Yandex Cloud

Virtio Device Fuzzing

Dima Stepanov
Software Engineer, Yandex.Cloud Hypervisor Team

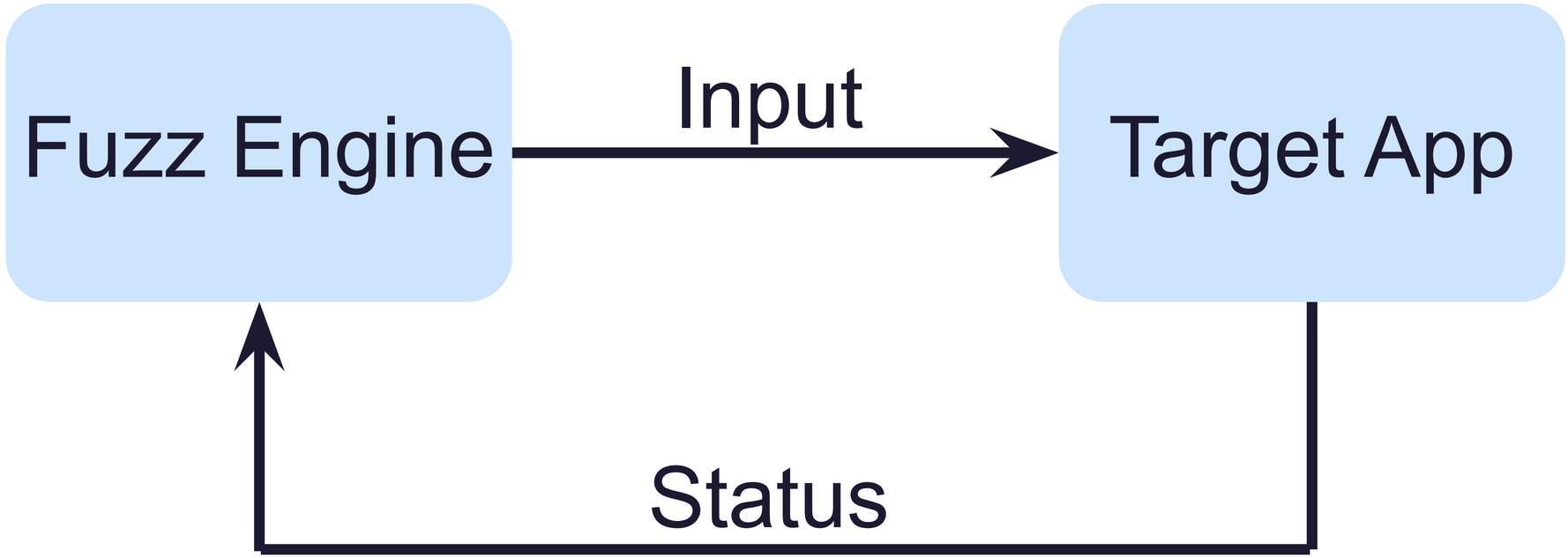
Table Of Contents

01

What is Fuzzing?

