MUNI
FI

Introduction to Penetration Testing Practice

PA211 Advanced Topics of Cyber Security

October 11, 2022

Ádám Ruman, Ivan Kotora

Agenda

- Response to the Exit Ticket from Last Week
- Introduction to This Part of the Course
- Offensive Security
- Introduction to Penetration Testing
- Penetration Testing Process – Overview
- Penetration Testing Process – Preparation
- Penetration Testing Process – Testing

Exit tickets from last week – I

- **Q:** From where does the auditbeat get the data?
- **A:** From **Linux audit framework**. It obtains **auditd** data.

- **Q:** After detecting something strange, what would be the next part?
- **A:** ***Response.*** *For example,* ***validation*** *of true positives,* ***investigation*** *of possible consequences, and* ***mitigation*** *of cyber threats.*

- **Q:** Why did we set MITRE ATT&CK Credential Access tactic in our rule?
- **A:** *The attacker could be trying to* ***obtain SSH password****. The tactic represents a* ***tag****.*

Exit tickets from last week – II

- **Q:** I am not really sure if kibana was the web app we used or whether it was elastic instead, it seemed as if those two names were used interchangeably.
- **A:** *We did **not** use "Elastic". We used **Kibana UI** and one of its sections containing **Elastic Security app**. See also [1].*

***Elastic** – name of company.*
***Kibana (UI)** – visualization and navigation of Elastic Stack.*
***Elastic Security** – SIEM (Security information and event management).*
***Elasticsearch** – a set of tools for data ingestion, storage, and analysis.*
***Logstash** – a pipeline collecting data from data sources.*
***Beat –** single-purpose data shipper.*

Week	Date	Class Topic
1	13.09.2022	Course organization and motivation
2	20.09.2022	Asset management
3	27.09.2022	Vulnerability management
4	04.10.2022	Threat management
5	11.10.2022	Penetration testing – introduction
6	18.10.2022	Penetration testing – process
7	25.10.2022	Penetration testing – report
8	01.11.2022	Penetration testing – exemplary report and presentations
9	08.11.2022	Introduction to web application hardening
10	15.11.2022	OS-level, virtualization and containerization
11	22.11.2022	Access control mechanisms
12	29.11.2022	Web server and application hardening
13	06.12.2022	Course feedback session

Part II – Penetration testing practice

- **Syllabus:** Process, report, and presentation
- **Objectives:**
  - **Understand the process** of authorized penetration testing
  - Focus **on the process**, not individual tools
- **Learning outcomes:**
  - **Hands-on experience** with penetration **testing** **of a realistic application** **in a team**
  - Knowledge of the **structure of a testing report**
  - Exercising skills for **preparing reports and presentation**
- **Assessment: 1 homework – report and presentation**
  - Deadline: November 1, 2022

Agenda

- Introduction to This Part of the Course
- Offensive Security
- Introduction to Penetration Testing
- Penetration Testing Process – Overview
- Penetration Testing Process – Preparation
- Penetration Testing Process – Testing

Why Offensive Security (OffSec)?

Blue teams implement **security measures** on a product or system.

However, someone has to test/assess the **adequacy and quality** of these measures.

