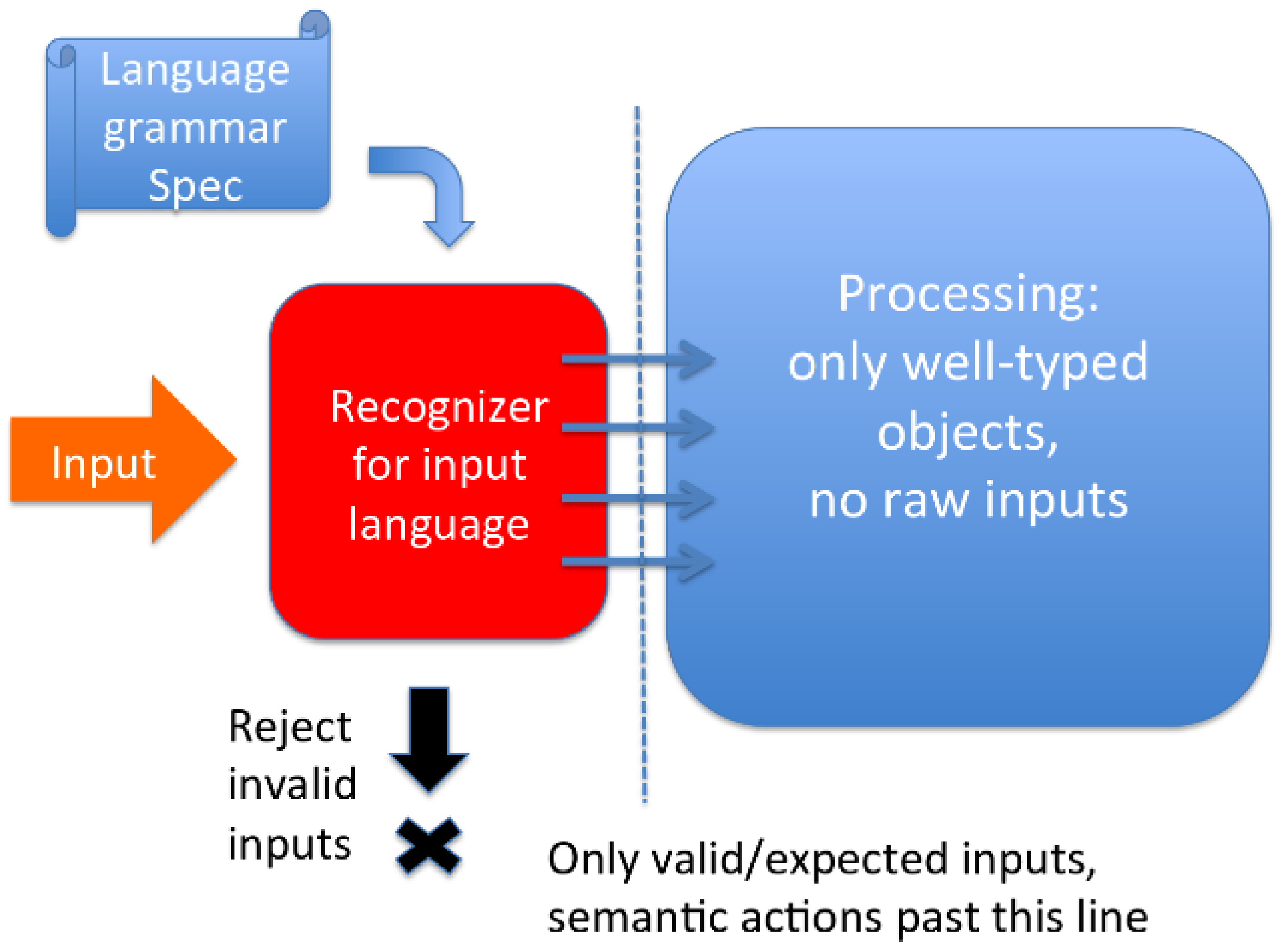
- What are the implications of vulnerability discoveries and mitigation strategies in this emerging infrastructure?
- Implement a proof-of-concept of *interface snap-ins for protocol conduits*, replacing a vulnerable feature of OpenSSL with a mitigating snap-in conduit.
- From 2013 to 2014 – Over 30 CVEs related to input validation with DNP3 implementations. (“Robus Master Serial Killer”, Sistrunk & Crain, 2014). The image below demonstrates a crafted zero length packet that could lead to an exploit. One of our goals is to demonstrate the efficacy of using LangSec to address vulnerabilities like these.
- Exploring the use of kernel hardening methods like Grsecurity/PaX to improve the security of implementations of ICS protocols.