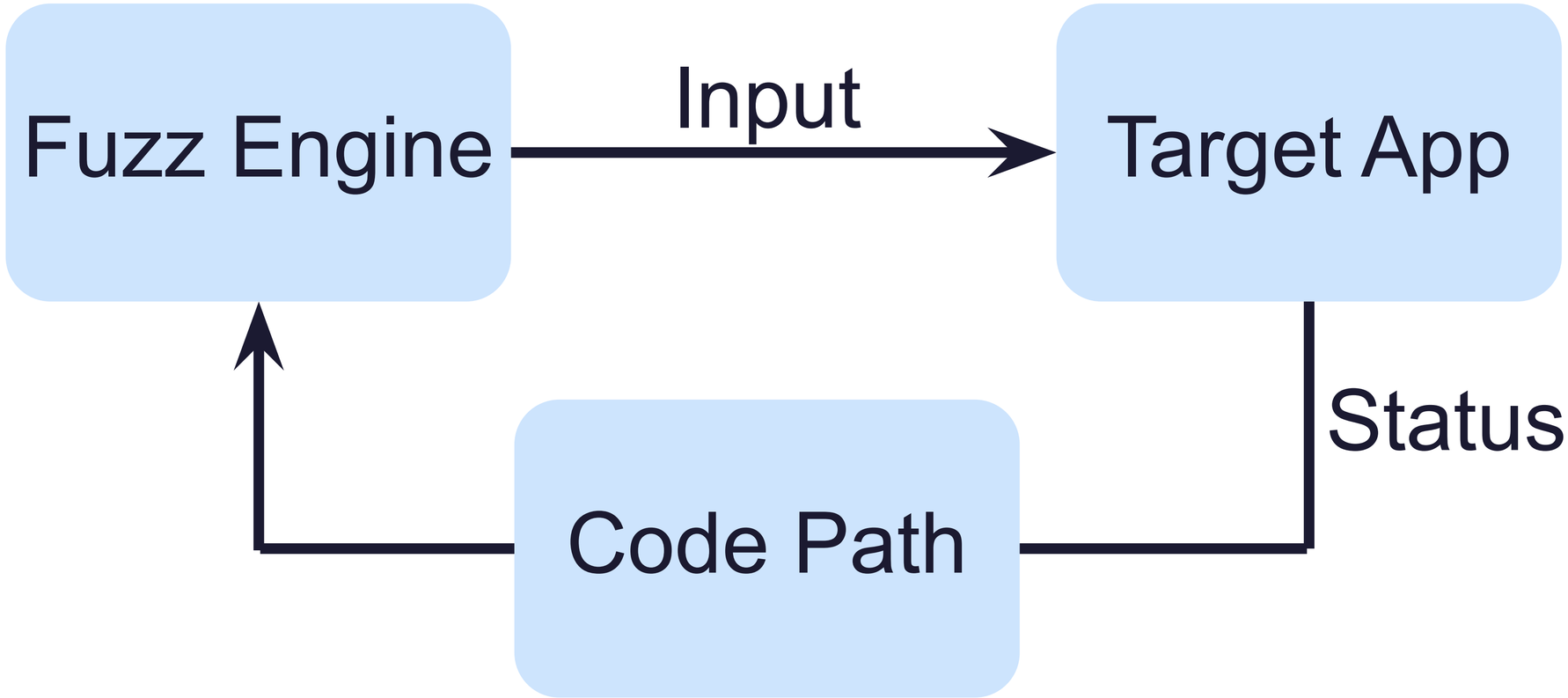
Fuzzing Technique: Part 1

black box, fuzz, test, crash

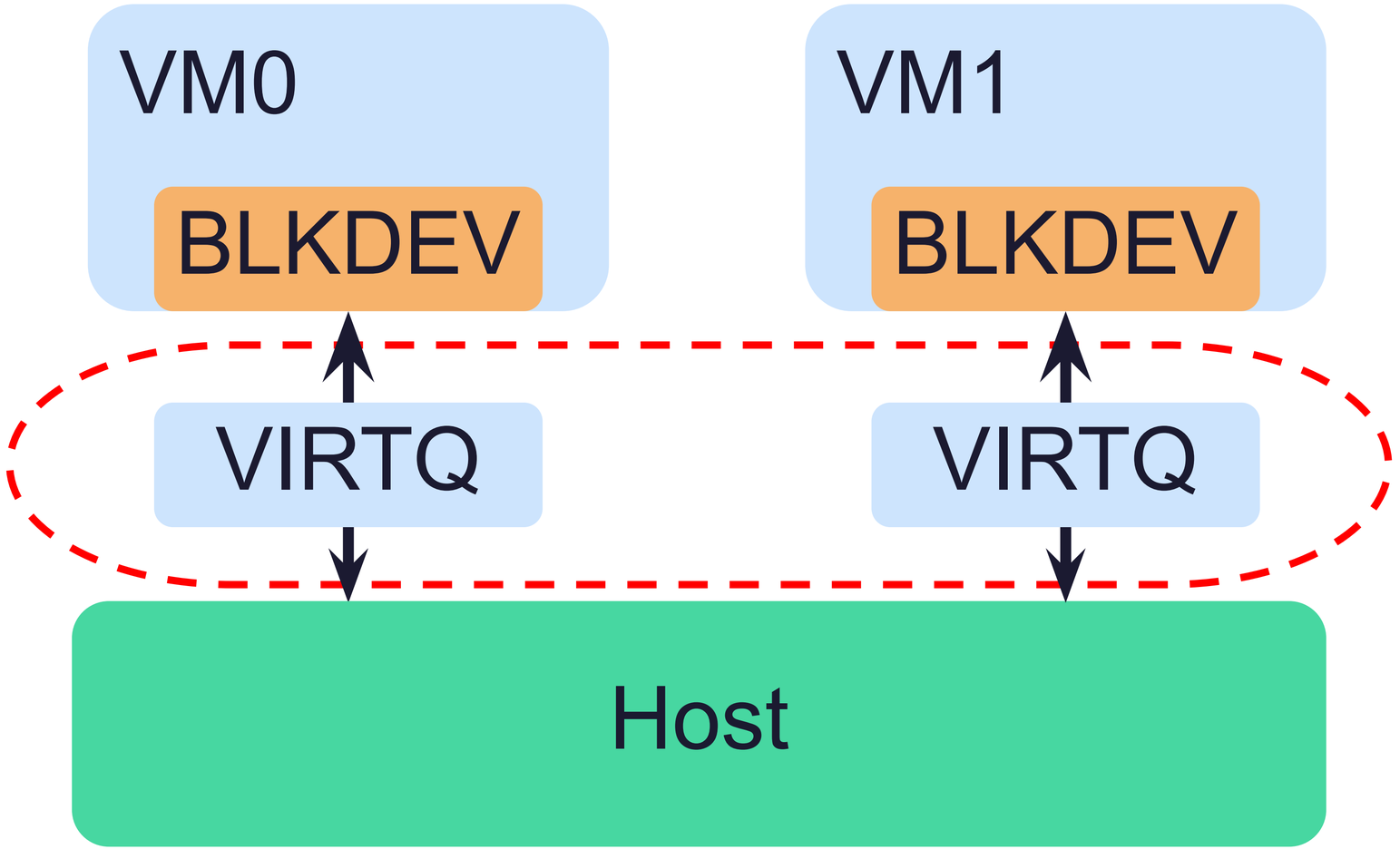