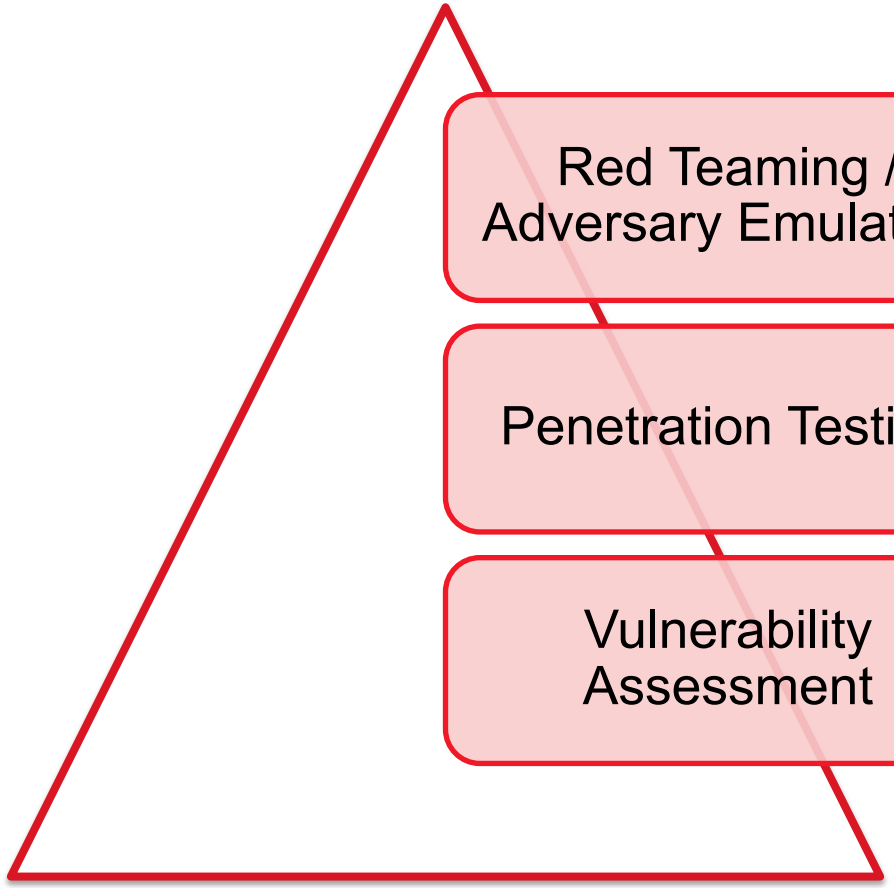
Audits are *cool 'n all*, **BUT** they solve problems on a different plane (design, documentation).

Leaving the testing to an actual attacker is a **bad idea**.

Thus, **red** teams specialize in offensive security. Their goal is to test security measures by trying to breach them – usually without destructive intentions.

MUNI FI

The Offensive Security Pyramid



- Target is the entire system with *users and defenders* included.
- Almost no limitations for red team operators.
- Includes vulnerability research and exploit development.
- Goal is to assess *processes*, detections, and responses to incidents.

- Target is either a part of a system or an application.
- Rather substantial limitations for penetration testers.
- Find well-known vulnerabilities and try exploiting them.
- Goal is to test the *technical implementation*.

- Target is either a part of a system or an application.
- Strong limits for the assessing team.
- Find all well-known vulnerabilities and assess if they are True Positives.
- Easy to automate → *scalable*.

Great blog articles can be found here, here and here (also motivation for the pyramid).

MUNI FI

Agenda

- Introduction to This Part of the Course
- Offensive Security
- Introduction to Penetration Testing
- Penetration Testing Process – Overview
- Penetration Testing Process – Preparation
- Penetration Testing Process – Testing

Definition(s)

- *“A method for gaining assurance in the security of an IT system by attempting to breach some or all of that system's security, using the same tools and techniques as an adversary might.”* – NCSC
- Multiple definitions in the NIST glossary.

Penetration Testing Primer – I

- **Why?** – To find security holes in a system before an adversary.
  - Assess the level of their criticality and associated risk.
  - Report them.
  - Provide remediation ideas where possible.

Penetration Testing Primer – II

- **Who?** – Internal or external penetration testers.

- As a CISO, which one would you prefer using?
  - I – Internal team.
  - II – External team.

Penetration Testing Primer – III

- **How often?** – Depends on the target.

- What do you think is the best practice for companies?
  - I – When we need to spend some money.
  - II – Periodically, driven by product development.
  - X – Periodically, strict time intervals.

The Duality of Penetration Testing

- The technical plane:
  - You are **testing implementation**.
  - You report to and offer remediation techniques to developers.

- The social plane:
  - Your main customer is the **executive board**/management of a company, and they are interested in the results as well.
  - For successful testing, synergy and good communication are needed between the tester and the customer.

Required Skills for Penetration Testers

- Well-versed in CVEs, and CWEs.
- Some experience with exploitation.
- **Knowledge about standard technologies:**
  - You need to know what you might disrupt/damage if not careful enough.
- Soft skills.
- Good to have an overview of laws regarding privacy.
- Hardcore programming and SWING skills are not needed, but it's good to be able to write scripts and understand code.

