Git-based CTF:
A Simple and Effective Approach to Organizing In-Course Attack-and-Defense Security Competition

Seongil Wi, Jaeseung Choi, Sang Kil Cha

KAIST

Capture The Flag (CTF)

CTF: Cybersecurity competition that involves capturing a flag





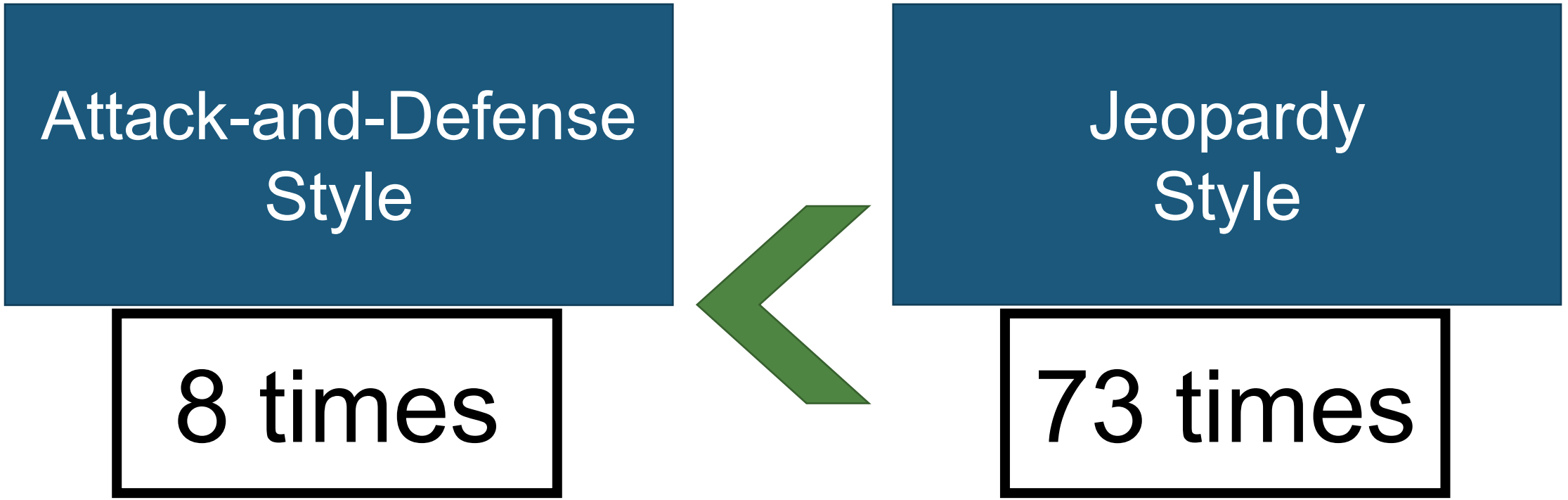
CSAW



Types of CTF

Attack-and-Defense Style	Jeopardy Style
Real time, Realistic	One-way, Problem solving