Fuzzing Technique: Part 2

grey box, coverage guided



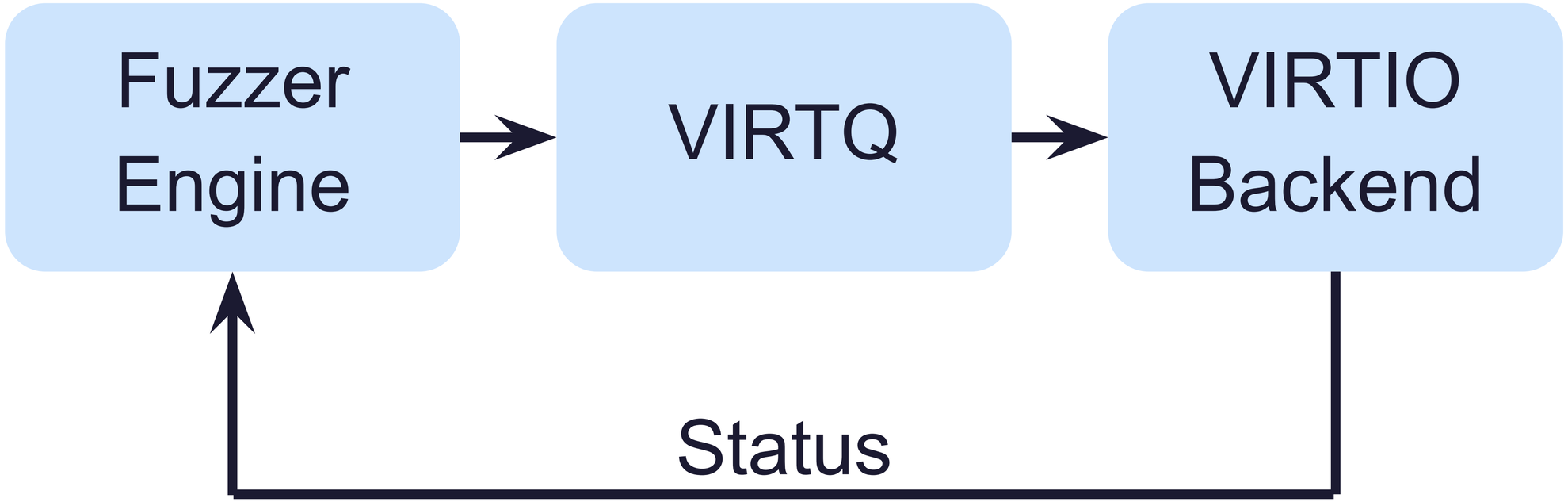
Attack Surface



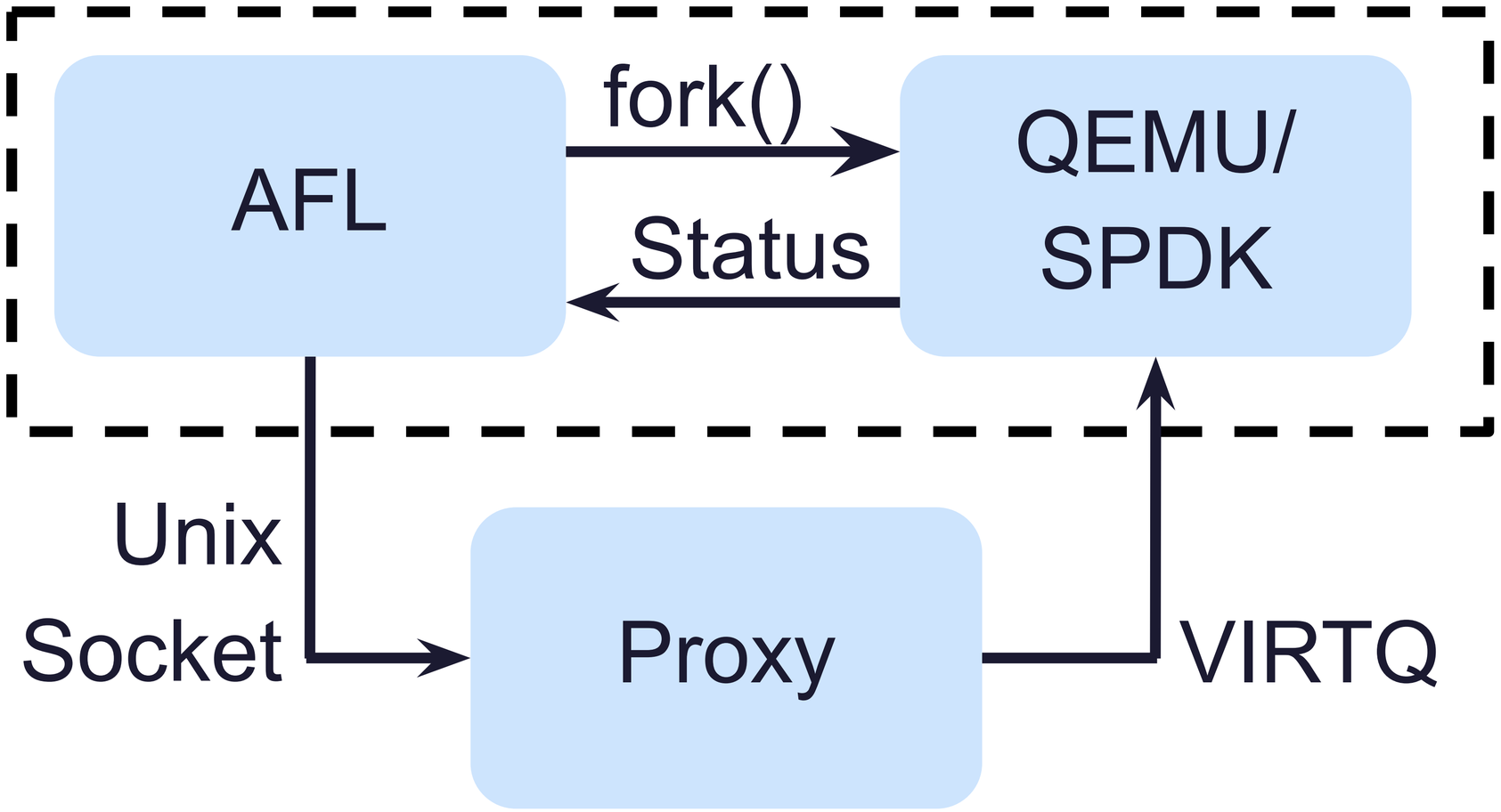
