NFDFuzz: A Stateful Structure-Aware Fuzzer for Named Data Networking

George Torres, Davide Pesavento, Junxiao Shi, Lotfi Benmohamed

National Institute of Standards and Technology

7th ACM Conference on Information-Centric Networking (ICN 2020)

National Institute of Standards and Technology
U.S. Department of Commerce

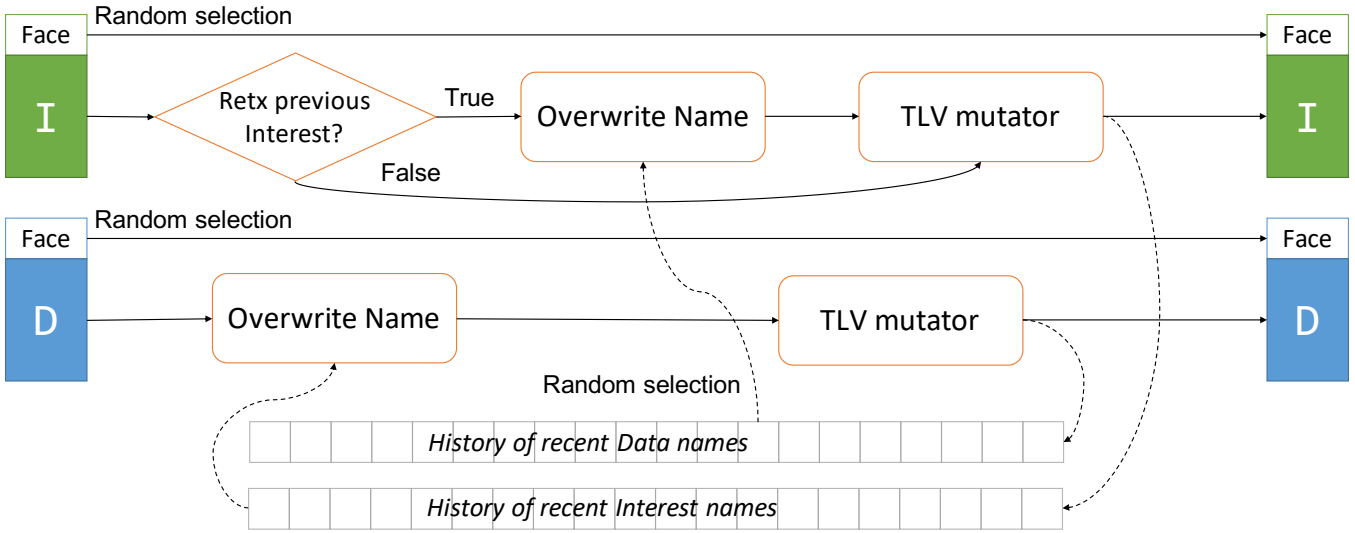
INFORMATION TECHNOLOGY LABORATORY

Introduction and Motivation

- **Fuzzing** is an automated software testing technique for finding programming errors
  - The fuzzer continuously generates invalid/unexpected inputs and feeds them to the program under test while monitoring it for crashes and other erroneous behavior
  - Particularly useful for network services, where the inputs (packets) are often untrusted
  - Very successful: as of June 2020, OSS-Fuzz found 20000 bugs in 300 open-source projects (https://google.github.io/oss-fuzz/)
- Not applied to NDN yet
- Enter **NFDFuzz**, the first fuzzer for NFD (NDN Forwarding Daemon)

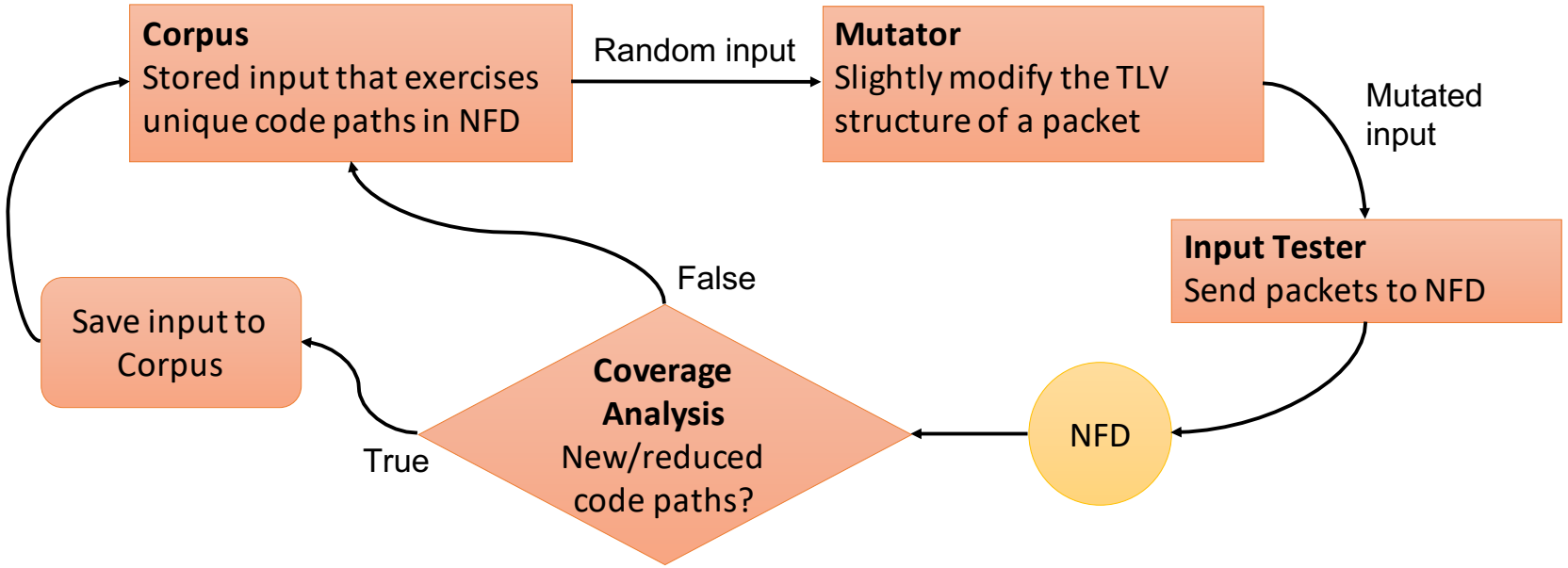
Challenges

- NDN packets are highly structured => fuzzer must be **structure-aware**
- NDN has a stateful data plane => fuzzer must be **stateful**
- A single packet is rarely sufficient to trigger a bug => fuzzer must be able to handle **sequences of packets**
- Some NDN features require more than one input/output face => fuzzer must create and maintain **multiple faces**



NFDFuzz Design

- LibFuzzer fuzzing engine + two-level custom mutator:
  - Packet-level mutator, aware of NDN Interest-Data matching semantics
  - TLV-level mutator, operating on individual TLV elements
- AddressSanitizer to detect memory errors at runtime
- FlatBuffers to serialize the inputs (packet traces) into persistent storage



Preliminary Results and Future Work

- Beta quality code available at https://github.com/gtorresz/nfdfuzzer
- 4 bugs found in just a few hours: one in PIT, one in Data decoding, two in forwarding strategies

Future work

- Expand coverage: NFD management, NDNLP, Nack, …
- Try other fuzzing engines: AFL++, Honggfuzz, …
- Investigate hybrid approach for packet generation/mutation

National Institute of Standards and Technology
U.S. Department of Commerce

INFORMATION TECHNOLOGY LABORATORY

THANK YOU