Number of CTF Events in 2018



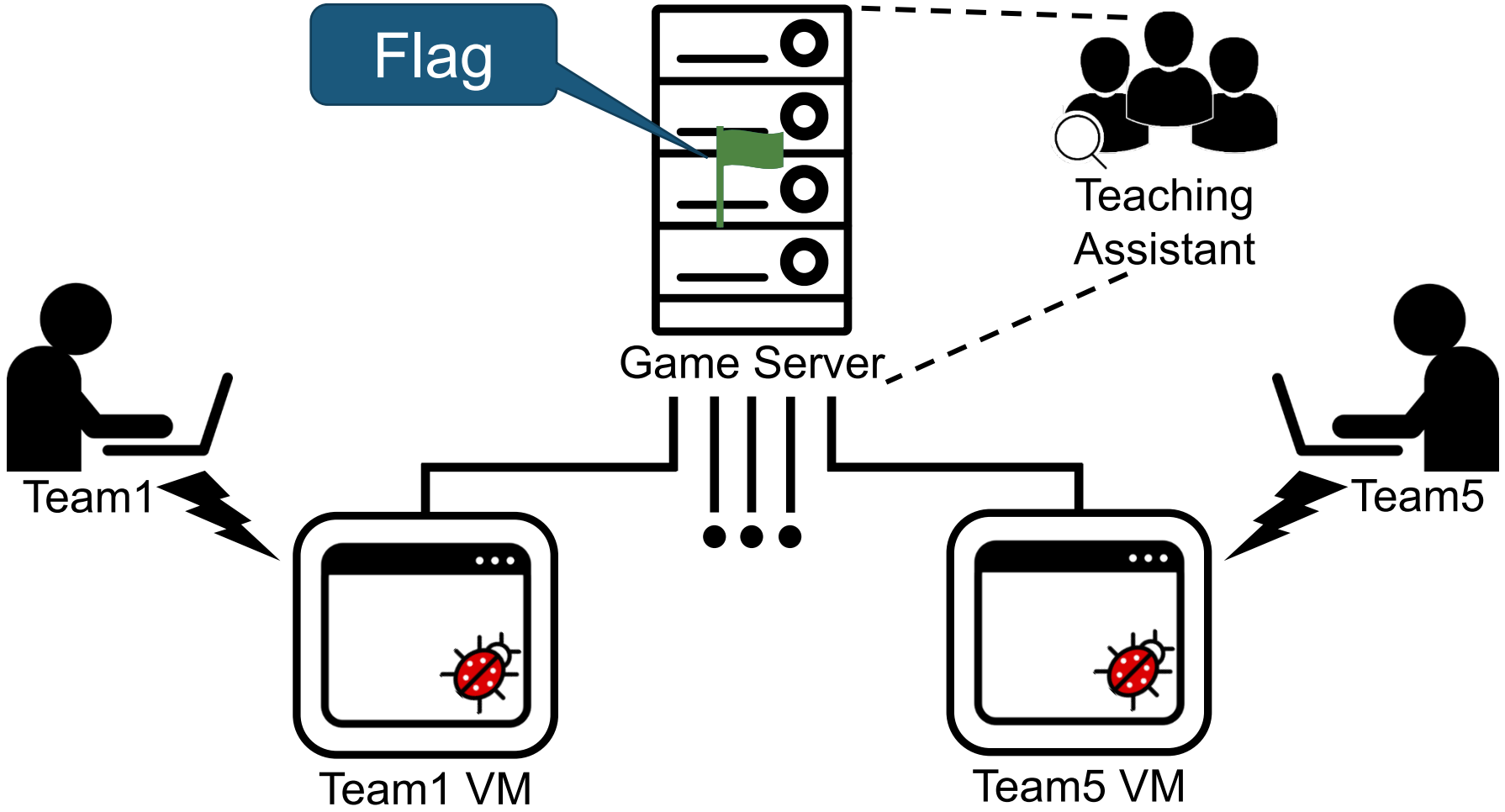
In-Course Attack and Defense CTF

- Class Capture-the-Flag Exercises,
  USENIX 3GSE '14

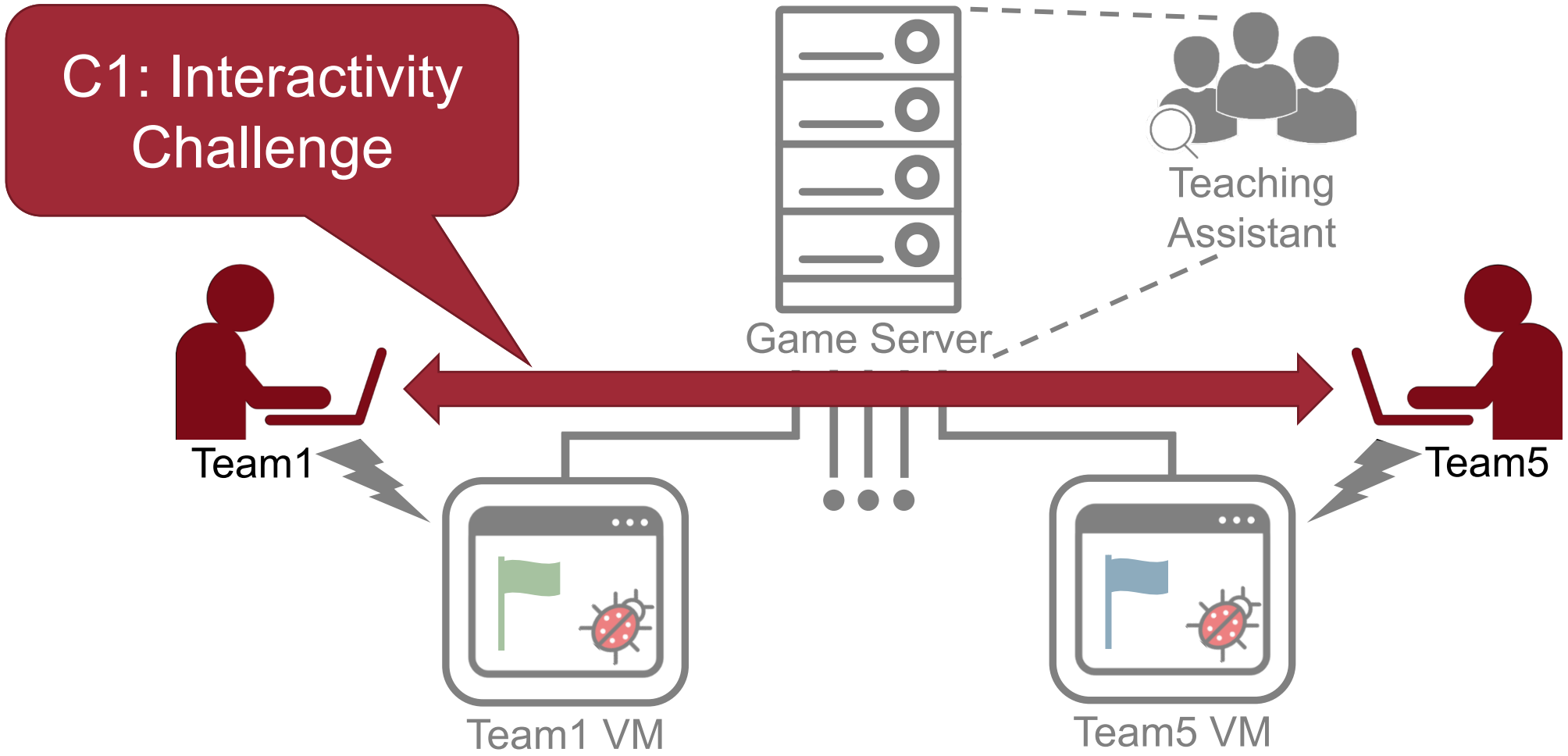
- Build It, Break It, Fix It: Contesting Secure Development,
  ACM CCS '16

Difficult to organize!

