Don’t trust your USB

How to find bugs in USB device drivers

OpenSource Security Ralf Spenneberg

Motivation

- compromise systems via USB
- find bugs, fix or exploit them :-)

BadUSB (2014)

Facedancer (2012)

USB Fuzzing for the Masses (2011)

QEMU

Sergej Schumilo - schumilo@fh-muenster.de

Impact

- CVE-2013-1285

The USB kernel-mode drivers in Microsoft Windows XP SP2 and SP3, Windows Server 2003 SP2, Windows Vista SP2, Windows Server 2008 SP2, R2, and R2 SP1, Windows 7 Gold and SP1, Windows 8, and Windows Server 2012 do not properly handle objects in memory, which allows physically proximate attackers to execute arbitrary code by connecting a crafted USB device, aka "Windows USB Descriptor Vulnerability," a different vulnerability than CVE-2013-1286 and CVE-2013-1287.

Publish Date : 2013-03-12 Last Update Date : 2013-11-02

- CVE-2013-1680

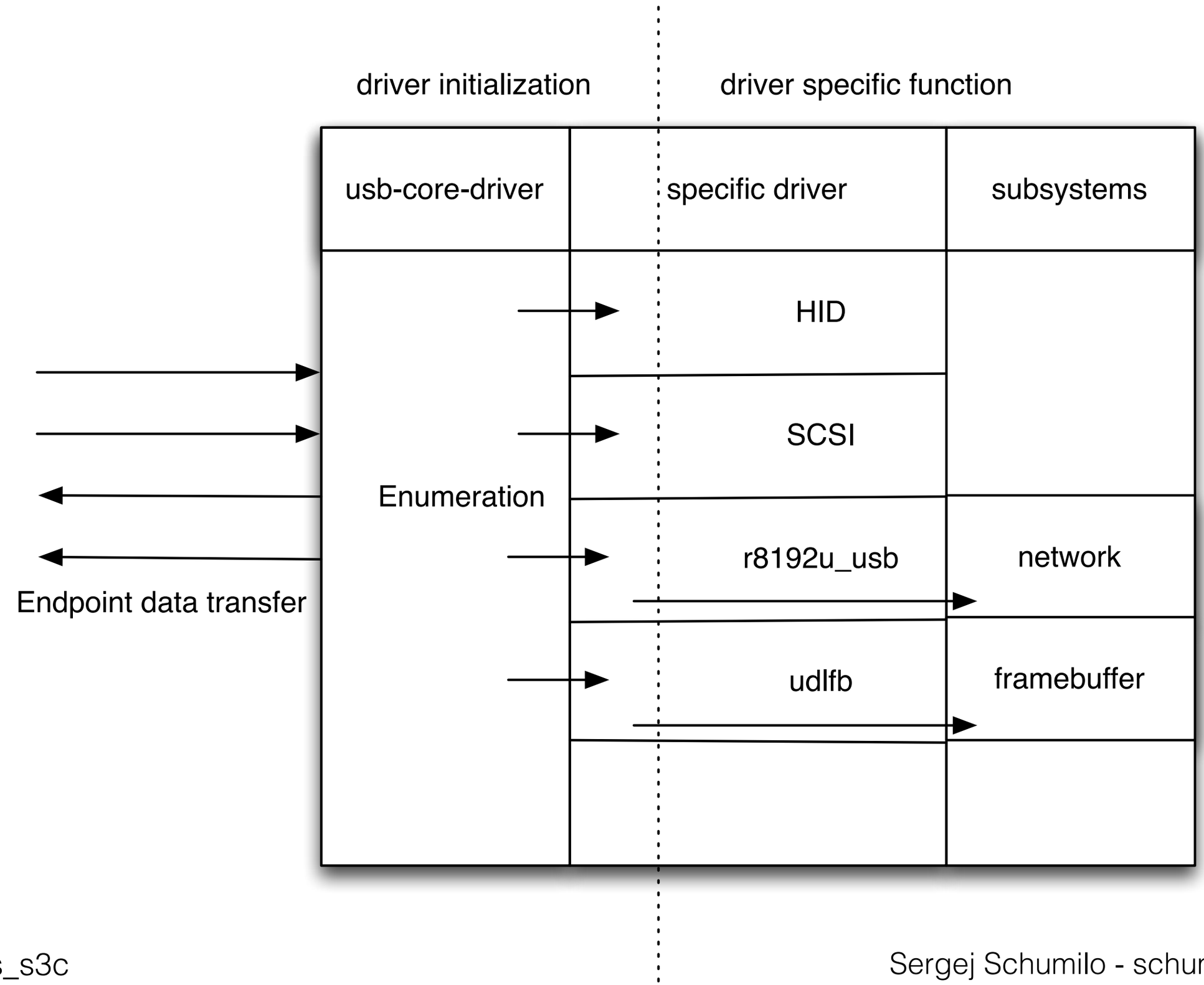
Heap-based buffer overflow in the wdm_in_callback function in drivers/usb/class/cdc-wdm.c in the Linux kernel before 3.8.4 allows physically proximate attackers to cause a denial of service (system crash) or possibly execute arbitrary code via a crafted cdc-wdm USB device.

Publish Date : 2013-03-22 Last Update Date : 2014-04-19

Plug and Root?

