Fusil the fuzzer

Victor Stinner

FOSDEM, february 2009

Fuzzing principles

- Isolate input vectors
- Inject faults
- Watch the software reactions

Fuzzing goals

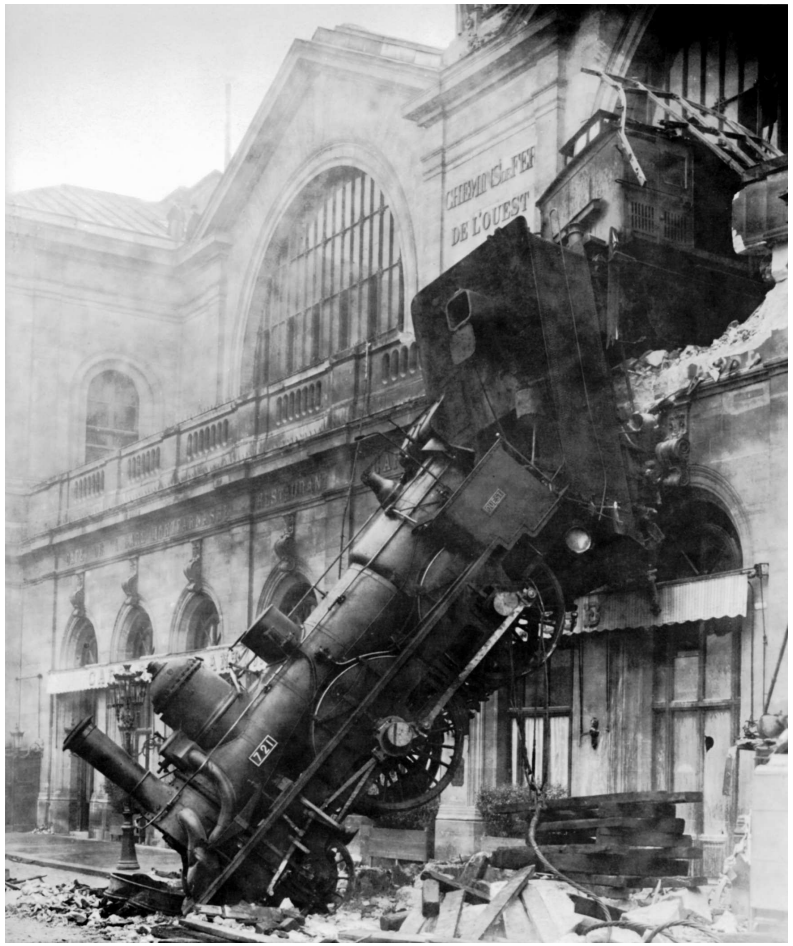
- Find bugs
- Test software stability
- Offender’s point of view