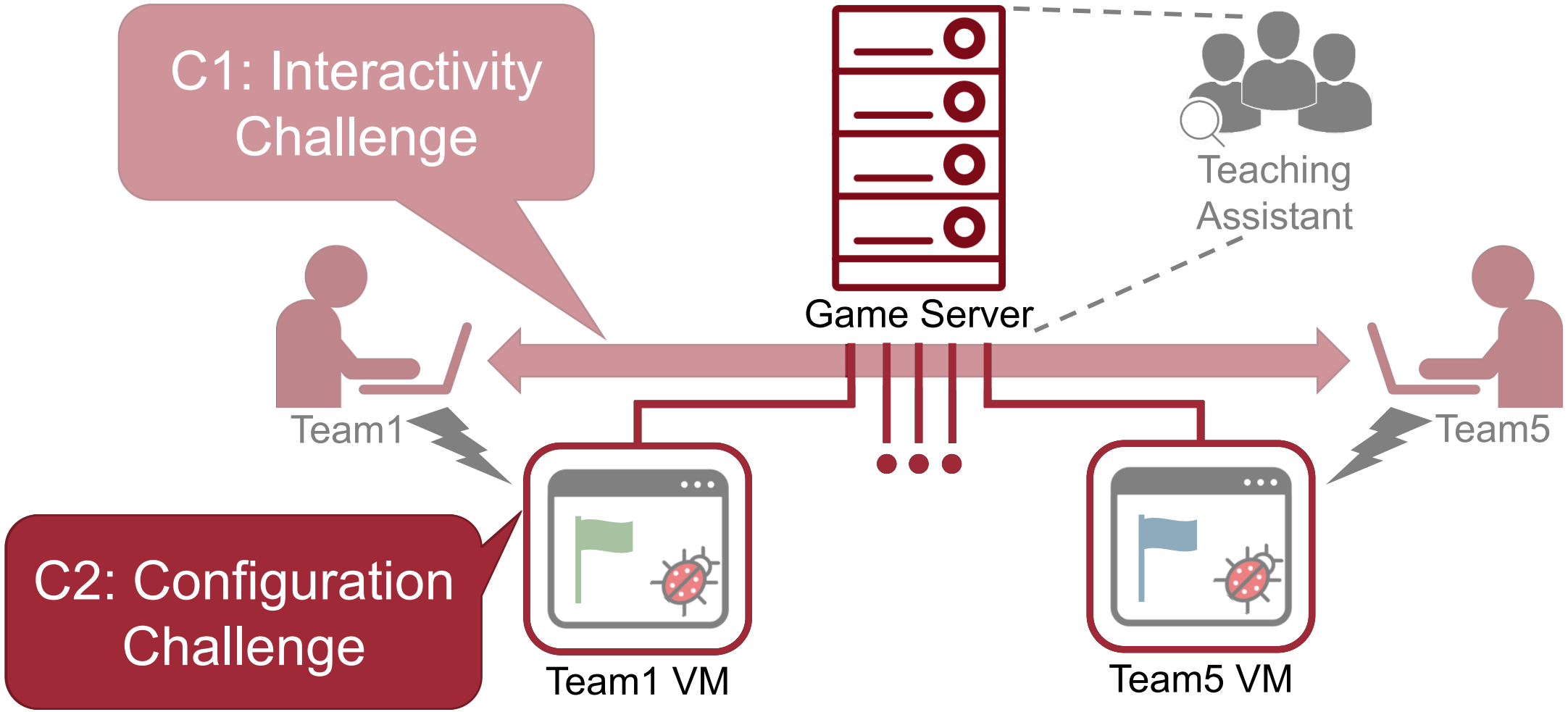
Attack & Defense CTF Infrastructures



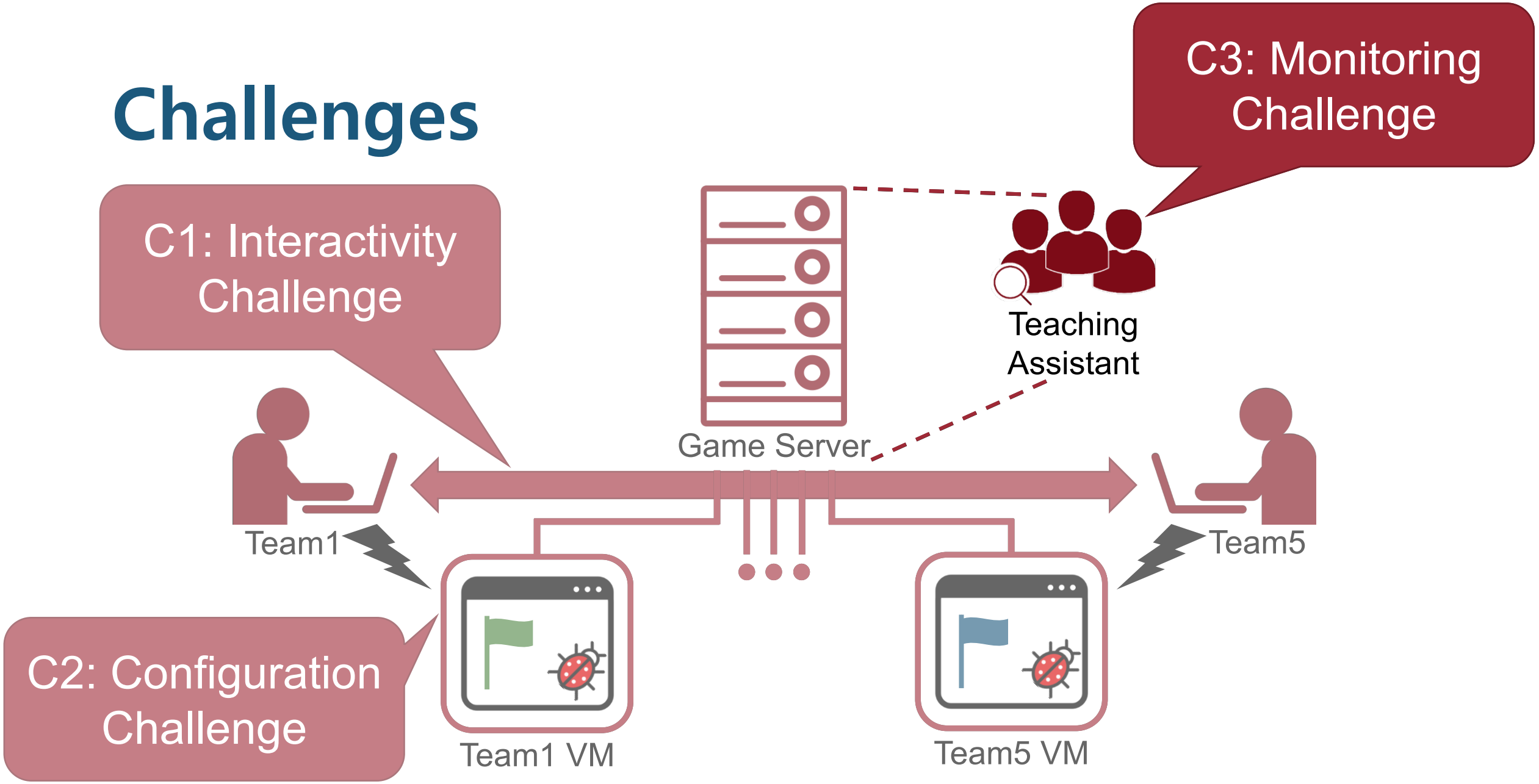
Challenges


C1: Interactivity Challenge
Game Server
Teaching Assistant
Team1
Team5
Team1 VM
Team5 VM