RESEARCH PLAN

Prevention

- Extend our work by constructing hardened implementations of more ICS protocols by extending the work done already done for the DNP3 protocol.
- Using Grsecurity/PaX, which is a state-of-the-art Linux hardening technology.


Language grammar Spec
Input
Recognizer for input language
Processing: only well-typed objects, no raw inputs
Reject invalid inputs
Only valid/expected inputs, semantic actions past this line

Mitigation

- Constructing dependency trees of running applications, along with the software modules and versions, and manually evaluate these with respect to the known vulnerabilities (such as CVEs).

Evaluation

- Use simulation techniques to evaluate aggregate effectiveness of at least two proposed hardening solutions against projected models of zero-day blooms.

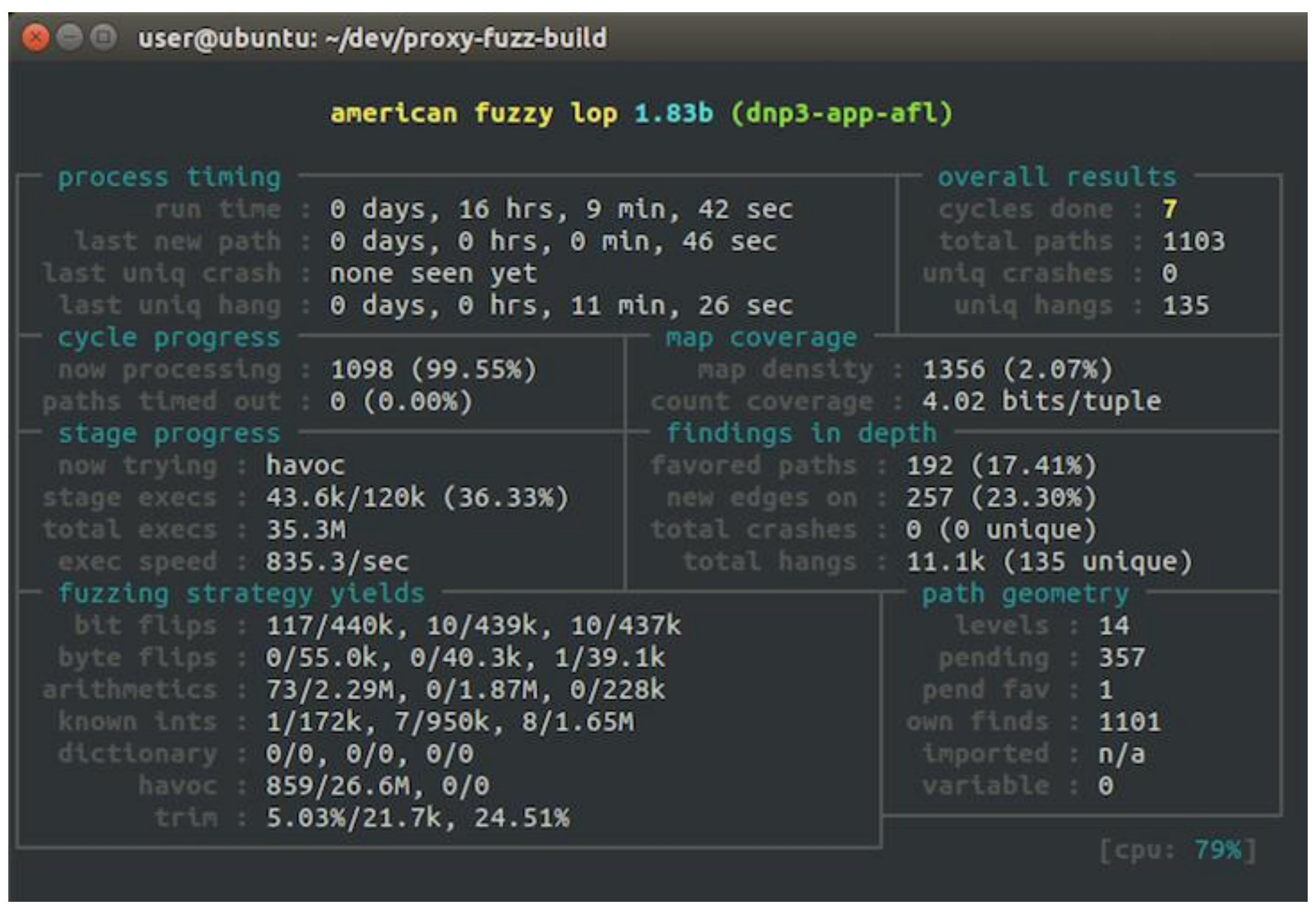
INITIAL REPORTS

- K. Palani, E. Holt, and S. Smith. “Invisible and Forgotten: Zero-Day Blooms in the IoT.” The 1st IEEE PerCom Workshop on Security, Privacy, and Trust in the IoT. March 2016. http://www.cs.dartmouth.edu/~sws/pubs/phs16.pdf
- S. Bratus, M. Patterson, A. Crain, S. Hallberg, S.W. Smith. “Implementing a Vertically Hardened DNP3 Control Stack for Power Applications." ACSAC Industrial Control System Security Workshop. December 2016. http://www.cs.dartmouth.edu/%7Esws/pubs/dnp3-icss2016.pdf

RESEARCH RESULTS

Prevention

- Our DNP3 parser survived the American Fuzzy Lop - the generic coverage-guided fuzzer.
- Our parser also survived the commercially available Model-based fuzzer, Aegis.



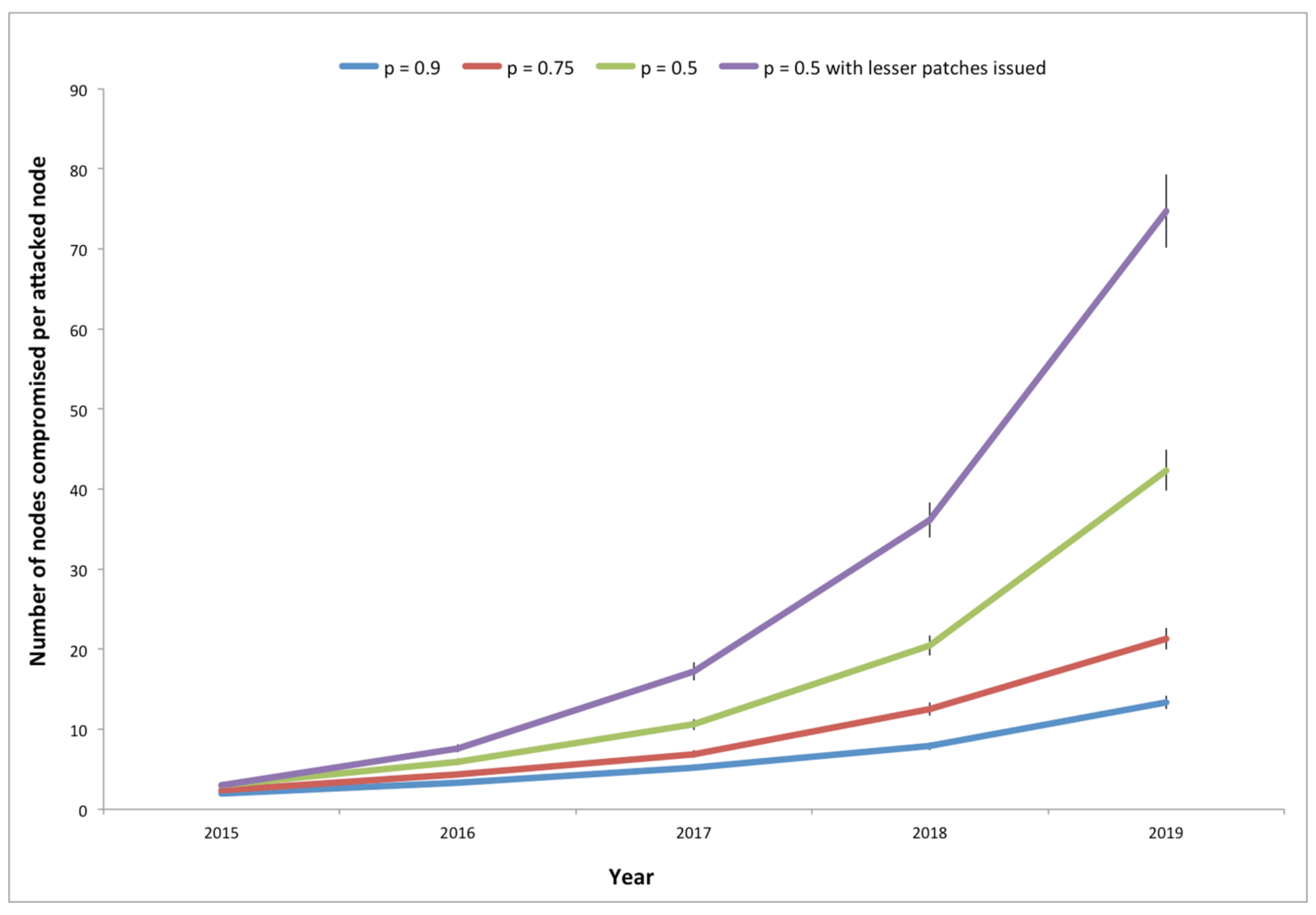
```
user@ubuntu: ~/dev/proxy-fuzz-build

                american fuzzy lop 1.83b (dnp3-app-afl)