02

How We Added Fuzzing

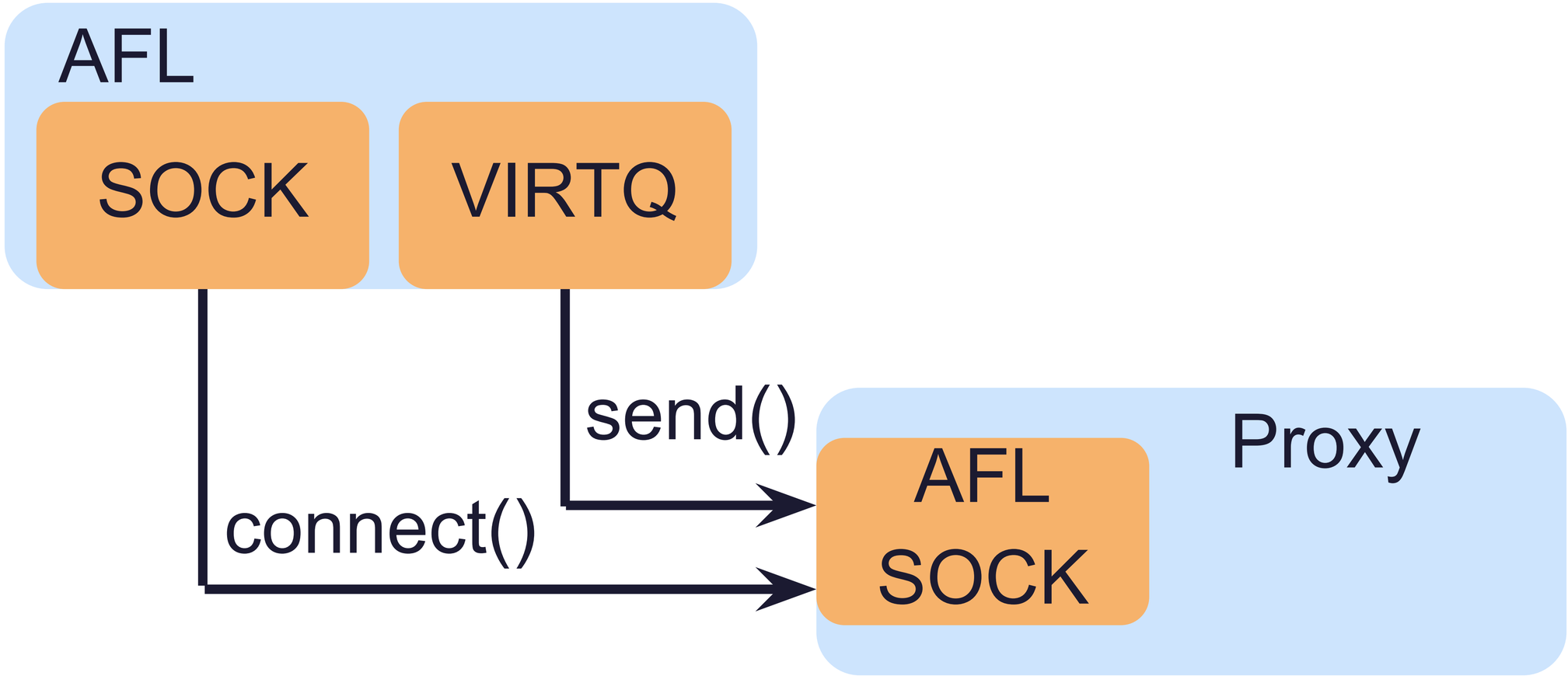
Fuzzing Scheme



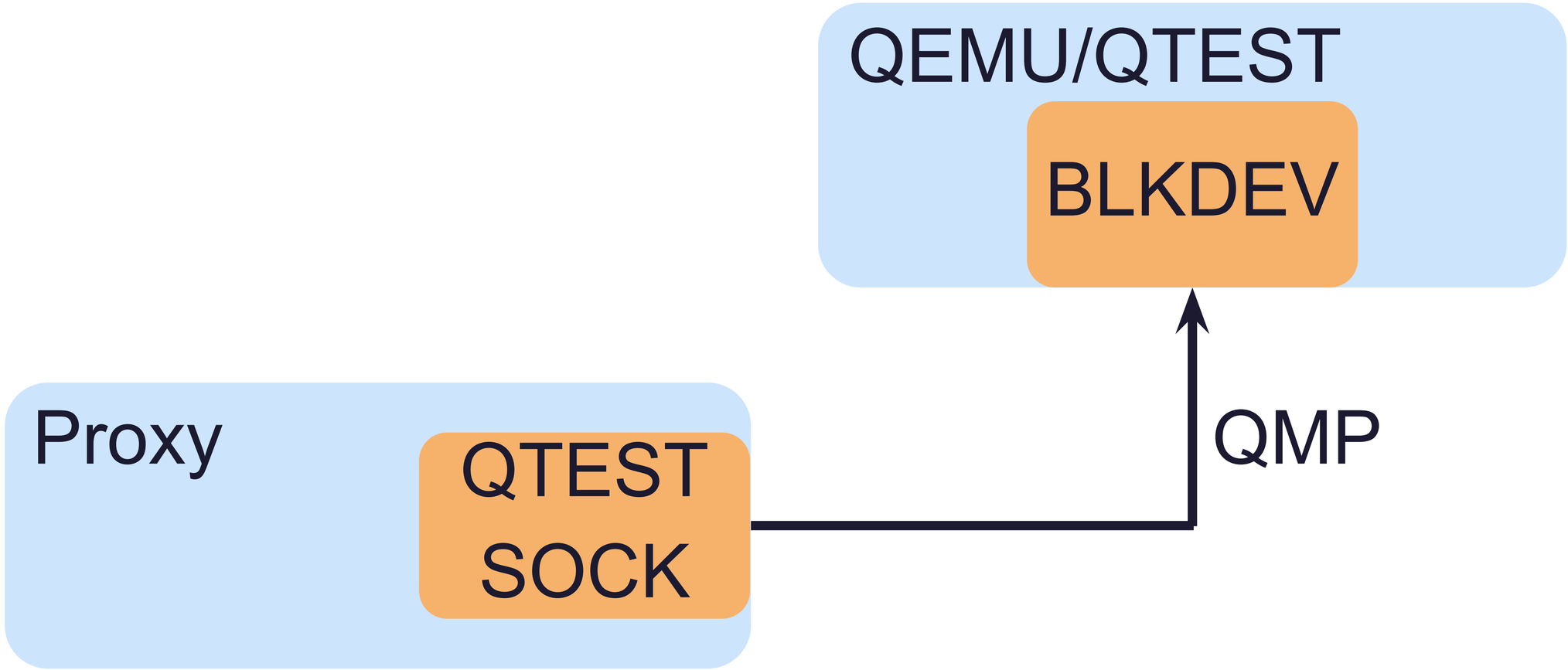
