TROOPERS TRAININGS Make the world a safer place.

HMTS HM Training Solutions

Training:

Fuzzing source-code & binary-only targets like a pro

Date of the training: **March 16-17, 2020** in Heidelberg, Germany

Book now using the voucher code: **TR20_HMTS** and save an additional 5% of the current valid rate of any package!

Overview

On day 1 we will see how to effective fuzz targets for which we have source code based on afl++ and libfuzzer. On day 2 we will target binary-only programs based on Qemu, afl-dyninst and Unicorn.

What is effective, how can we use structure information when fuzzing, how to perform large fuzzing campaigns, does intel-pt help us, and what about other platforms like ARMv7, AARCH64 and PowerPC - this training will give you the answers! The goal of this class it to perform software security fuzzing (black box, grey box, white box) for finding security vulnerabilities.

For source code (white box) we will take a look at afl++ and libfuzzer, understand how they work, prepare our targets in an optimized manner and run them against real-world targets. Then we have a look at the various mutators, schedulers and custom options and what afl compatible fuzzer variants (important!) can help us to make the fuzzing better.

ERNW Insight GmbH
Carl-Bosch-Str. 4
69115 Heidelberg
BIC: GENODE61HD3

Tel. +49 – 6221 – 48 03 90
Fax +49 – 6221 – 41 90 08
Ust-ID DE 304 445 827
IBAN: DE84 6729 0100 0067 1236 03

Managing Director:
Kerstin Schröter

In the next step we will look at how we can add data structures to the fuzzing to make it effective. Finally, we will plan a comprehensive large fuzzing campaign for a target - how many instances to run, with which mutators and schedulers and afl-variants, and when to replace them. So we don't lose time watching fuzzing UI stats, we will use time while fuzzing to talk about crash analysis, checking the code coverage of our fuzzing and talking about cool afl features.

For binary-only targets (black/grey box) we will see what options we have to fuzz those targets with input driven feedback. Our main tools will be Qemu and afl-dyninst together with afl++ to fuzz targets, but we will also have a look at good alternatives to these. Additionally, we will have an introduction to the Unicorn engine for full systems emulation.

With this on our belt we will even be able to fuzz targets from other platforms like ARM on a fast Intel CPU. Additionally, we will talk (and try out) other solutions like intel-pt and symbolic execution engines like DynamoRIO and Pintools for fuzzing. The focus is on Linux on Intel processors but *BSD and Windows, and ARM/AARCH64/… are covered as well - for source code and binary-only fuzzing.

Day 1 - Source code

- Introduction to afl++;
- How to prepare targets for afl++;
- Running afl++ effectively;
- Introduction to libfuzzer;
- How to effectively code API tests with libfuzzer;
- How to run libfuzzer effectively;
- Structure fuzzing with afl++ and libfuzzer;
- Customer mutators and special input requirements;
- How to fuzz network services in targets;
- afl-compatible alternatives;
- Verifying fuzzing code coverage;
- Tips, tricks, features for afl++;
- Setting up a comprehensive, effective fuzzing campaign for a target.

Day 2 - Binary-only

- Possible sources for fuzzing feedback;
- Fuzzing with Qemu;
- Fuzzing with afl-dyninst;
- Alternatives for feedback driven fuzzing binary-only targets;
- Introduction to the Unicorn engine;
- Introduction to symbolic execution for fuzzing;
- Using winafl on Windows to fuzz DLLs;
- “I have 256 crashes - now what?” - we will answer this :-)!

Prerequisites

Students should have a good experience with Linux and be comfortable coding in C/C++, also basic debugging experience is helpful.

Requirements

Students must bring a laptop with Linux installed (VM or native), Kali Linux is highly recommended. For the Windows components, a Windows VM is recommended for those interested. The trainer is one of the maintainers of afl++. Expect the training to go way past the 17:00 deadline!

About the Trainer

Marc “van Hauser” Heuse started with his security research in 1993 even before finishing school. He is well known for being the founder of the security research group The Hacker’s Choice (www.thc.org) and for his security tools like hydra, thc-ipv6, THC-SCAN, SuSEfirewall and many others. In his career he was team leader and manager at KPMG and n.runs and for over 10 years he is now working as an independent security researcher now at mh-sec, focusing on automotive penetration testing and other embedded platforms. Currently he is working using binary instrumentation and fuzzing for his automotive security projects and research and is the co-author of the blackbox feedback fuzzing tool afl-dyninst and the afl follow-up projects afl++.

Booking

Recommended online booking of trainings through:
https://troopers.de/tickets/

Voucher code: TR20_HMTS
Using this voucher code automatically gives you an additional 5% off the current valid price! You can register with this code until seats have run out.

Contact

Need assistance? Don't hesitate to call us. We are fluent in English and German. +49 151 16228365 or info@troopers.de

Booking is also possible offline through your trusted partner:

HM Training Solutions, Falkenstrasse 6, 63820 Elsenfeld, Germany
+49 6022 508200 / info@hmtrainingsolutions.com
+49 6022 5089999 / www.hmtrainingsolutions.com