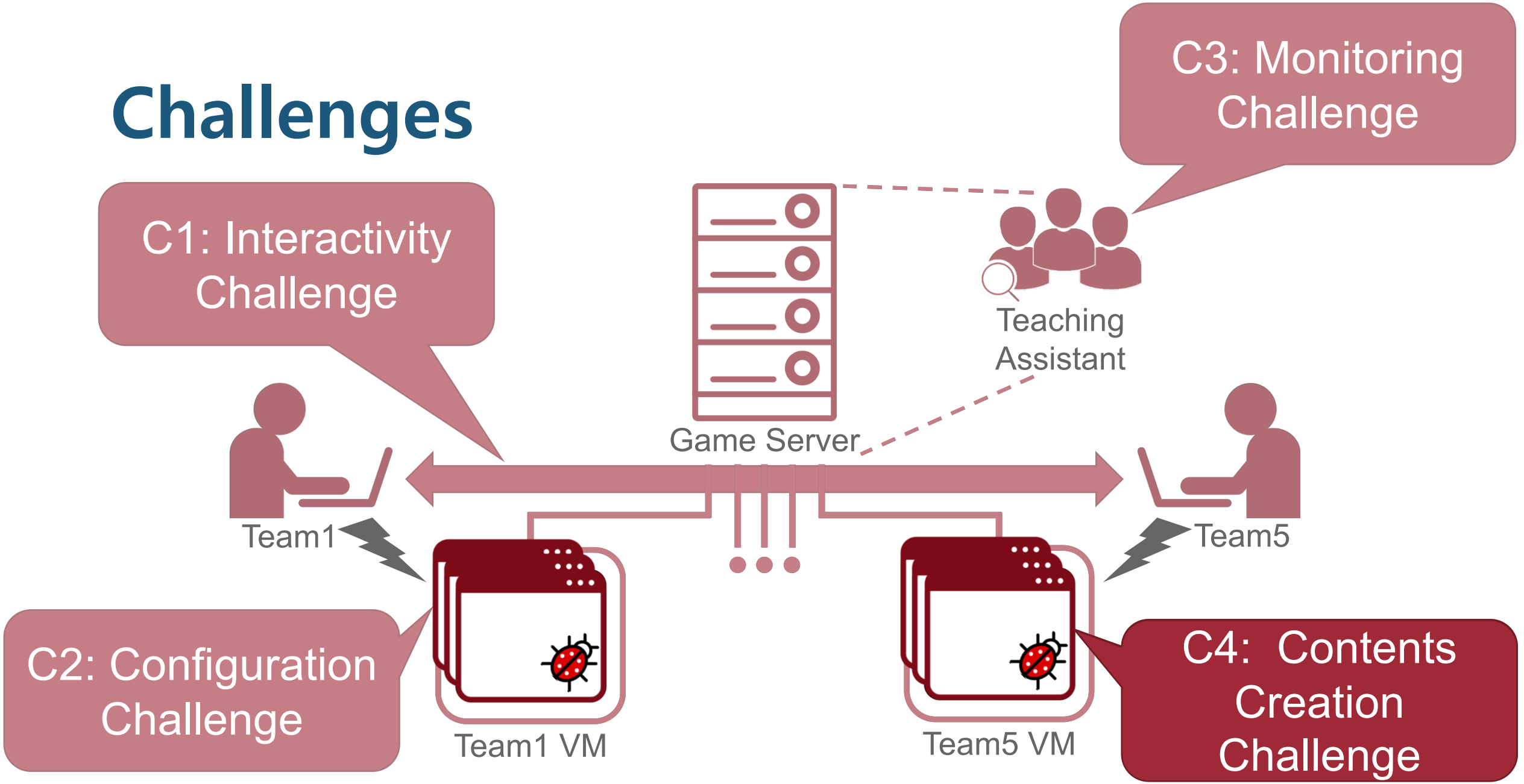
Challenges


C1: Interactivity Challenge
Game Server
Teaching Assistant
Team1
Team5
C2: Configuration Challenge
Team1 VM
Team5 VM

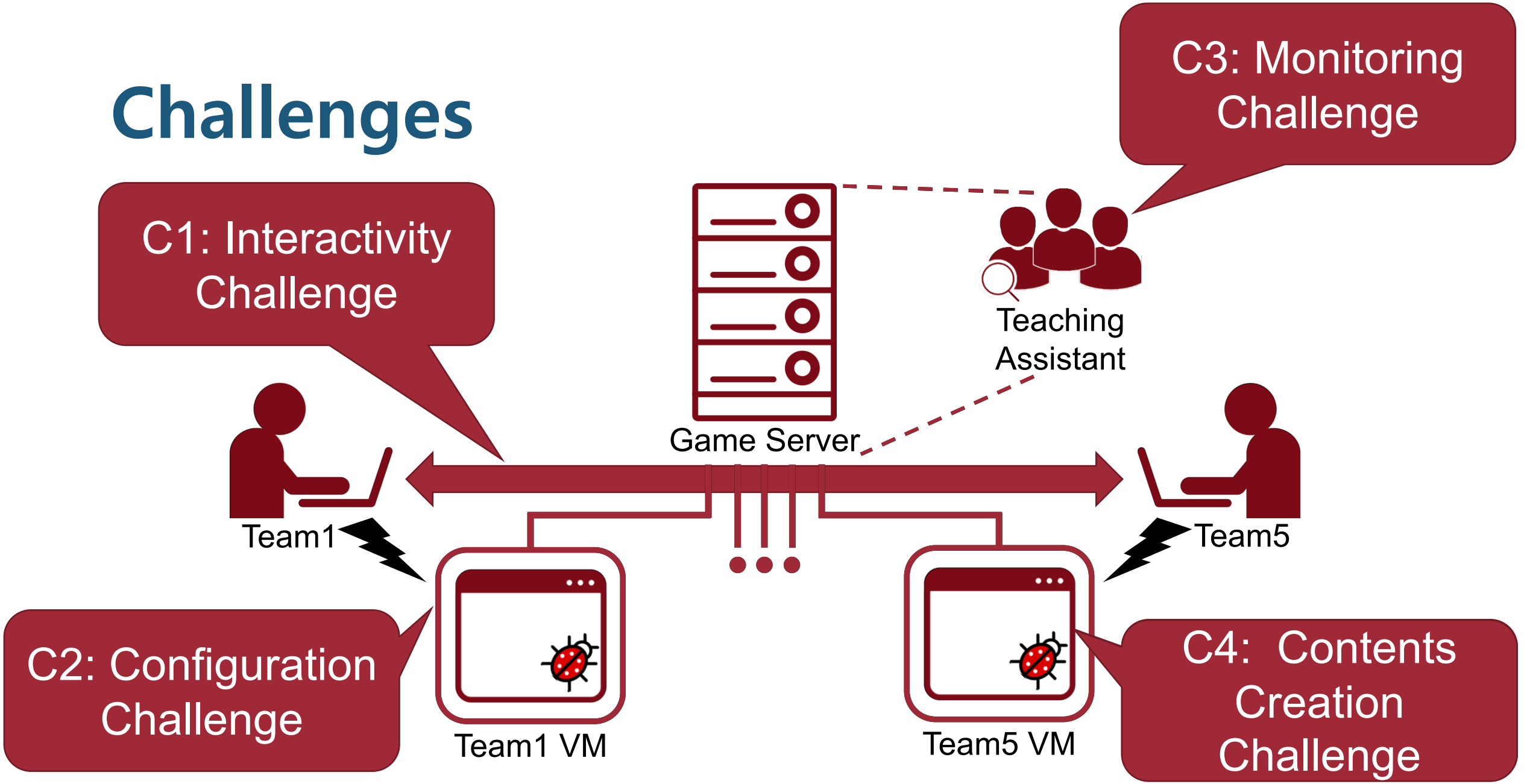
Challenges


C3: Monitoring Challenge
C1: Interactivity Challenge
Teaching Assistant
Game Server
Team1
Team5
C2: Configuration Challenge
Team1 VM
Team5 VM

Challenges


C3: Monitoring Challenge
C1: Interactivity Challenge
Teaching Assistant
Game Server
Team1
Team5
C2: Configuration Challenge
Team1 VM
Team5 VM
C4: Contents Creation Challenge