Real Fuzzing



AFL-Proxy Communication



Proxy-QEMU Communication



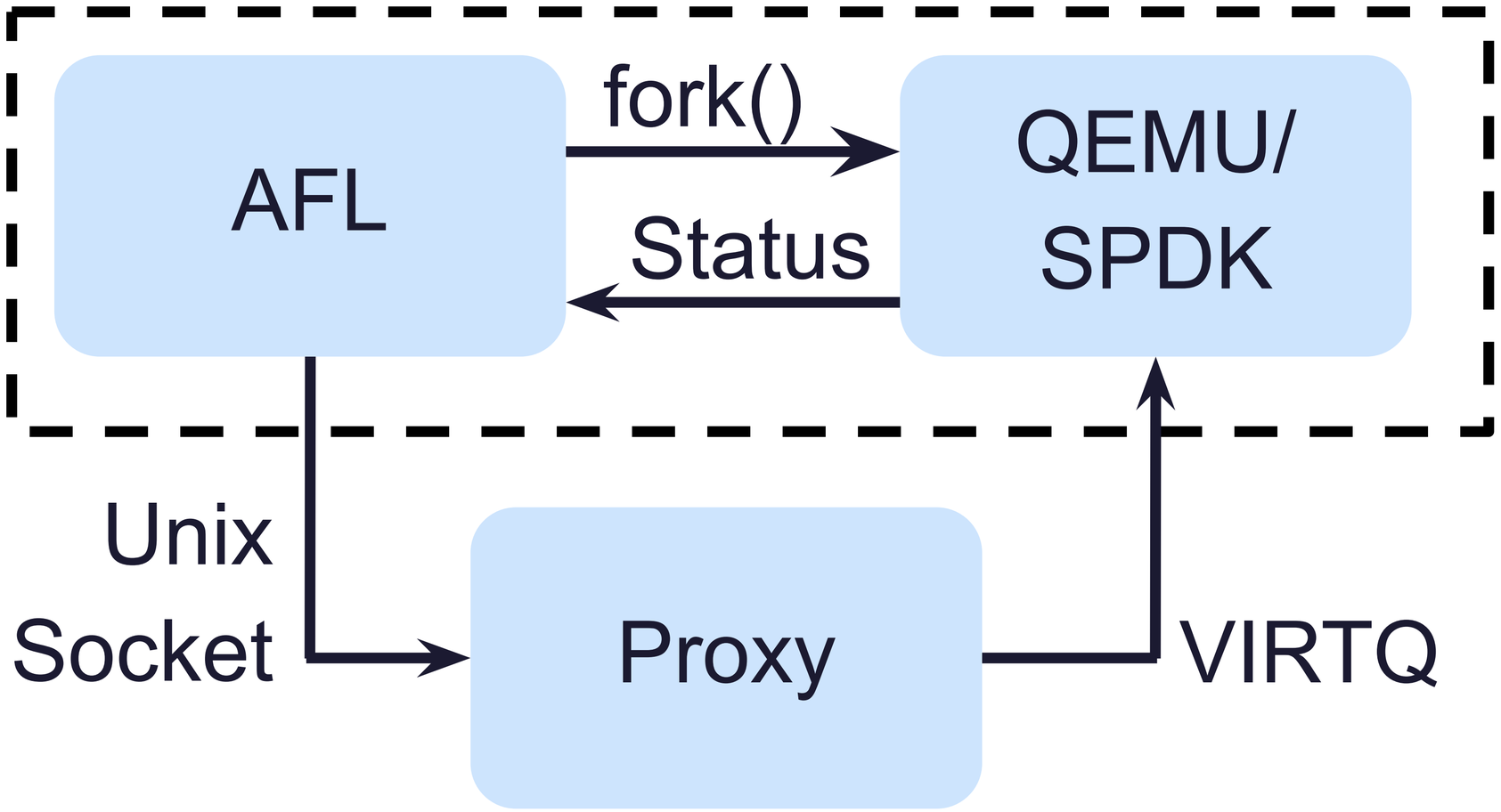
QEMU Status

code instrumentation