┌─ process timing ─────────────────────────────────────┬─ overall results ─────┐
│        run time : 0 days, 16 hrs, 9 min, 42 sec      │  cycles done : 7      │
│   last new path : 0 days, 0 hrs, 0 min, 46 sec       │  total paths : 1103   │
│ last uniq crash : none seen yet                      │ uniq crashes : 0      │
│  last uniq hang : 0 days, 0 hrs, 11 min, 26 sec      │   uniq hangs : 135    │
├─ cycle progress ────────────────────┬─ map coverage ─┴───────────────────────┤
│  now processing : 1098 (99.55%)     │    map density : 1356 (2.07%)          │
│ paths timed out : 0 (0.00%)         │ count coverage : 4.02 bits/tuple       │
├─ stage progress ────────────────────┼─ findings in depth ────────────────────┤
│  now trying : havoc                 │ favored paths : 192 (17.41%)           │
│ stage execs : 43.6k/120k (36.33%)   │  new edges on : 257 (23.30%)           │
│ total execs : 35.3M                 │ total crashes : 0 (0 unique)           │
│  exec speed : 835.3/sec             │   total hangs : 11.1k (135 unique)     │
├─ fuzzing strategy yields ───────────┴───────────────┬─ path geometry ────────┤
│   bit flips : 117/440k, 10/439k, 10/437k            │    levels : 14         │
│  byte flips : 0/55.0k, 0/40.3k, 1/39.1k             │   pending : 357        │
│ arithmetics : 73/2.29M, 0/1.87M, 0/228k             │  pend fav : 1          │
│  known ints : 1/172k, 7/950k, 8/1.65M               │ own finds : 1101       │
│  dictionary : 0/0, 0/0, 0/0                         │  imported : n/a        │
│       havoc : 859/26.6M, 0/0                        │  variable : 0          │
│        trim : 5.03%/21.7k, 24.51%                   ├────────────────────────┘
└─────────────────────────────────────────────────────┘             [cpu: 79%]
```