Darrin Barrall and David Dewey 2005 at Black Hat

USB in a nutshell



Enumeration

usb core driver

```
fuzzing — bash — 88×13
[   85.612222] usb 1-1: new full-speed USB device number 34 using xhci_hcd
[   85.807599] usb 1-1: New USB device found, idVendor=0bb4, idProduct=0a09
[   85.809178] usb 1-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[   85.810777] usb 1-1: Product: vUSBf_fuzzing_device
[   85.811522] usb 1-1: Manufacturer: dummy_manufacturer
[   85.812391] usb 1-1: SerialNumber: %%%%dummy%%%%
[   85.838423] usbcore: registered new interface driver ipaq
[   85.842418] usbserial: USB Serial support registered for PocketPC PDA
[   85.846361] ipaq 1-1:1.0: PocketPC PDA converter detected
[   85.853804] usb 1-1: PocketPC PDA converter now attached to ttyUSB0
[   86.603790] usb 1-1: USB disconnect, device number 34
[   86.609620] ipaq ttyUSB0: PocketPC PDA converter now disconnected from ttyUSB0
[   86.620612] ipaq 1-1:1.0: device disconnected
```

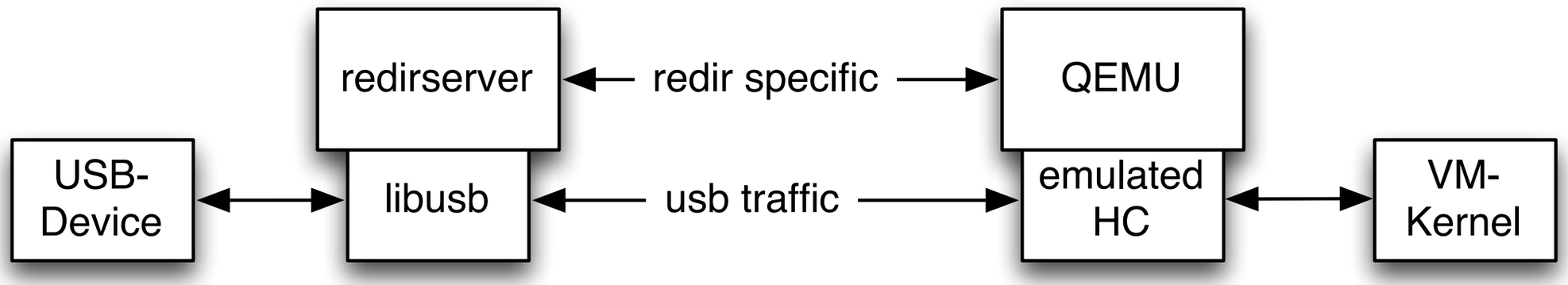
specific device driver

Vendor: 0x0bb4 (HTC)
Product: 0x0a09 (PocketPC Sync)