Penetration Testing Taxonomy

Based on testing team	• Internal • External
Based on target	• Web. • Infrastructure. • Mobile. • Physical, etc.
Based on access	• Black-Box – no prior information about the target. • Grey-Box – some information about the target (design, topology). • White-Box – full information about the target (source code, technical details).

Penetration Testing in Product Development

Provide Training
Ensure everyone understands security best practices.
Learn more >

Define Security Requirements
Continually update security requirements to reflect changes in functionality and to the regulatory and threat landscape.
Learn more >

Define Metrics and Compliance Reporting
Identify the minimum acceptable levels of security quality and how engineering teams will be held accountable.
Learn more >

Perform Threat Modeling
Use threat modeling to identify security vulnerabilities, determine risk, and identify mitigations.
Learn more >

Establish Design Requirements
Define standard security features that all engineers should use.
Learn more >

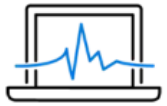
Define and Use Cryptography Standards
Ensure the right cryptographic solutions are used to protect data.
Learn more >

Manage the Security Risk of Using Third-Party Components
Keep an inventory of third-party components and create a plan to evaluate reported vulnerabilities.
Learn more >

Use Approved Tools
Define and publish a list of approved tools and their associated security checks.
Learn more >

Perform Static Analysis Security Testing (SAST)
Analyze source code before compiling to validate the use of secure coding policies.
Learn more >

Perform Dynamic Analysis Security Testing (DAST)
Perform run-time verification of fully compiled software to test security of fully integrated and running code.
Learn more >



Perform Penetration Testing
Uncover potential vulnerabilities resulting from coding errors, system configuration faults, or other operational deployment weaknesses.
Learn more >

Establish a Standard Incident Response Process
Prepare an Incident Response Plan to address new threats that can emerge over time.
Learn more >

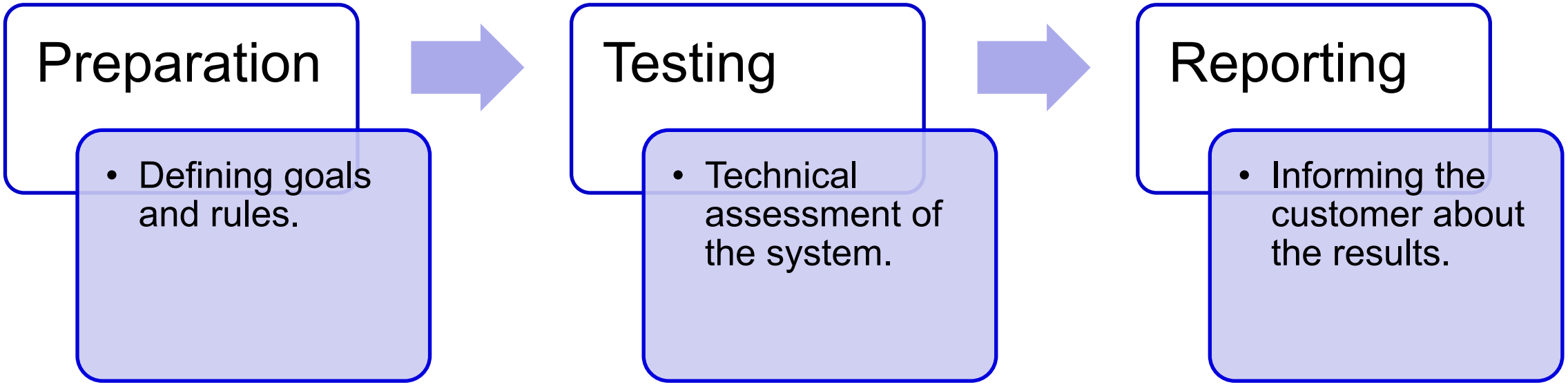
Source: [MS-SDLC.]

MUNI FI

Agenda

- Introduction to This Part of the Course
- Offensive Security
- Introduction to Penetration Testing
- Penetration Testing Process – Overview
- Penetration Testing Process – Preparation
- Penetration Testing Process – Testing

Penetration Test Life-Cycle



MUNI FI

Agenda

- Introduction to This Part of the Course
- Offensive Security
- Introduction to Penetration Testing
- Penetration Testing Process – Overview
- Penetration Testing Process – Preparation
- Penetration Testing Process – Testing

Test Preparation

Settle the goals.