Evaluation

- We computed the number of nodes that get compromised per attacked node over time for different patchability constants (p), which is the probability that a patch successfully reaches a node.



BROADER IMPACT

- The long term goal of our work is to address avoiding zero-day blooms as much as possible. Our work related to evaluation of zero-day blooms could also be extended to evaluate the performance of mitigation and prevention schemes.

INTERACTION WITH OTHER PROJECTS

- This work motivates the urgent need for the scalable and fast authentication techniques being proposed in the Namespace and Cryptographic Complexity work by Palani (Illinois), Nicol (Illinois), Anantharaman (Dartmouth) and Smith (Dartmouth).
- The work also draws parallels from previous work by Nicol (Illinois) studying worm infestations in the large-scale Internet.

FUTURE EFFORTS

- Demonstrate efficacy of parsers on modbus and SSP21.
- Take into consideration heterogeneous vulnerability/patching patterns and lags, topological implications on patching and attacks, emergence of thingbots, and net physical impact of attacks and thingbots.
- Building generic fuzzers from parsers in order to fuzz test the parser logic.
- Produce a prototype of a snap-in replacement tool for practical scenarios.

CYBER RESILIENT ENERGY DELIVERY CONSORTIUM | CRED-C.ORG

FUNDING SUPPORT PROVIDED BY THE U.S. DEPARTMENT OF ENERGY AND THE U.S. DEPARTMENT OF HOMELAND SECURITY