Challenges

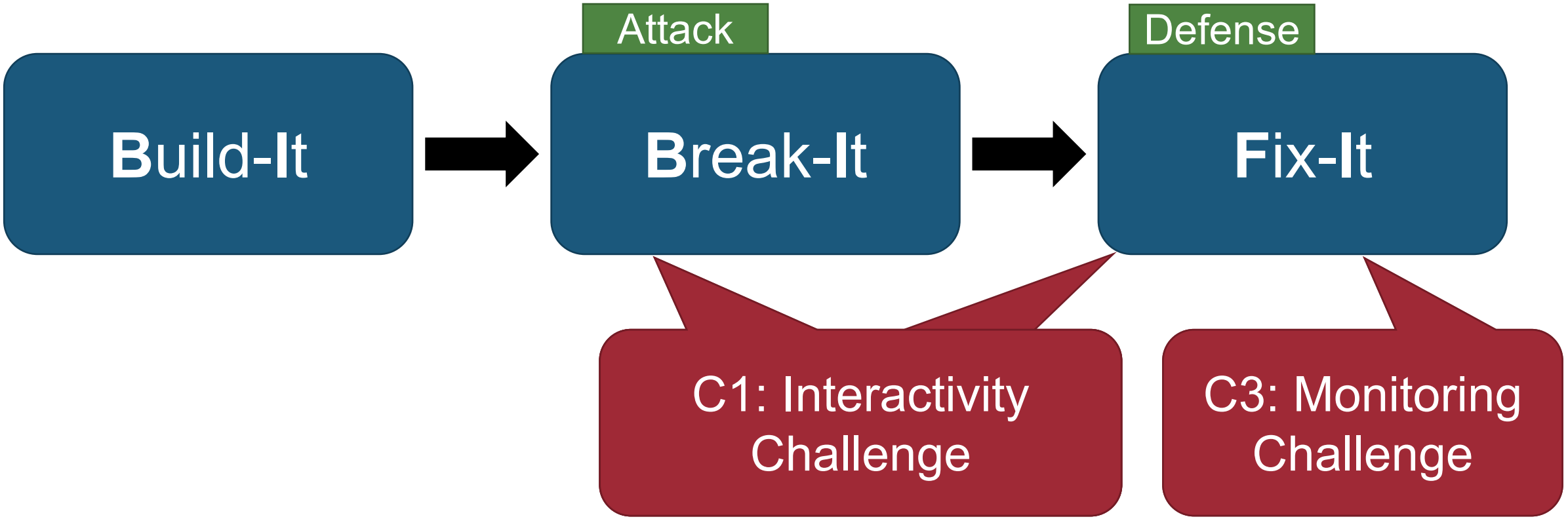

C3: Monitoring Challenge
C1: Interactivity Challenge
Teaching Assistant
Game Server
Team1
Team5
C2: Configuration Challenge
Team1 VM
Team5 VM
C4: Contents Creation Challenge

Recent Researches:

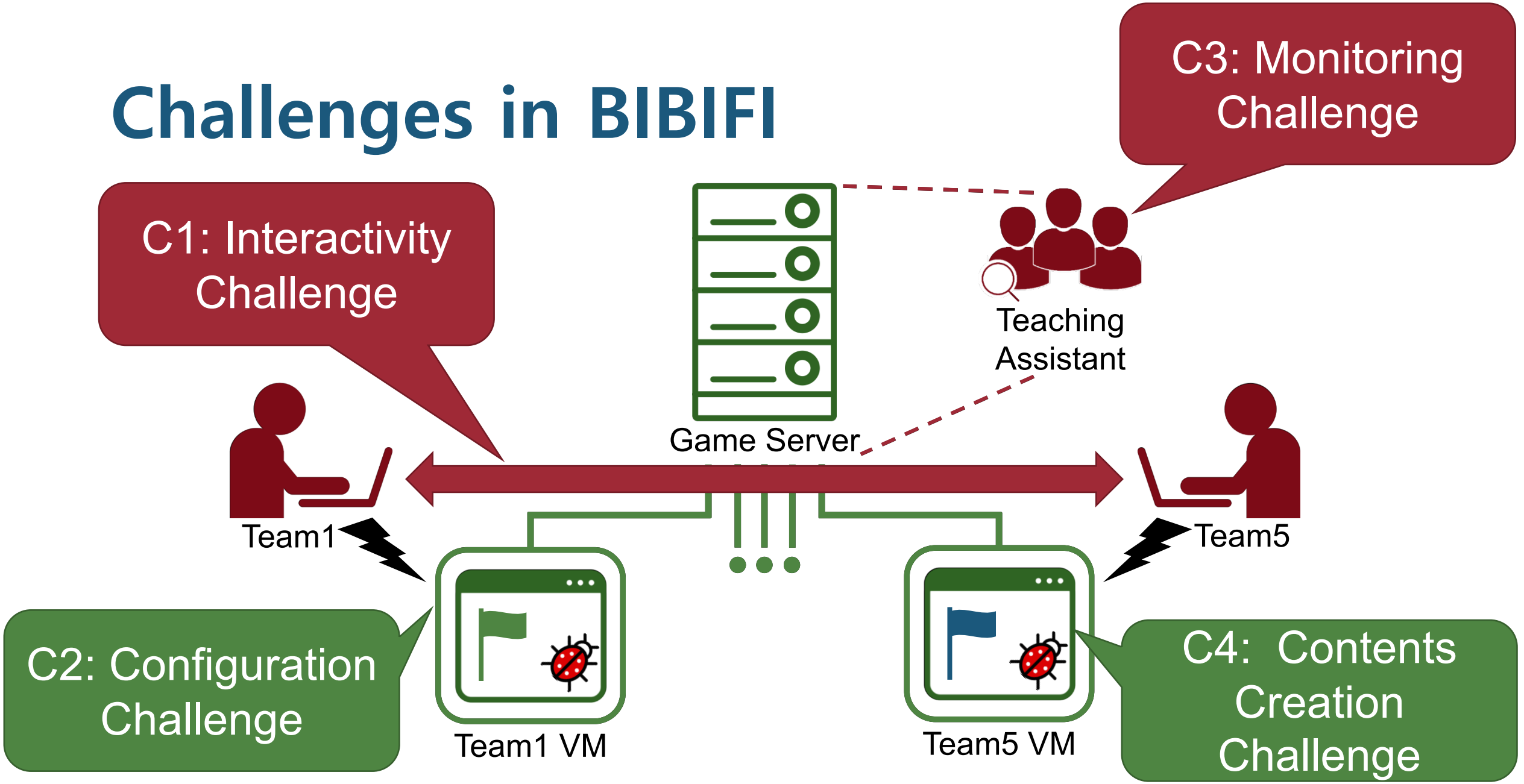
- C1: Interactivity Challenge
  - SWPAC, ***USENIX ASE '17***
- C3: Monitoring Challenge
  - VM-based Framework, ***USENIX 3GSE '15***
- C4: Contents Creation Challenge
  - BIBIFI, ***ACM CCS '16***
  - SecGen, ***USENIX ASE '17***