QEMU Status

persistent mode

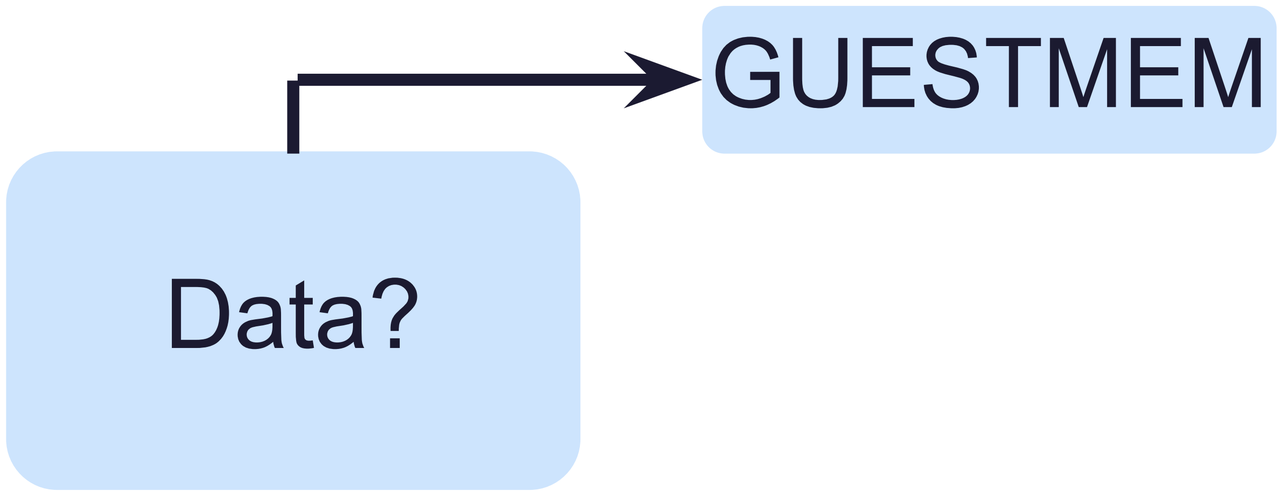
Real Fuzzing



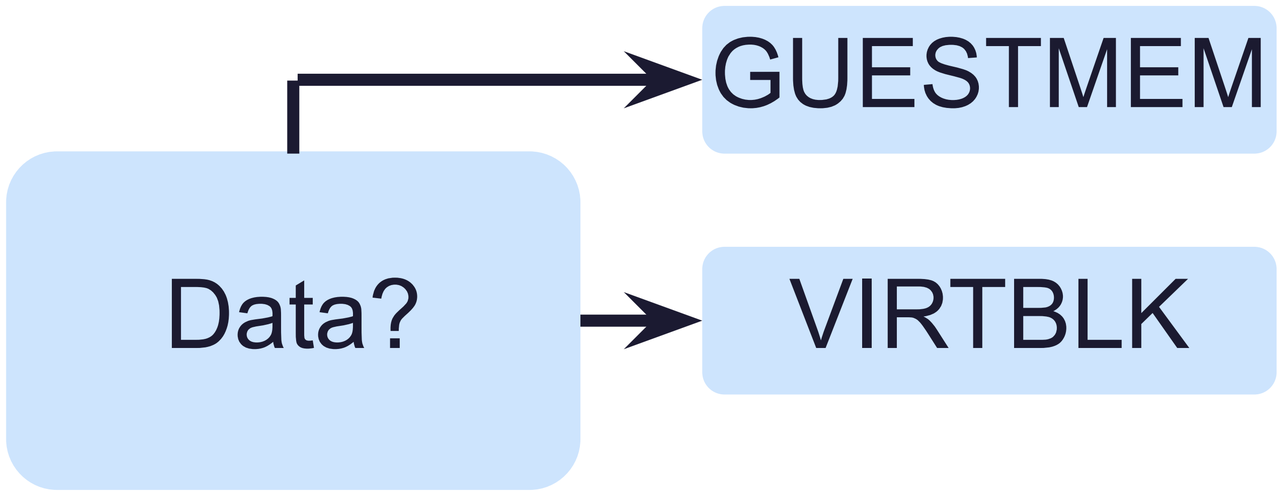
Input

Data?