Advantages of the fuzzing

- Simple
- Fast
- Efficient
- Free or closed software

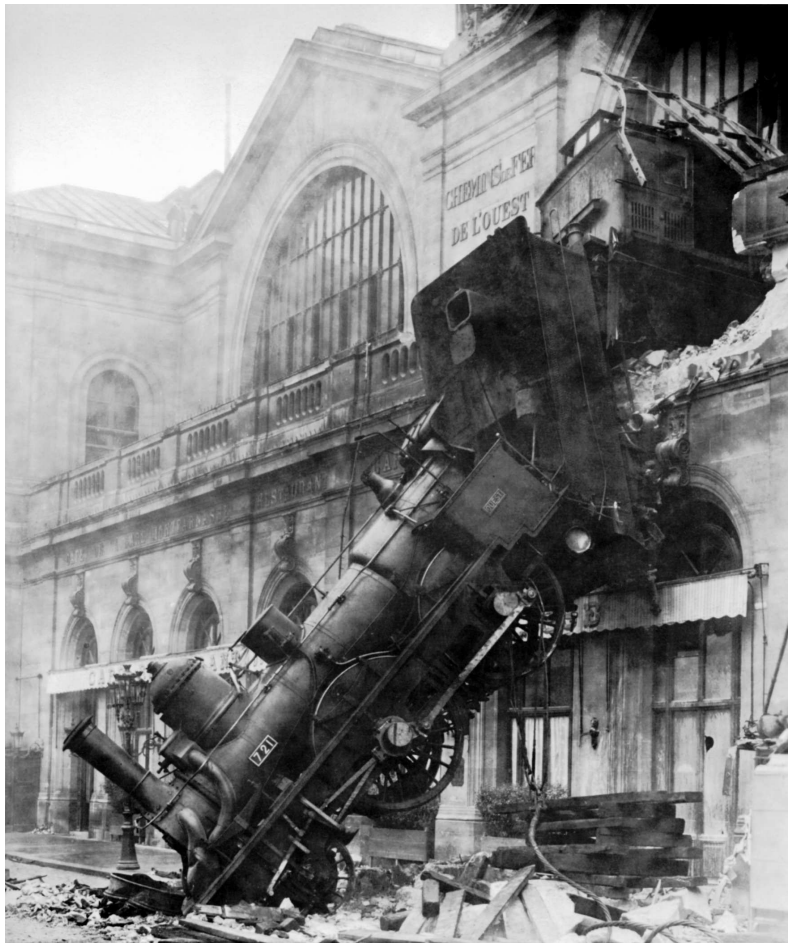
Fusil the fuzzer

- Python library
- 20+ fuzzers
- Found bugs in ClamAV, Gimp, GNU libc, Python, ...
- Test command line programs