Define the scope.

Agree on engagement rules.

Make testing plans.

Defining Goals

- *Prioritization*
  - Which system parts are most **valuable** for the company?
  - Does the system process or store **sensitive** data?
- *Filtering out client requirements:*
  - You are **not able to meet** (unqualified).
  - Are not worth the resources.
  - Are impractical or **unrealistic**.
- A lot of communication is involved.
  - You want to **help** your client, not make them feel dumb.

The Scope

- Describes the **precise** target of the test.
  - Systems, applications, or their components.
- *De facto* the "**holy writing**" for penetration testing.
- Violating the scope is the fastest way to lose contracts and credibility.

Rules of Engagement

- Time constraints.
- Should we do social engineering?
- Special requirements – mainly if the test is done in production.
  - You **can not disrupt** the functionality of assets.
  - How to handle sensitive data.
- What if something gets out of hand accidentally?
  - Reporting rules, recovery plans, and responsibility sharing.
- Sometimes testers learn valuable (for a competitor) information – solved by a **Non-Disclosure Agreement** (NDA).

Planning

- Receive documentation from the client.
  - About the system or devices.
  - For efficiency and safety.
  - Contacts.
- Choose a **methodology**. (more later)
- Create a test schedule – for synchronization and special timings.

Agenda

- Introduction to This Part of the Course
- Offensive Security
- Introduction to Penetration Testing
- Penetration Testing Process – Overview
- Penetration Testing Process – Preparation
- Penetration Testing Process – Testing