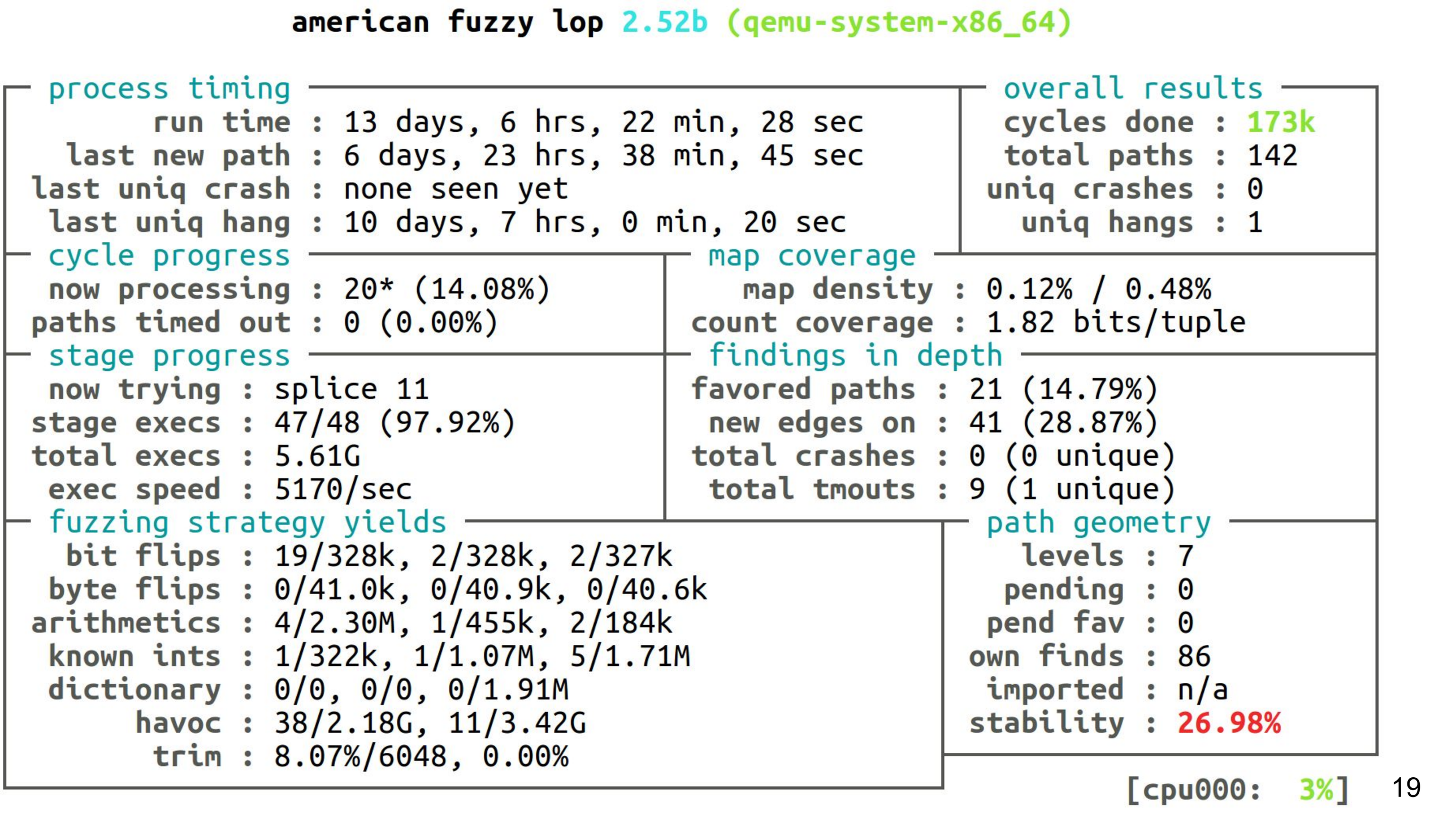
Input



Input



Input


american fuzzy lop 2.52b (qemu-system-x86_64)
process timing
run time : 13 days, 6 hrs, 22 min, 28 sec
last new path : 6 days, 23 hrs, 38 min, 45 sec
last uniq crash : none seen yet
last uniq hang : 10 days, 7 hrs, 0 min, 20 sec
overall results
cycles done : 173k
total paths : 142
uniq crashes : 0
uniq hangs : 1
cycle progress
now processing : 20* (14.08%)
paths timed out : 0 (0.00%)
map coverage
map density : 0.12% / 0.48%
count coverage : 1.82 bits/tuple
stage progress
now trying : splice 11
stage execs : 47/48 (97.92%)
total execs : 5.61G
exec speed : 5170/sec
findings in depth
favored paths : 21 (14.79%)
new edges on : 41 (28.87%)
total crashes : 0 (0 unique)
total tmouts : 9 (1 unique)
fuzzing strategy yields
bit flips : 19/328k, 2/328k, 2/327k
byte flips : 0/41.0k, 0/40.9k, 0/40.6k
arithmetics : 4/2.30M, 1/455k, 2/184k
known ints : 1/322k, 1/1.07M, 5/1.71M
dictionary : 0/0, 0/0, 0/1.91M
havoc : 38/2.18G, 11/3.42G
trim : 8.07%/6048, 0.00%
path geometry
levels : 7
pending : 0
pend fav : 0
own finds : 86
imported : n/a
stability : 26.98%
[cpu000: 3%]

03

Issues Found

VIRTIO assert (QEMU)

assert()

```
struct virtq_desc {
        le64 addr;        /* Address (guest-physical). */
        le32 len;         /* Length. */
        le16 flags;       /* The flags as indicated above. */
        le16 next;        /* Next field if flags & NEXT */
};

$ hexdump -C ./id0001.dump
00000000  40 e0 fd 3f 00 00 00 00  00 00 00 00 06 00 04 00  |..|
00000010  00 00 00 00 00 00 00 00                           |..|
```

SIGSEGV (SPDK)

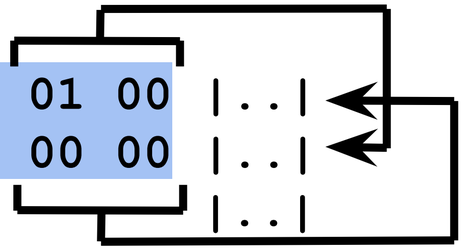
memory access violation

```
struct virtq_desc {
        le64 addr;        /* Address (guest-physical). */
        le32 len;         /* Length. */