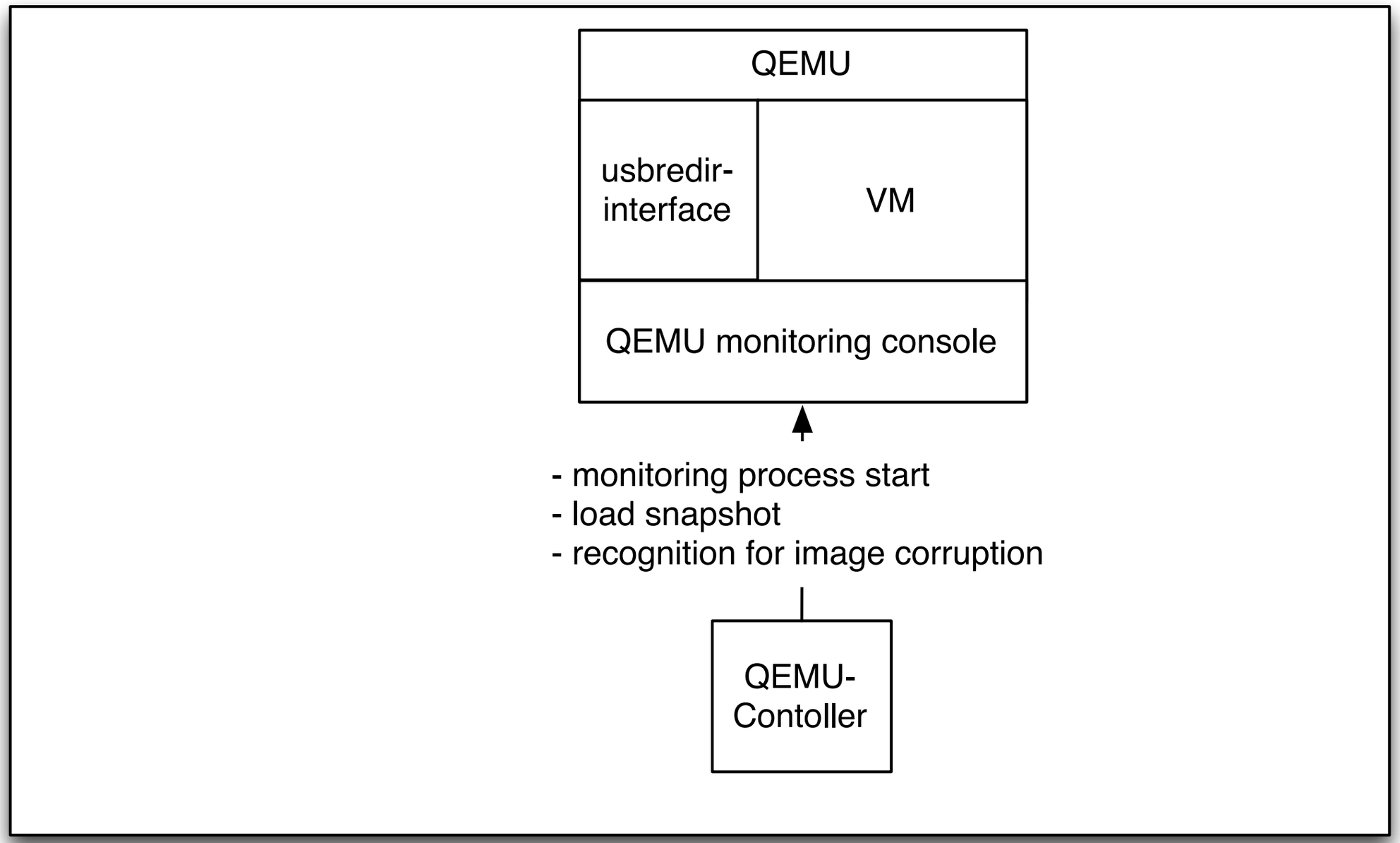
vUSBf Framework

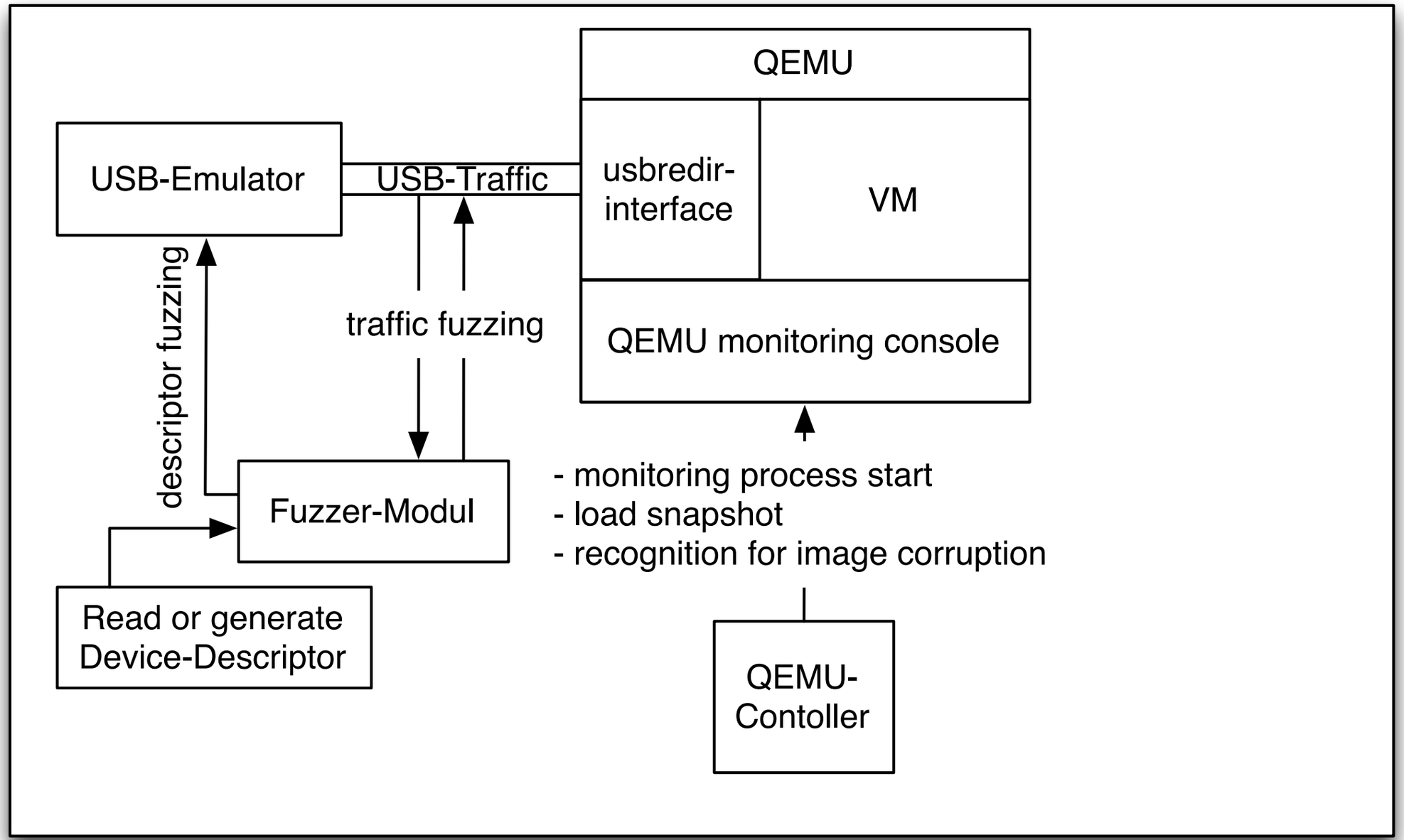
- usage of QEMU and KVM for virtualization
- usage of USB Redirection interface for USB data injection
  - send usb traffic through TCP, UDP or Unix sockets
  - wrap usb data into USB Redirection protocol headers
  - it supports USB 1.0, 1.1, 2.0, 3.0