Handle only a subset of the challenges

Previous Work: BIBIFI, *ACM CCS '16*



Challenges in BIBIFI

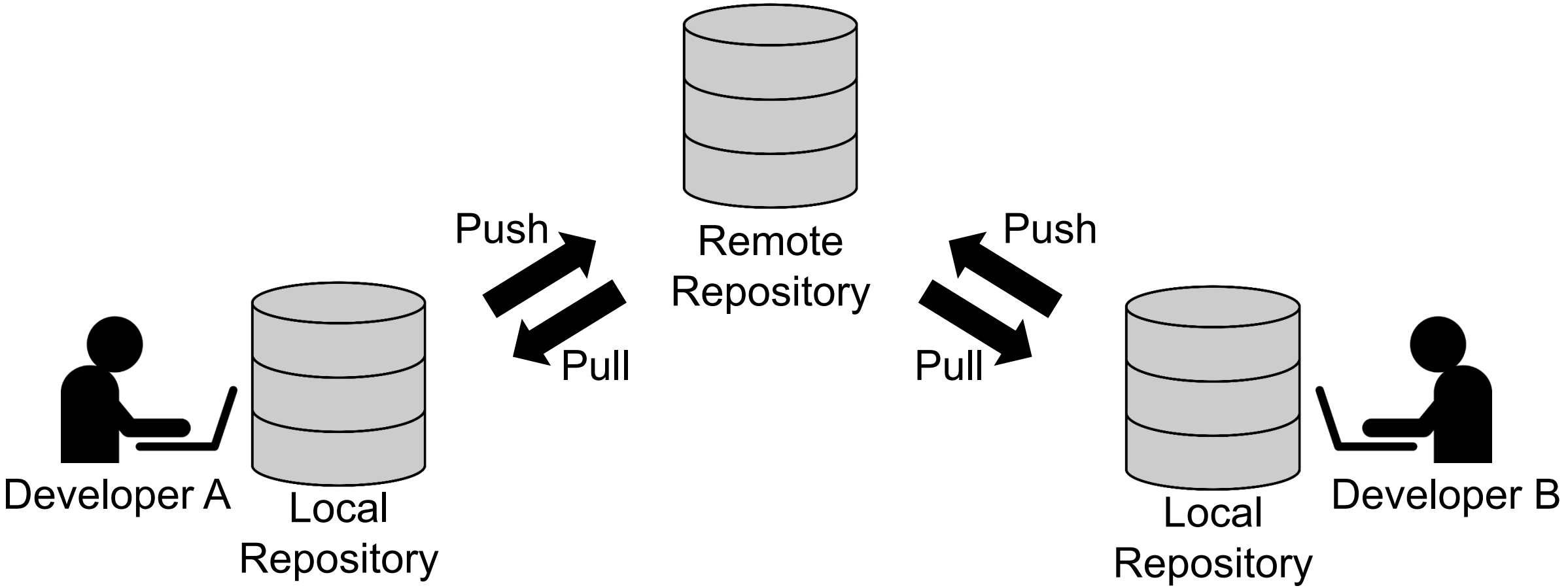


Can we handle all the challenges?