        le16 flags;       /* The flags as indicated above. */
        le16 next;        /* Next field if flags & NEXT */
};

# hexdump -C sigsegv.dump
00000000  30 00 00 00 00 00 00 00  10 00 00 00 01 00 01 00 |..|
00000010  40 00 00 00 00 00 00 00  00 02 10 00 03 00 02 00 |..|
00000020  40 02 00 00 00 00 00 00  01 00 00 00 02 00 03 00 |..|
00000030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 |..|
```

CVE-2019-9547 (SPDK)

endless loop, DoS

```
struct virtq_desc {
        le64 addr;        /* Address (guest-physical). */
        le32 len;         /* Length. */
        le16 flags;       /* The flags as indicated above. */
        le16 next;        /* Next field if flags & NEXT */
};

# hexdump -C hang01.dump
00000000  b0 00 00 00 00 dd ff 80  00 00 00 00 01 00 01 00  |..|
00000010  40 14 00 00 f3 00 00 00  00 00 00 00 05 00 00 00  |..|
00000020  00 00 00 00 00 00 00 00                           |..|
```



04

Future Improvements

Improvement Plans

1. Fuzz virtio-net, virtio-sock, virtio-*

Improvement Plans

1. Fuzz virtio-net, virtio-sock, virtio-*
2. Fuzz PCI

Improvement Plans

1. Fuzz virtio-net, virtio-sock, virtio-*
2. Fuzz PCI
3. Enhance start corpus

Improvement Plans

1. Fuzz virtio-net, virtio-sock, virtio-*
2. Fuzz PCI
3. Enhance start corpus
4. Update Qtest

Improvement Plans

1. Fuzz virtio-net, virtio-sock, virtio-*
2. Fuzz PCI
3. Enhance start corpus
4. Update Qtest
5. Verify test cases

05

Useful Links

Links

1. **QEMU changes (afl-fuzz branch)**: https://github.com/yandex/qemu/tree/afl-fuzz
2. **AFL changes (qemu-fuzz branch)**: https://github.com/yandex/AFL/tree/qemu-fuzz

Yandex Cloud

Fuzz the Code for Fun and Profit!

Dima Stepanov

Software Engineer, Yandex.Cloud Hypervisor Team
dimastep@yandex-team.ru