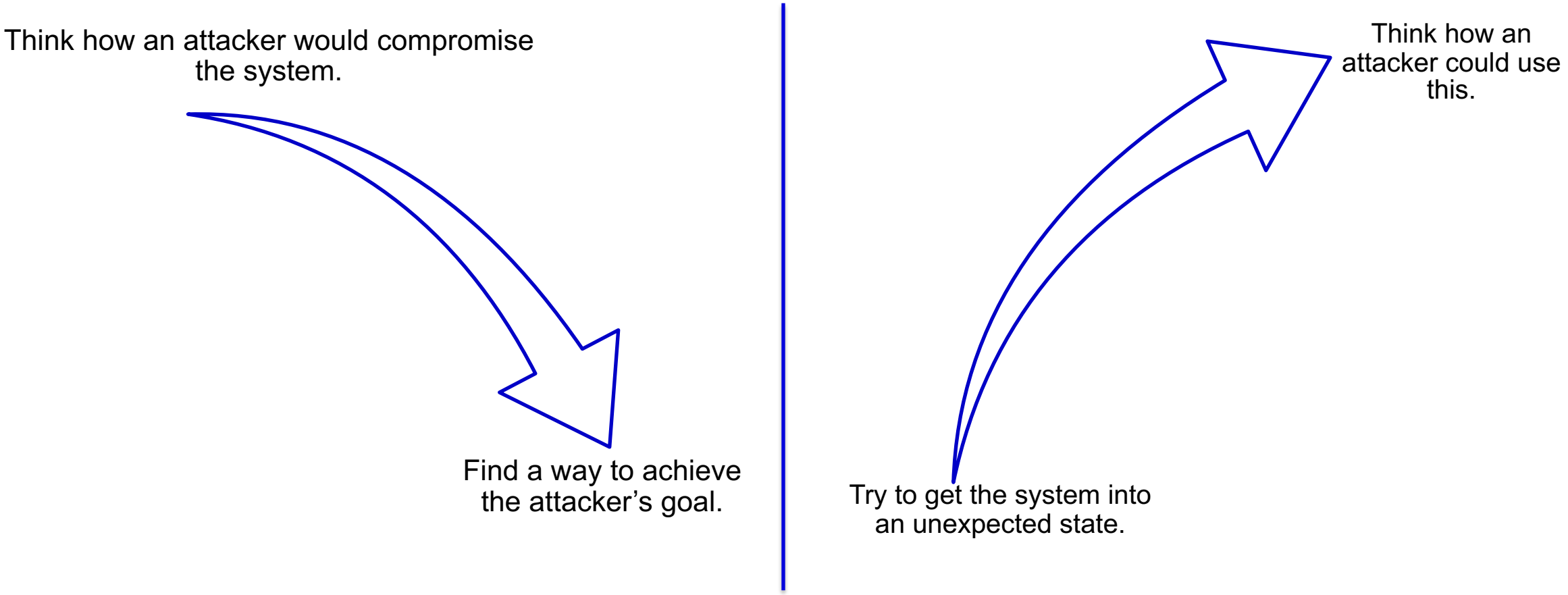
The Test

MUNI FI

Reconnaissance

- Exploring the system as a user.
  - Find discrepancies with the documentation.
- Scanning the target.
- OSINT (Open-Source INTelligence)

Threat Modeling & Vulnerability Identification



MUNI FI

Exploitation

- We found vulnerabilities and we have an idea how to misuse them.
- Create a **proof-of-concept exploit** (PoC).
- Should you use it right away?

Exploitation

- We found vulnerabilities and we have an idea how to misuse them.
- Create a PoC exploit.
- Should you use it right away? No.
- Exploits might be **damaging** – consider all scenarios before action.

Post Exploitation

- We used an exploit (it was not disruptive) and got **new capabilities** (knowledge or access).
- We could probably do harm, but that's not the goal.
- We use our new capabilities to **dig deeper** – iteratively.

General “two cents”

- If you break something, **don’t delay reporting** it to the client.
- Take **notes** of anything that might prove valuable.
- Screenshots are a great help, but be careful with info leakage.
- If possible, use a **throwaway environment** for the tests – to prevent leakage.

Collaborative Tools

- **Sharing information** between penetration testers is crucial.
- Some tools (CoreImpact, CobaltStrike) have **built-in collaboration options** or at least result exporting.
- PwnDoc – mostly for collaborative report writing.
- PenteRep.

Penetration Testing Methodologies

- Guidelines for different scenarios.
- OWASP Web Security Testing Guide
- Open-Source Security Testing Methodology Manual
- Penetration Testing Execution Standard
- NIST 800-115

Agenda

- Introduction to This Part of the Course
- Offensive Security
- Introduction to Penetration Testing
- Penetration Testing Process – Overview
- Penetration Testing Process – Preparation
- Penetration Testing Process – Testing

Reporting will be covered in two weeks.

MUNI FI

Optional Reading Materials

- Metasploit Unleashed.
- Vx-Underground.
- All the links in the slides.

Team formation

- In this part of the course, you will work in **a team of three.**
- This settings simulates a real work role and workplace.
- The team organization is up to you.
- You will submit your **homework** (report and presentation) **as a team**. All team members will receive the same number of points.
- Now meet your colleagues a sit together.

Teams

- Team **1**: Kobyda, **Simon**; Caby, **Jules**; Urban, **Michal**
- Team **2**: Biloš, **Tomáš**; Smejkal, **Jan**; Kleman, **Matej**
- Team **3**: Filo, **Denis**; Štěpán, **Daniel**; Mercell, **Peter**
- Team **4**: Fouček, **Šimon**; Šoška, **Marek**; Mann, **Radomír**
- Team **5**: Saloň, **Benjamin**; Ambros, **Samuel**; Mika, **Kristián**
- Team **6**: Saloň, **Juraj Samuel**; Fischer, **Glenn**; Rýpar, **David**

Change your seats.

Boot your own machines.

Teams

- Team **1**: Kobyda, **Simon**; Caby, **Jules**; Urban, **Michal**
- Team **2**: Biloš, **Tomáš**; Smejkal, **Jan**; Kleman, **Matej**
- Team **3**: Filo, **Denis**; Štěpán, **Daniel**; Mercell, **Peter**
- Team **4**: Fouček, **Šimon**; Šoška, **Marek**; Mann, **Radomír**
- Team **5**: Saloň, **Benjamin**; Ambros, **Samuel**; Mika, **Kristián**
- Team **6**: Saloň, **Juraj Samuel**; Fischer, **Glenn**; Rýpar, **David**

MUNI

FI