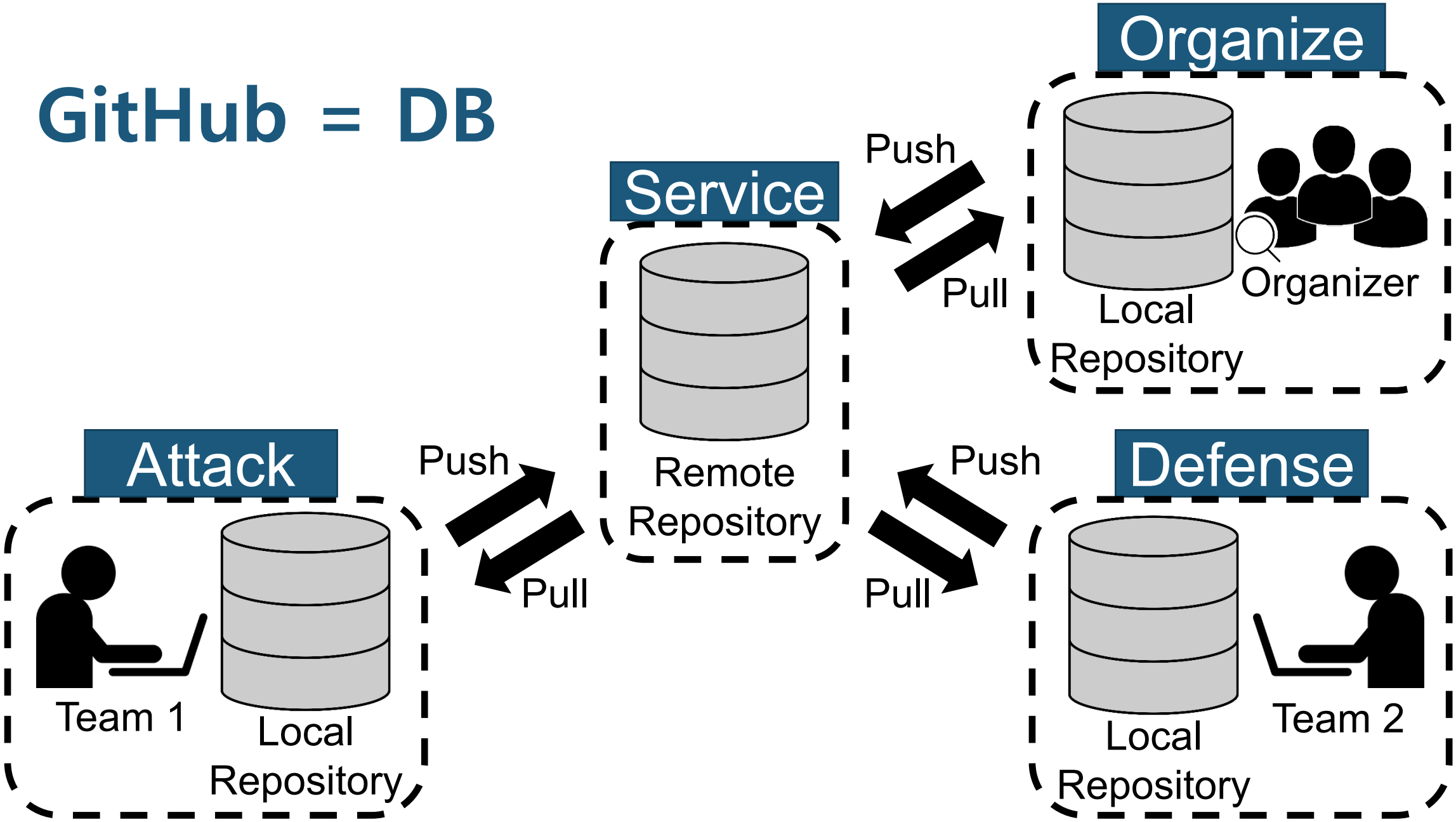
We propose

Git-based CTF

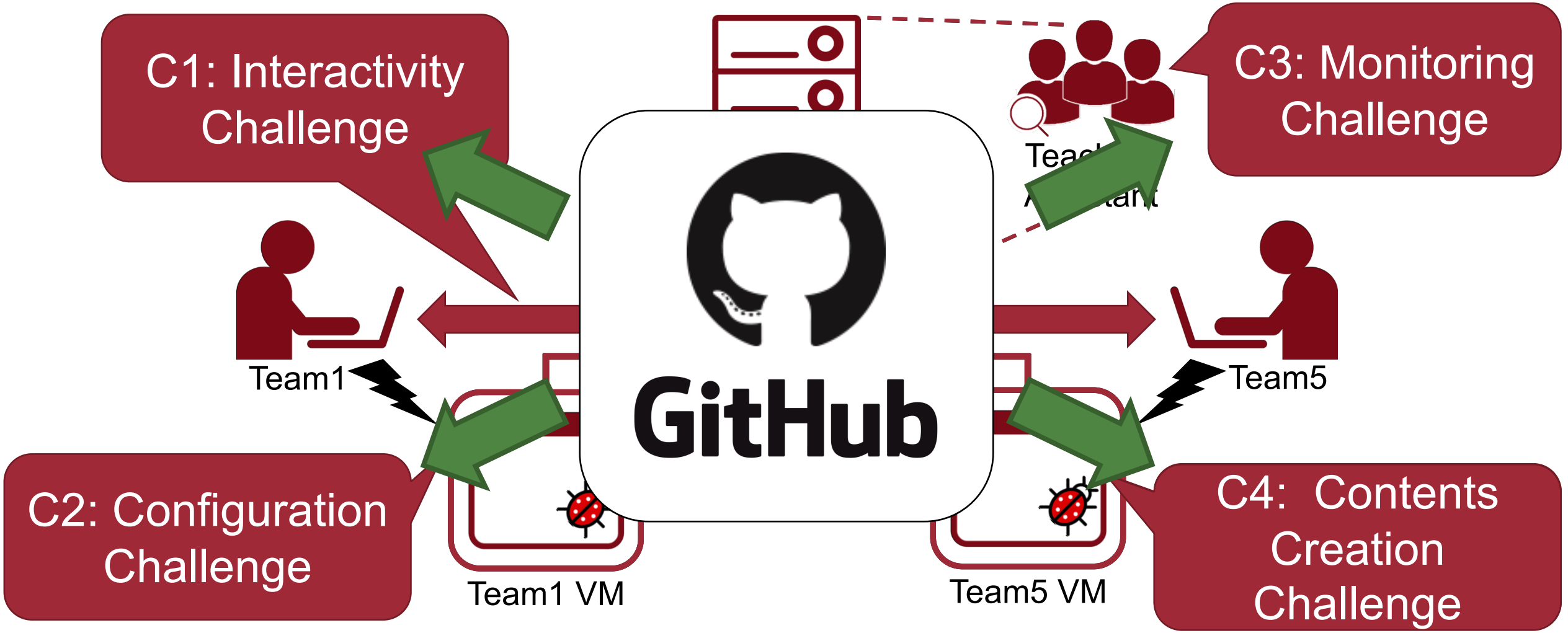
GitHub as a CTF Framework



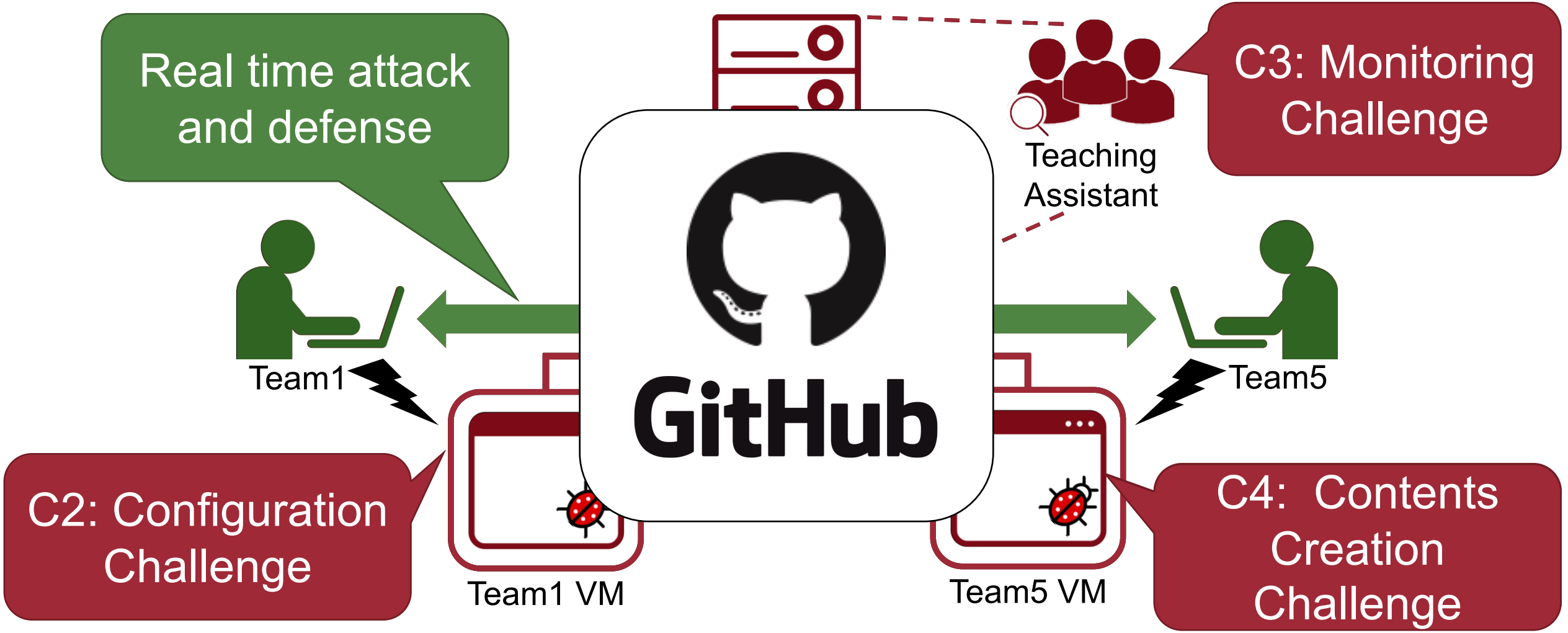
GitHub = DB


Organize
Push
Service
Pull
Organizer
Local
Repository
Remote
Repository
Attack
Push
Push
Defense
Pull
Pull
Team 1
Local
Repository
Team 2
Local
Repository

