Introducing SEB Server

Daniel R. Schneider, Project Leader and Lead Developer SEB
Educational Development and Technology (LET), ETH Zurich

10th SIG e-Assessment Workshop, June 20, 2018

Swiss MOOC Service (SMS)

- SEB Server is part of swissuniversities CUS-P5 project "Swiss MOOC Service" of EPFL, USI, SUPSI, HES-SO and ETHZ

- National MOOC platform for the Swiss institutions of higher education. Goals:
  - Establishing the Open edX based MOOC platform, running on server infrastructure in Switzerland
  - Integrating it in different national services like AAI, etc.
  - Enabling high-stake online examinations on this MOOC platform

Two SMS Subprojects

- SEB integration in Open edX:
    - Option to require and authenticate the use of Safe Exam Browser for examinations taken with Open edX
    - Similar to SEB integrations in Learning Management Systems (LMS) like ILIAS, OpenOLAT, Moodle
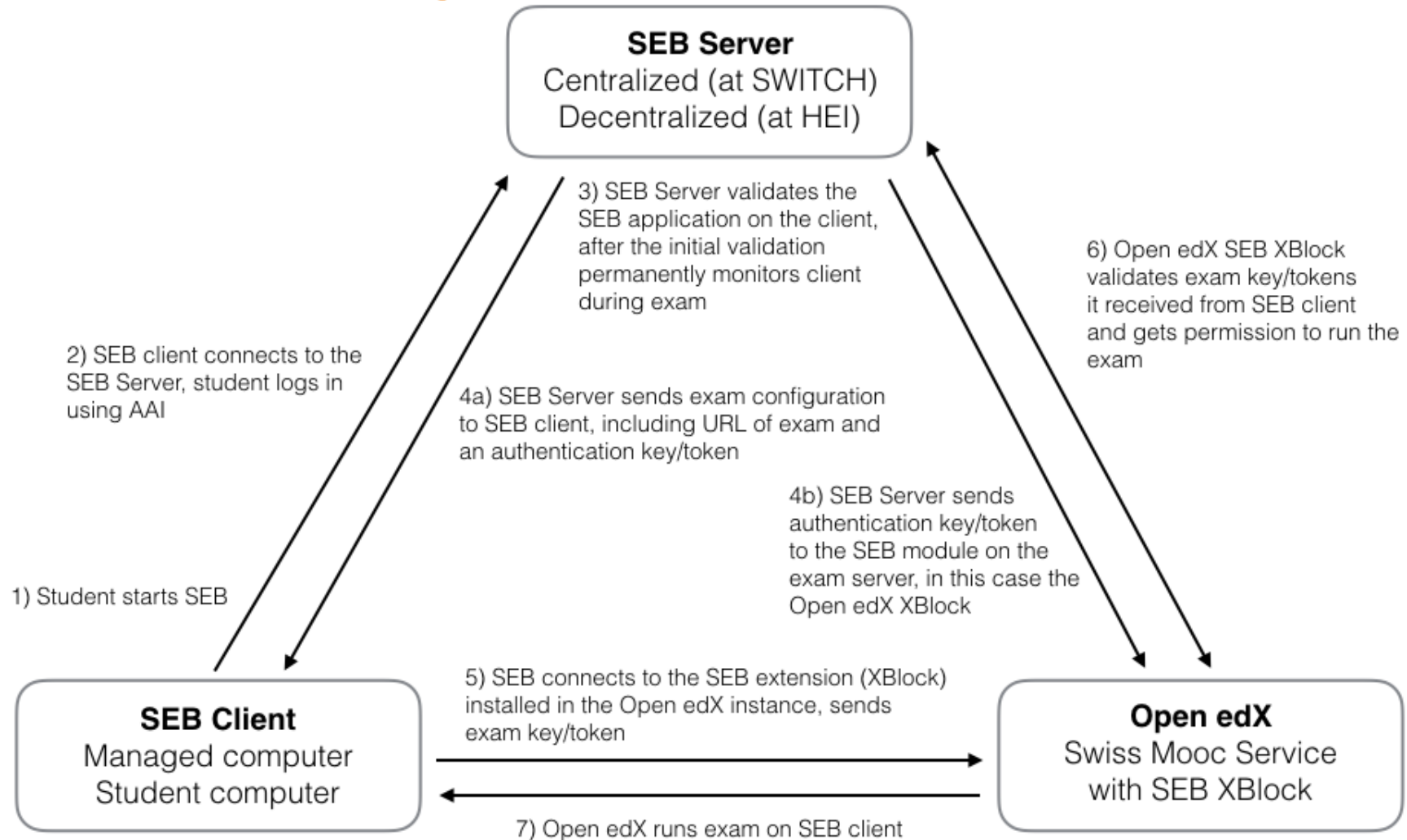
- Implementation of the SEB Server

SEB Server

- Central management of SEB clients for exams: Clients receive exam settings automatically from the SEB Server.
- Simple and user friendly interface for an exam specialist to set up the configuration for a quiz
- Improved security with automatic three-way authentication between SEB clients, exam system and SEB Server.
- Easy to operate monitoring functionality for
  - capturing incidents, SEB logs
  - optional logging of user activity
  - optional remote supervising/proctoring such as screen recording without requiring third party software solutions

Characteristics of SEB Server

Keeping the SEB exam solution modular and compatible with a wide range of exam systems

- Open source and freely available

- Well documented API to connect to exam systems

- Installation on common open source servers, Docker image

- Scalable: A centralized SEB Server for a large number of institutions will be possible

Functional Diagram

SEB Server
Centralized (at SWITCH)
Decentralized (at HEI)

3) SEB Server validates the SEB application on the client, after the initial validation permanently monitors client during exam

6) Open edX SEB XBlock validates exam key/tokens it received from SEB client and gets permission to run the exam

2) SEB client connects to the SEB Server, student logs in using AAI

4a) SEB Server sends exam configuration to SEB client, including URL of exam and an authentication key/token

4b) SEB Server sends authentication key/token to the SEB module on the exam server, in this case the Open edX XBlock

1) Student starts SEB

SEB Client
Managed computer
Student computer

5) SEB connects to the SEB extension (XBlock) installed in the Open edX instance, sends exam key/token

Open edX
Swiss Mooc Service
with SEB XBlock

7) Open edX runs exam on SEB client

Thank you for your attention.
Please discuss…

Daniel R. Schneider
Thomas Piendl

daniel.schneider@let.ethz.ch
thomas.piendl@let.ethz.ch

ETH *zürich*