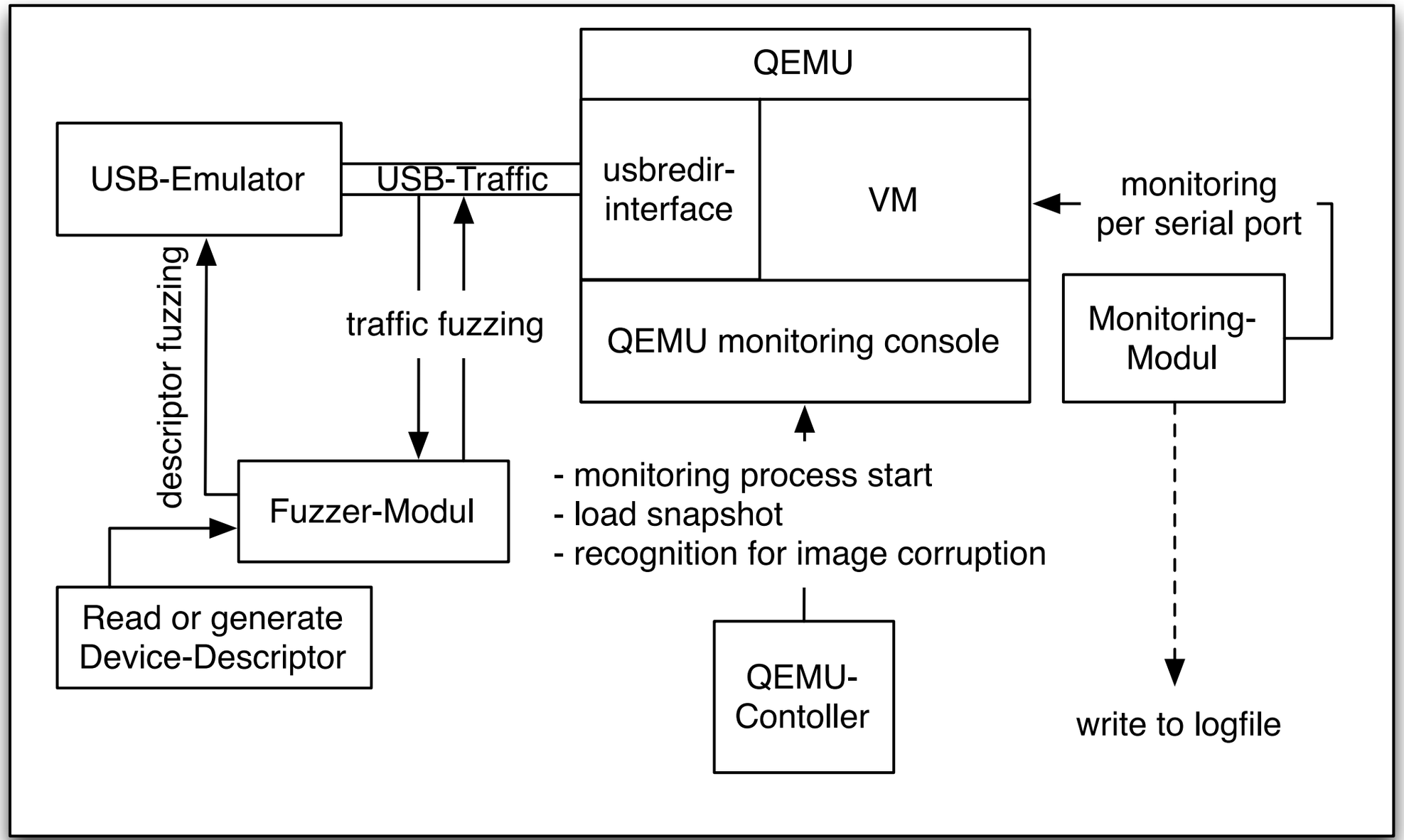
USB Redirection



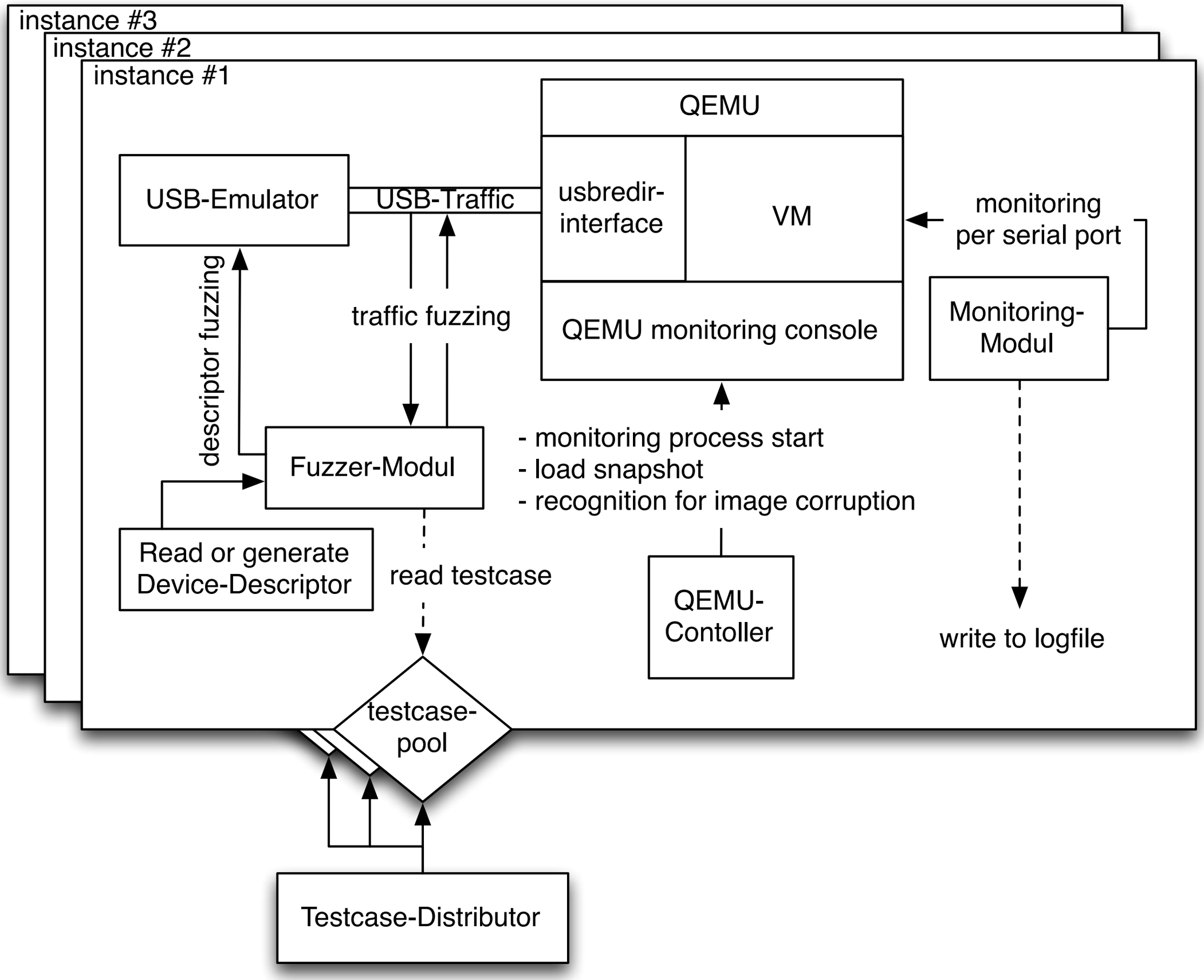
- provides remote USB-device
- part of the SPICE suite


