Crypto for developers

2014.01.08

Jean-Philippe Aumasson (@veorq)

The training

This training is about cryptography for non-cryptographers: how to make sure you choose the right API and algorithms, and how to avoid bugs in your own implementations. Indeed, cryptography is hard to get right in software, and seemingly minor software bugs often have a major impact. Bad choices of algorithms or protocols are another common source of failures.

The goal of this training is thus to provide the knowledge and tools in order to

- Make the right choices of API and algorithms according to engineering constraints.
- Avoid common implementation errors in implementations of cryptography.
- Perform tests that will maximize the chances of detecting bugs.
- Identify vulnerabilities when reviewing cryptographic code or specifications.

The training is about 7 hours of lecture with discussions and code-review exercices. If all participants speak French I will present in French, otherwise in English.

Teaching philosophy

The most efficient way to learn defense is to review previous attacks—if you understand how to exploit a bug and the potential consequences, you are unlikely to leave it in your code. The training will thus include code-review exercices where participants will be encouraged to ask questions and challenge the trainer.

Review of vulnerabilities will be supported by the adequate theory so as to understand the fundamental reasons of the problem, and to be able to spot a similar vulnerability when it appears in a different form.

Target audience

This training is mainly intended to software developers, but also to penetration testers, security researchers, sysadmins, students, information security professionals, and anyone interested in cryptography. The content and pace of the training will be adapted to the audience.

What you will learn

During this training, you will learn

- Security notions like semantic security and perfect forward secrecy.
- How to use strong randomness and how to test a pseudorandom generator.
- How to choose a safe API (with examples from OpenSSL, NaCl, etc.).
- How to defend against the most common attacks (including timing attacks and oracle attacks), illustrated with examples of AES, RC4, RSA.
- How to properly configure crypto in HTTPS and SSH services.
- How to best protect passwords using cryptography.
- How to choose the right algorithm and key size, how to test crypto algorithms.
- How to use elliptic-curve cryptography instead of RSA, and for which benefits.

Prerequisites

Participants only need to know the basics of cryptography: the notions of confidentiality, integrity, and authenticity, as well as the principle of public-key encryption and signature. It is expected that participants have basic notions of networking and processor architecture, and that they are familiar with the C and Python languages.

The trainer

Jean-Philippe Aumasson is Principal Cryptographer at Kudelski Security, Switzerland. He received a PhD in cryptography from EPFL in 2009, and has authored more than 30 research articles in the field of cryptography and cryptanalysis.

Jean-Philippe is known for designing the cryptographic functions BLAKE (one of 5 SHA-3 finalists), SipHash (used in Perl, Ruby, etc.), BLAKE2 (used in WinRAR, etc.). He has talked at security conferences including Hashdays, Chaos Communications Congress, PasswordsCon, Black Hat. In 2013 he initiated the Cryptography Coding Standard and the Password Hashing Competition, which are open collaborative projects aimed to improve the overall state of security.

Pricing

The day of training including digital copies of all material (slides and code) is CHF 500, and CHF 200 for students. We can only accept up to 20 participants.