REVEN-64-v2
Deterministic full-system analysis for x64

Label DGA-RAPID 182906025 – 2018-02-21

24 months – extended to 2021-02-20

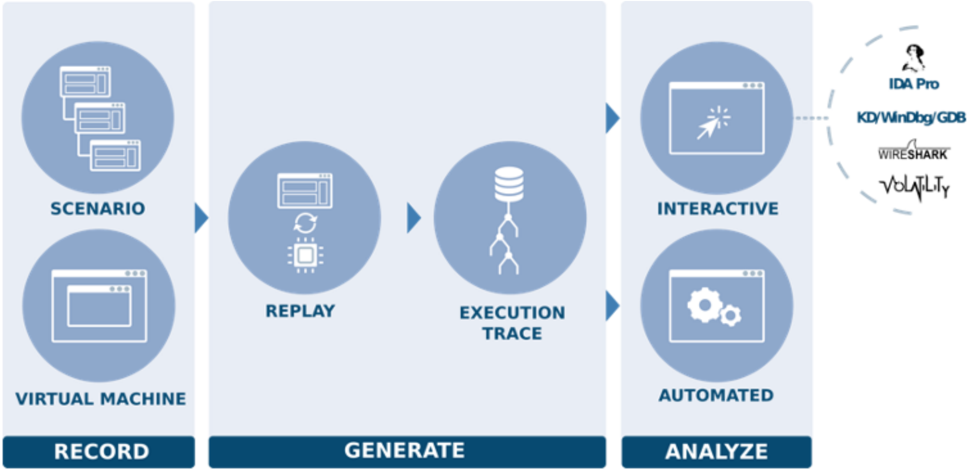
Louis Dureuil
ldureuil@tetrane.com (@lodurel)

RESSI 2020 - 2020-12-18

REVEN-64-v2 Project

- Tetrane: software editor in Mâcon, proposes the REVEN solution
- Motivation: System-wide timeless analysis
- Existing solution: REVEN v1
- Challenges of project:
  - Address Intel x86 64-bit architecture
  - Improve accuracy of record/replay
  - Build more advanced analysis algorithms from timeless analysis
- Approach:
  - Platform generic wrt Recorder/Replayer, extensible to other architectures (ARM)
  - API to accurately model the Execution Trace (Context/Transition)
  - Use-case oriented approach to analysis: whole trace algorithms (search in memory, memory history), data flow analysis (taint), vulnerability detection

REVEN


SCENARIO
VIRTUAL MACHINE
RECORD
REPLAY
EXECUTION TRACE
GENERATE
INTERACTIVE
AUTOMATED
ANALYZE
IDA Pro
KD/WinDbg/GDB
WIRESHARK
VoLaTiLiTy

Focus on: our research on Taint analysis algorithm

How to go from system-wide timeless analysis to forward and **backward**, **interprocess** taint analysis?

. . . 3 man-years later

- Make sense of the x86 64-bit instructions
  - Move away from our bespoke symbolic representation
  - Lift to LLVM IR
- Propagate data flow
  - Custom LLVM propagation algorithms
- Scale to billions of instructions
  - Memory history optimization
- Validate taint correctness
  - 230+ manually crafted unit tests
  - Integration tests: validate inter-process tainting
  - Scaling tests: use on real-world execution traces

REVEN-64-v2: Some Results and Future Work

Results

- A generic wrt record/replay x86/x64 system-wide timeless analysis platform
- Taint analysis instrumental for analysis: Reproduce and analyze modern CVEs with REVEN
  - CVE-2020-16898, CVE-2020-17087, CVE-2019-1347, CVE-2019-0708 (BlueKeep), ...
- Contributions to open-source: remill (https://github.com/lifting-bits/remill) (x86_64 to LLVM IR lifter)
- Use-After-Free detection in CVE based on taint analysis

Future Work

- Ongoing research to add higher-level algorithms: Vulnerability detection, System monitoring
- **AArch64** support
  - prototype with Qemu-RR, which record/replay framework?
  - Next: handle memory contexts, real-world traces (Android...), address ARM specificities wrt Intel architectures
- Advanced taint: taint graph, pointer tainting
- Application-oriented analysis

Some pointers to go further

- https://www.tetrane.com/
- Articles on some applications of REVEN: https://blog.tetrane.com
- REVEN playgrounds: https://www.tetrane.com/demos.html
- PatchGuard whitepaper: https://blog.tetrane.com/downloads/Tetrane_PatchGuard_Analysis_RS4_v1.00.pdf
- Our open-source contributions: https://github.com/tetrane/tetrane-oss

Questions?

ldureuil@tetrane.com (@lodurel)