QEMU
usbredir-
interface
VM
QEMU monitoring console
- monitoring process start
- load snapshot
- recognition for image corruption
QEMU-
Contoller


QEMU
USB-Emulator
USB-Traffic
usbredir-
interface
VM
descriptor fuzzing
traffic fuzzing
QEMU monitoring console
Fuzzer-Modul
- monitoring process start
- load snapshot
- recognition for image corruption
Read or generate
Device-Descriptor
QEMU-
Contoller



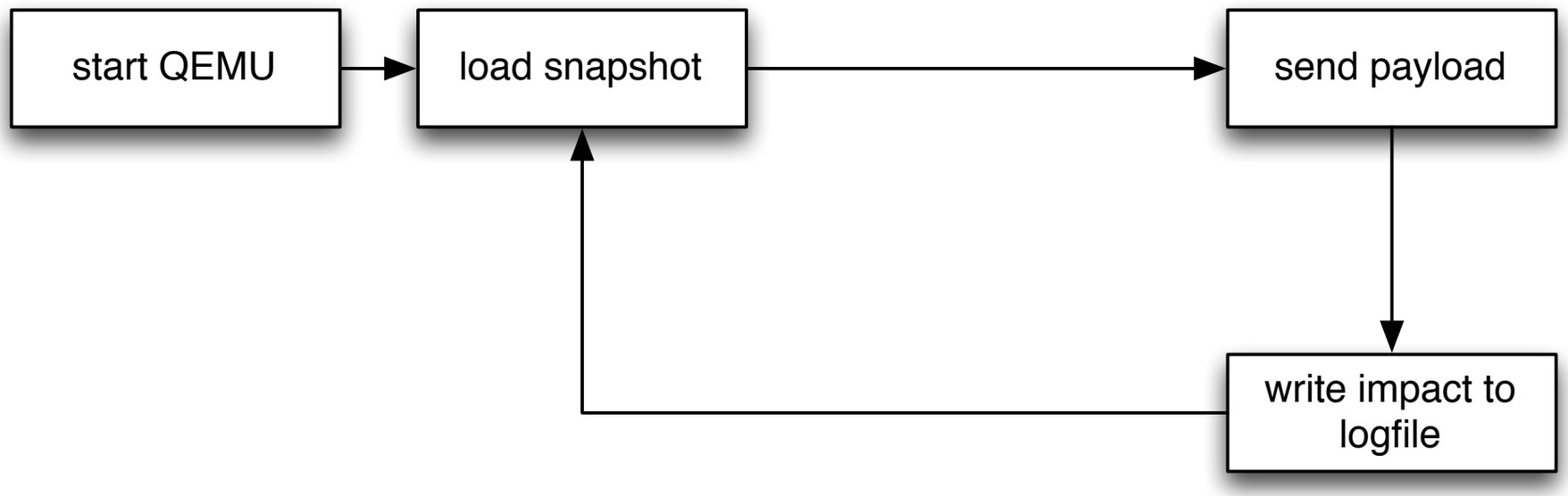
Sergej Schumilo - schumilo@fh-muenster.de