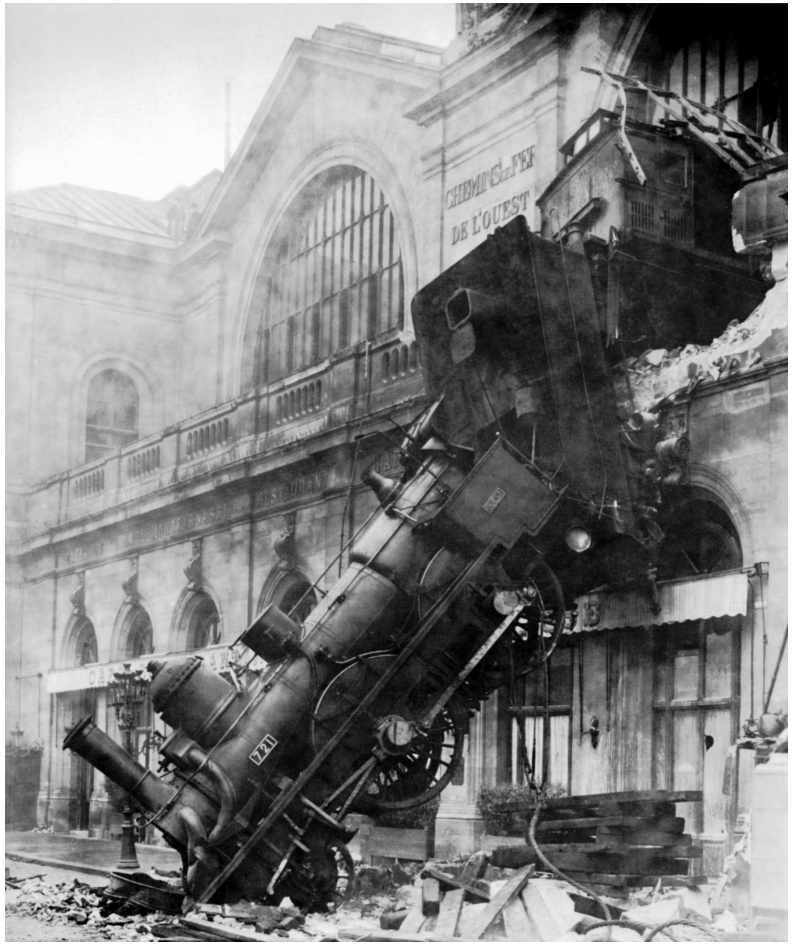
Input vectors

- Command line
- Files
- Environment variables
- TCP sockets



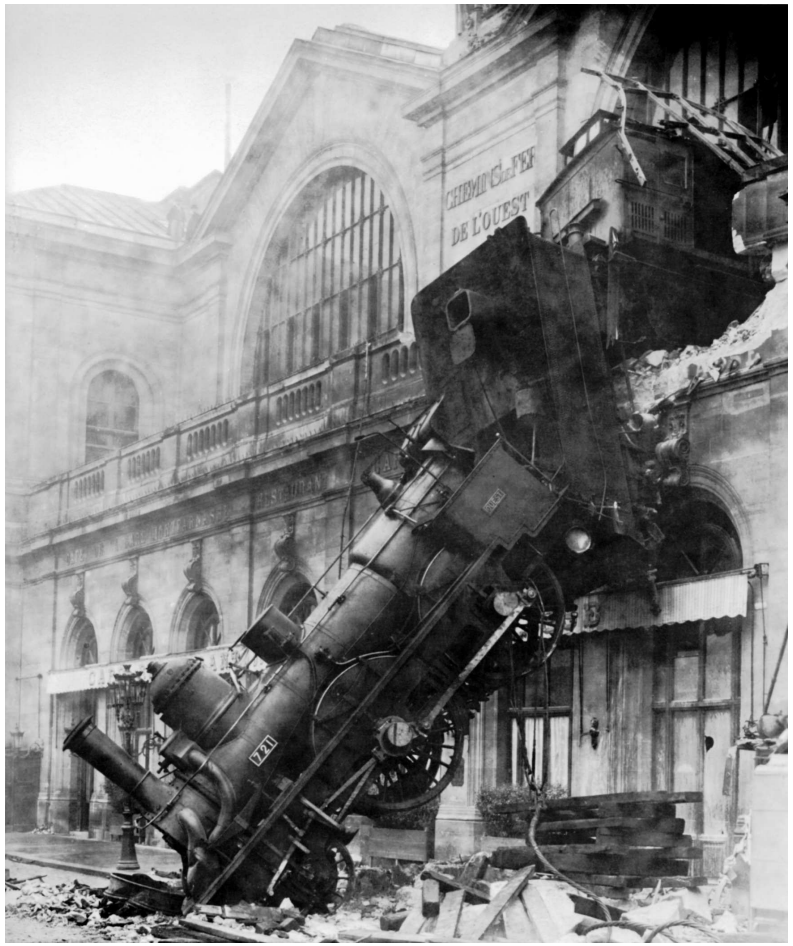
The different probes

- Standard output / logs
- Process exit status
- Network ping
- Integrated debugger (ptrace)



Probes score system

- Weighted sum of probe scores
- -100%: inputs rejected
- 0%: nothing special
- 100%: success (crash)



Generate pure random data

- nc host port < /dev/urandom
- Crashes softwares with no QA
- ... with no security
- ... with few users

Inject faults in valid data

- Invert bits, replace bytes, etc.
- Truncate file
- Crashes all audio/video players
- Easiest technique

Generate data using a model

- Generate random data
- ... respecting the specifications
- Deeper tests
- Needs specifications

Example of the Python fuzzer

- List Python modules
- List module functions
- Random function calls
- Random argument number and types

Application fuzzers

- ClamAV (antivirus)
- Firefox
- Gstreamer, mplayer, VLC (audio/video)
- Image Magick

Library fuzzers

- gettext (i18n)
- poppler (PDF): Evince and Kpdf
- printf() of the libc

Programming languages fuzzers

- Python
- PHP
- Random functions, random arguments

Demonstration

- Crash Python
- Replay the crash
- Analyze the crash

Criticity of an error

- Denial of service
- Lost data
- Permanent error
- Difficulty to reproduce the error

Bug or vulnerability?

- Steal data (read)
- Modify files (write)
- Inject code (execute)

Editors' answers

- *We don't care!*
- Apply patches
- Interested to test Fusil
- Reflect the project quality

Ideas for new faults

- failmalloc, /dev/full
- Time jumps
- Send signals
- Network fuzzers

Ideas of new probes

- Code coverage (gcov, DynamoRio)
- Memory fault (Valgrind, Purify)

Other fuzzer libraries

- Sulley, Peach, EFS
- Bunny, Flayer, Scapy
- zzuf, PROTOS, SPIKE
- ...

Why using Fusil?

- Keep "success" files
- Replay script
- Crash informations
- Safety

Questions ?

- Fusil: GPLv2
- Linux, FreeBSD, (Windows)
- Written in Python 2.5
- http://fusil.hachoir.org

Picture sources

- http://commons.wikimedia.org
- File:Train_wreck_at_Montparnasse_1895.jpg
- File:Bundesarchiv_Bild_102-12503,_Autounfall.jpg
- File:Bundesarchiv_Bild_102-10248,_Mecklenburg,_Autounglück.jpg
- File:B-24_Kopfstand.jpg
- File:Men_inspecting_wreckage_of_first_Toronto_airplane_crash.jpg
- File:AralShip.jpg