Sergej Schumilo - schumilo@fh-muenster.de

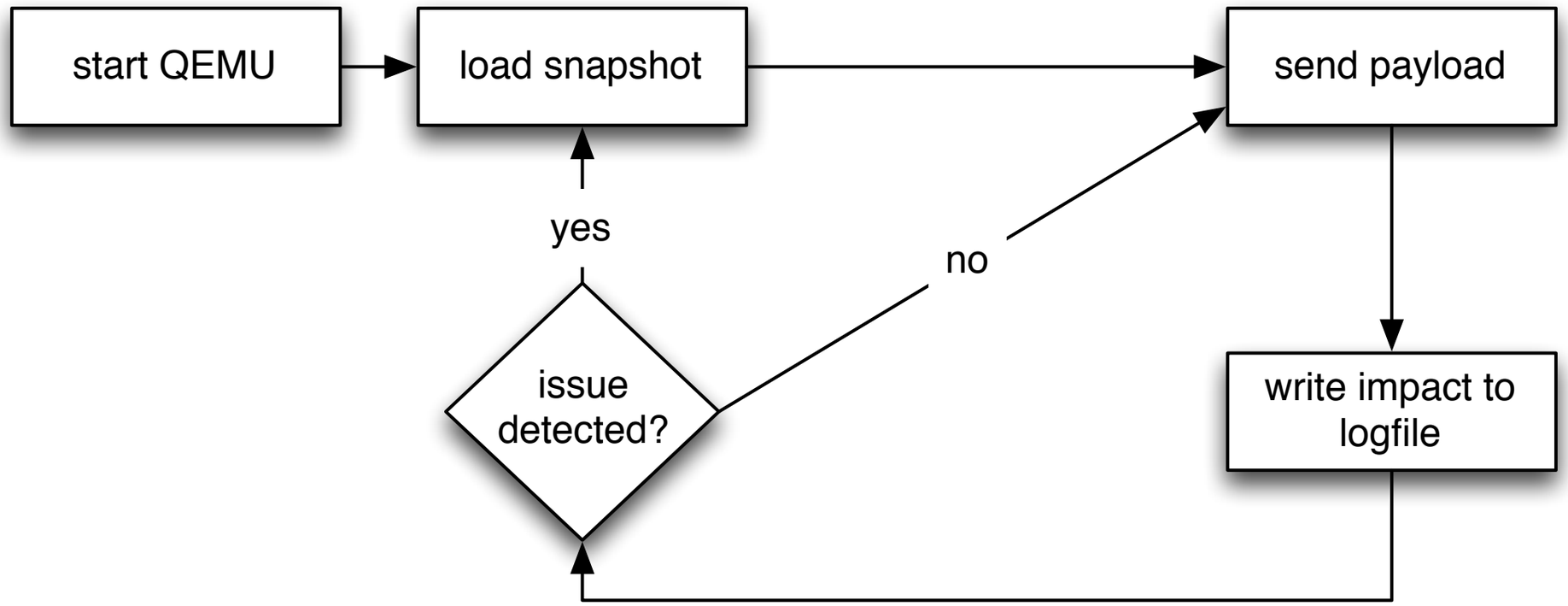
vUSBf

reload mode



vUSBf

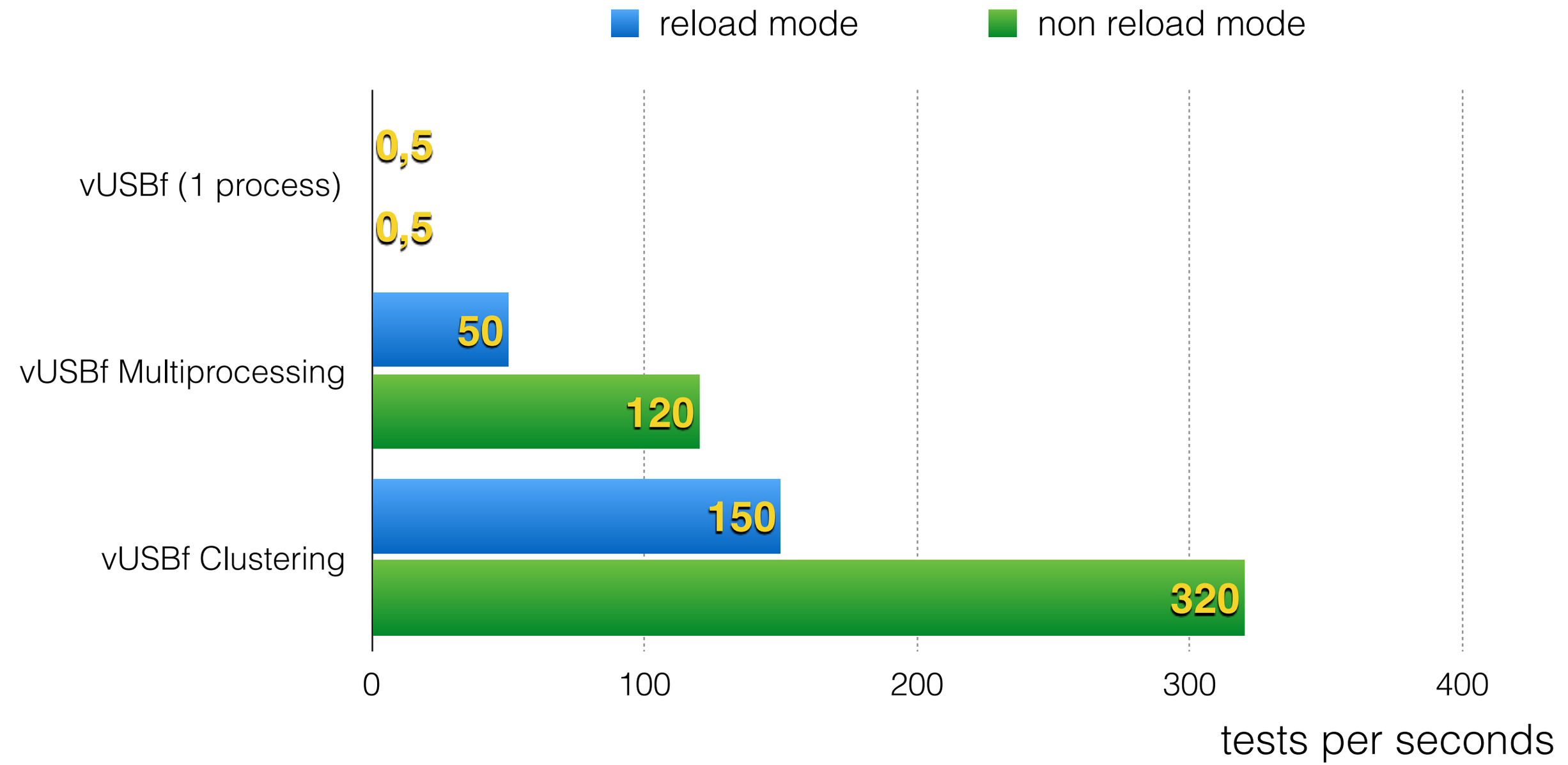
Non-reload mode



Reproducibility

- test cases are related to unique IDs
- file export for a sequence of test cases
- offers high reproducibility in combination with snapshots

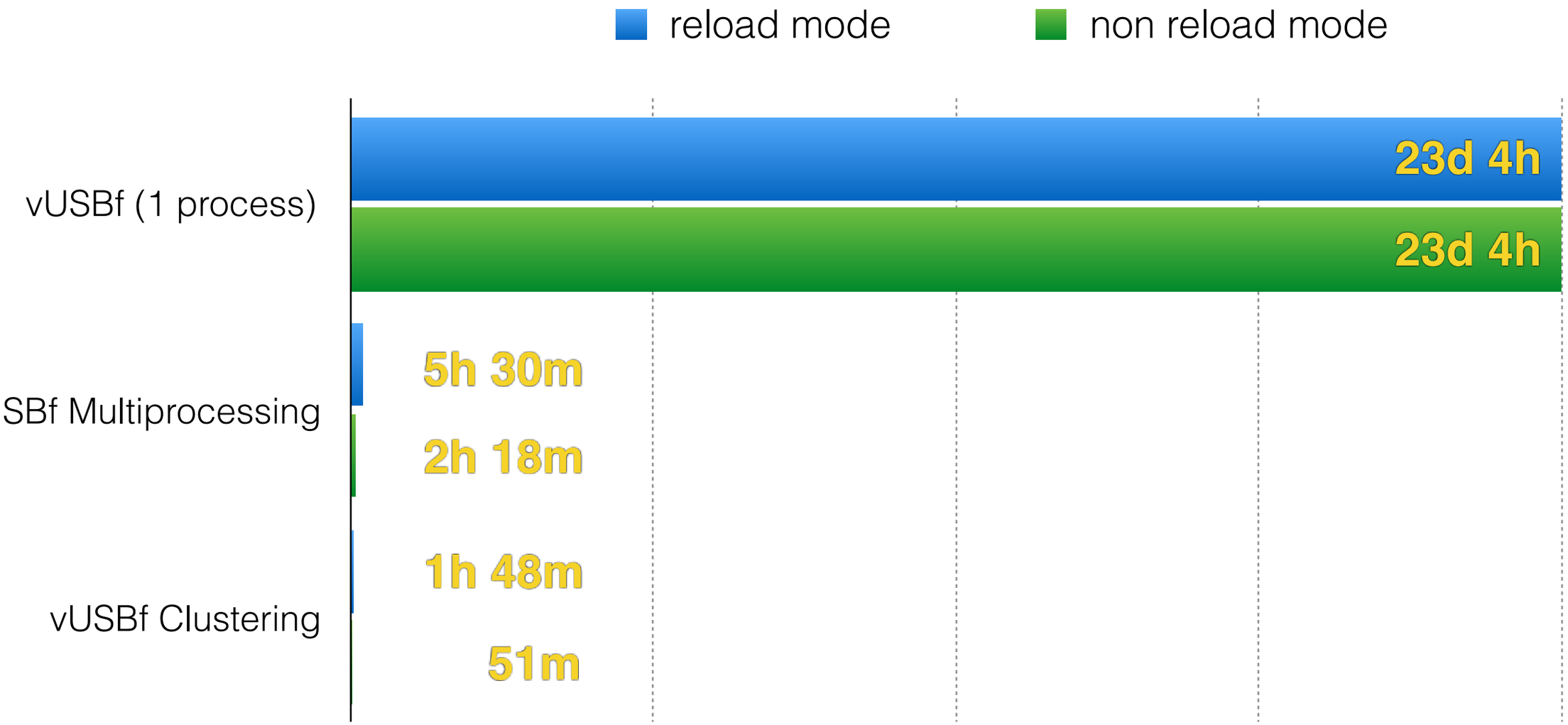
vUSBf Performance



Sergej Schumilo - schumilo@fh-muenster.de

vUSBf Performance

runtime for 1 million tests:



Demo

Challenges

- Linux monitoring is simple!
  - other OS?
- investigation / fixing of bugs
- additional specific USB-emulators

```
BUG: Bad rss-counter state mm:ffff880002a13c00 idx:1 val:2
BUG: Bad rss-counter state mm:ffff8800057e4a80 idx:1 val:2
BUG: Bad rss-counter state mm:ffff8800057e5180 idx:1 val:2
BUG: Bad rss-counter state mm:ffff8800057e5880 idx:1 val:2
BUG: Bad rss-counter state mm:ffff880006390e00 idx:1 val:2
BUG: Bad rss-counter state mm:ffff880006391500 idx:1 val:2
BUG: Bad rss-counter state mm:ffff880006391c00 idx:1 val:2
BUG: Bad rss-counter state mm:ffff880007d0c700 idx:1 val:2
BUG: unable to handle kernel NULL pointer dereference at 0000000000000002
BUG: unable to handle kernel NULL pointer dereference at 0000000000000003
BUG: unable to handle kernel NULL pointer dereference at 0000000000000004
BUG: unable to handle kernel NULL pointer dereference at 0000000000000008
BUG: unable to handle kernel NULL pointer dereference at 000000000000000f
BUG: unable to handle kernel NULL pointer dereference at 0000000000000042
BUG: unable to handle kernel NULL pointer dereference at 0000000000000043
BUG: unable to handle kernel NULL pointer dereference at 0000000000000044
BUG: unable to handle kernel NULL pointer dereference at 0000000000000046
BUG: unable to handle kernel NULL pointer dereference at 0000000000000048
BUG: unable to handle kernel NULL pointer dereference at 0000000000000058
BUG: unable to handle kernel NULL pointer dereference at 0000000000000068
BUG: unable to handle kernel NULL pointer dereference at 0000000000000070
BUG: unable to handle kernel NULL pointer dereference at 0000000000000098
BUG: unable to handle kernel NULL pointer dereference at 00000000000000a0
BUG: unable to handle kernel NULL pointer dereference at 00000000000000a8
BUG: unable to handle kernel NULL pointer dereference at 00000000000000c0
BUG: unable to handle kernel NULL pointer dereference at 00000000000000e0
BUG: unable to handle kernel NULL pointer dereference at 0000000000000190
BUG: unable to handle kernel NULL pointer dereference at 0000000000000260
BUG: unable to handle kernel NULL pointer dereference at 00000000000002a0
BUG: unable to handle kernel NULL pointer dereference at 000000000000037f
BUG: unable to handle kernel NULL pointer dereference at           (null)
BUG: unable to handle kernel paging request at 00000000000232e4
BUG: unable to handle kernel paging request at 0000001d0101046c
BUG: unable to handle kernel paging request at ffffeb88000085c0
BUG: unable to handle kernel paging request at ffffebe000000040
BUG: unable to handle kernel paging request at ffffebe1f4000000
BUG: unable to handle kernel paging request at fffffffffffffd8
drm_kms_helper: panic occurred, switching back to text console
Fixing recursive fault but reboot is needed!
kernel BUG at /build/buildd/linux-3.13.0/mm/slub.c:3365!
kernel BUG at /build/buildd/linux-3.13.0/net/core/dev.c:6385!
Kernel panic - not syncing: Attempted to kill init! exitcode=0x00000009
Kernel panic - not syncing: Attempted to kill init! exitcode=0x0000000b
Kernel panic - not syncing: Fatal exception in interrupt
systemd-udevd[1177]: '/bin/sh -c 'test -e /sys//devices/pci0000:00/0000:00:04.0/usb1/1-1/power/level && echo on > /sys//devices/pci0000:00/0000:00:04.0/
usb1/1-1/power/level'' [1241] terminated by signal 11 (Segmentation fault)
systemd-udevd[1177]: 'mtp-probe /sys/devices/pci0000:00/0000:00:04.0/usb1/1-1 1 46' [1246] terminated by signal 11 (Segmentation fault)
systemd-udevd[1177]: 'mtp-probe /sys/devices/pci0000:00/0000:00:04.0/usb1/1-1 1 53' [1272] terminated by signal 11 (Segmentation fault)
systemd-udevd[1177]: 'mtp-probe /sys/devices/pci0000:00/0000:00:04.0/usb1/1-1 1 74' [1351] terminated by signal 11 (Segmentation fault)
systemd-udevd[1181]: '/bin/sh -c 'test -e /sys//devices/pci0000:00/0000:00:04.0/usb1/1-1/power/level && echo on > /sys//devices/pci0000:00/0000:00:04.0/
usb1/1-1/power/level'' [1361] terminated by signal 11 (Segmentation fault)
systemd-udevd[1181]: 'mtp-probe /sys/devices/pci0000:00/0000:00:04.0/usb1/1-1 1 91' [1362] terminated by signal 11 (Segmentation fault)
usb_modeswitch_[1192]: segfault at 0 ip 0000000000401e1f sp 00007fff25c4fc00 error 4 in usb_modeswitch_dispatcher[400000+a000]
usb_modeswitch_[1198]: segfault at 0 ip 0000000000401e1f sp 00007fffcce90dc0 error 4 in usb_modeswitch_dispatcher[400000+a000]
usb_modeswitch[1231]: segfault at 10 ip 00000000004023ad sp 00007fff40de2de0 error 4 in usb_modeswitch[400000+c000]
usb_modeswitch[1234]: segfault at 10 ip 00000000004023ad sp 00007fff32c1ad30 error 4 in usb_modeswitch[400000+c000]
usb_modeswitch[1265]: segfault at 10 ip 00000000004023ad sp 00007fff11aad0a0 error 4 in usb_modeswitch[400000+c000]
usb_modeswitch_[1271]: segfault at 0 ip 0000000000401e1f sp 00007fff8f34a0f0 error 4 in usb_modeswitch_dispatcher[400000+a000]
usb_modeswitch[1293]: segfault at 10 ip 00000000004023ad sp 00007fffc2d26b50 error 4 in usb_modeswitch[400000+c000]
```

Conclusion:

vast amount of bugs :-)

verified using facedancer

USB-fuzzing in practical time frames

Questions?

www.github.com/schumilo

Comrade-in-arms are welcome :-)