Our Goal: Handle All the Challenges



Handle Interactivity Challenge

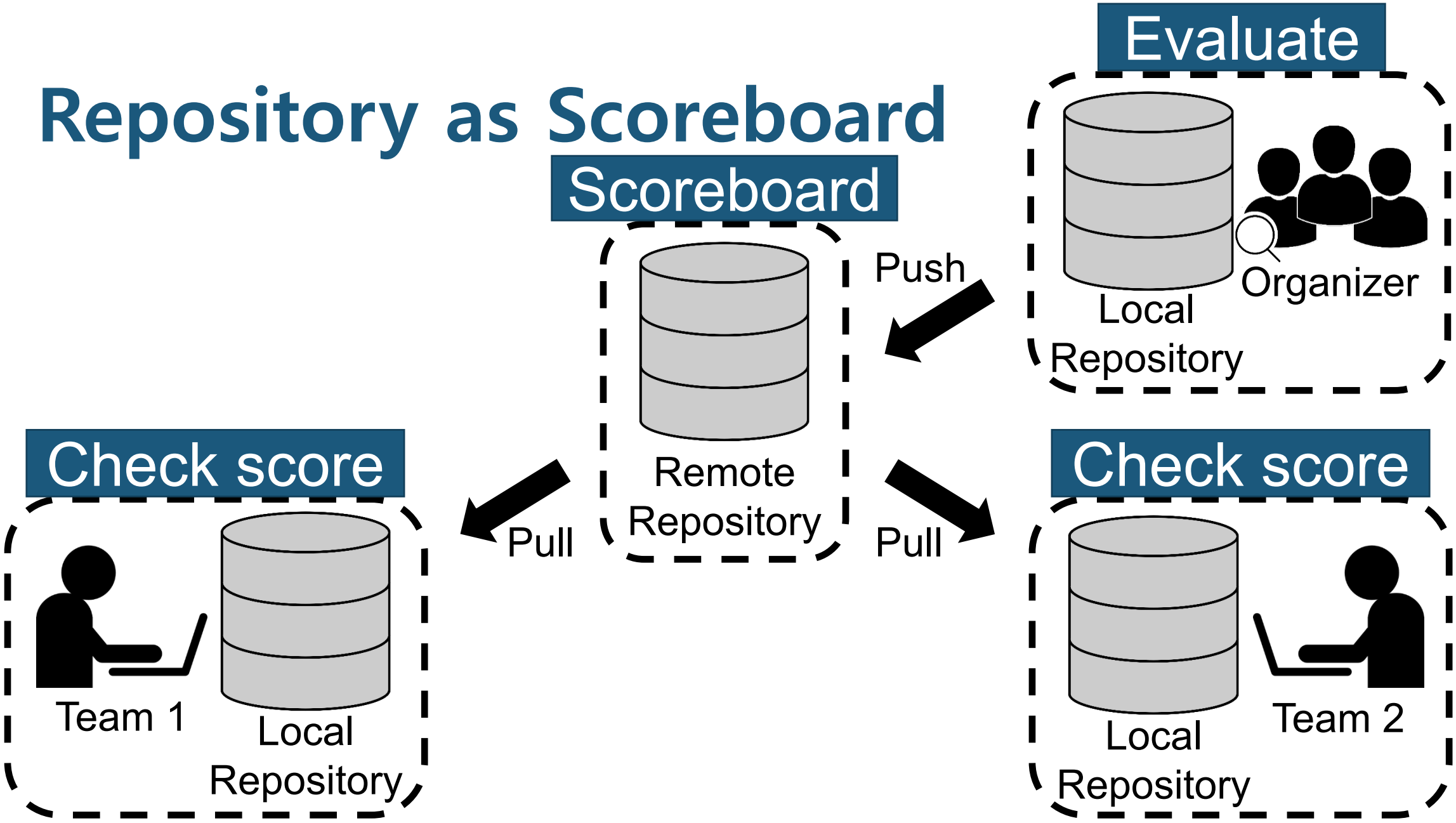


Real Time Attack and Defense

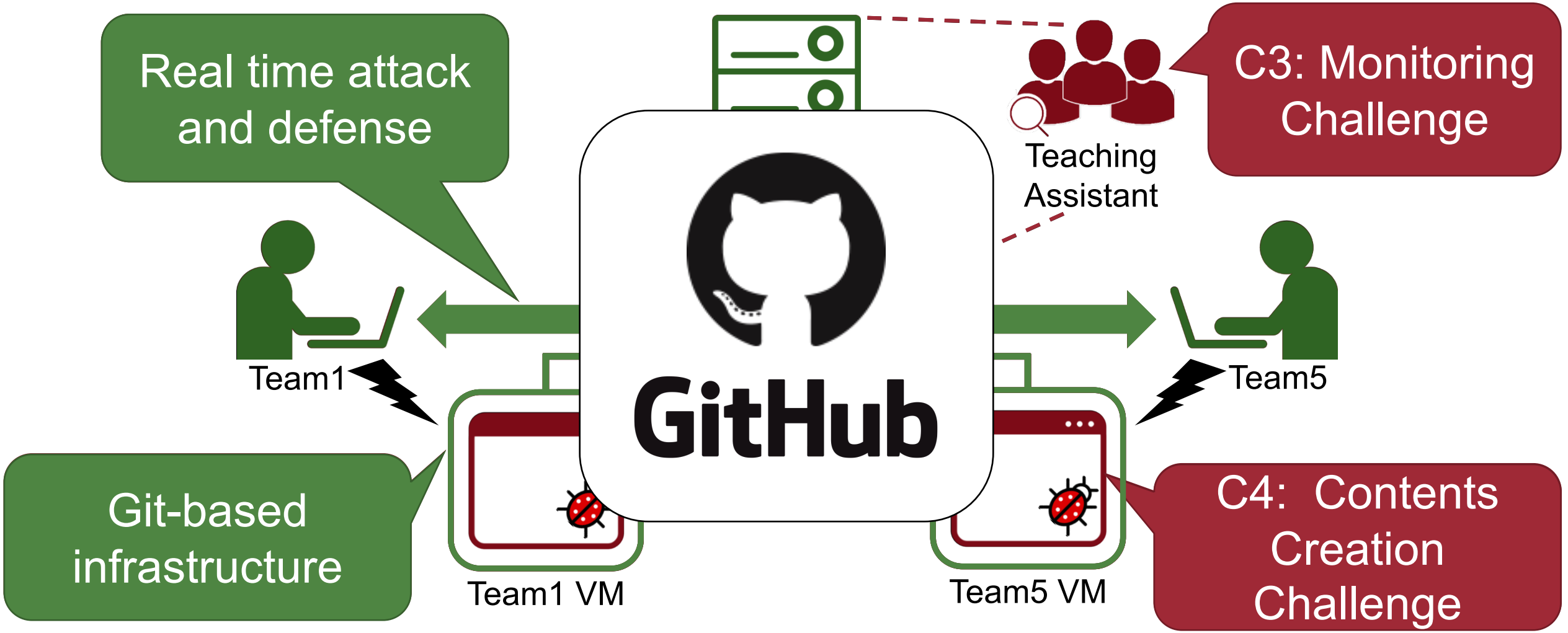


Periodically award points until it is fixed by the defending team

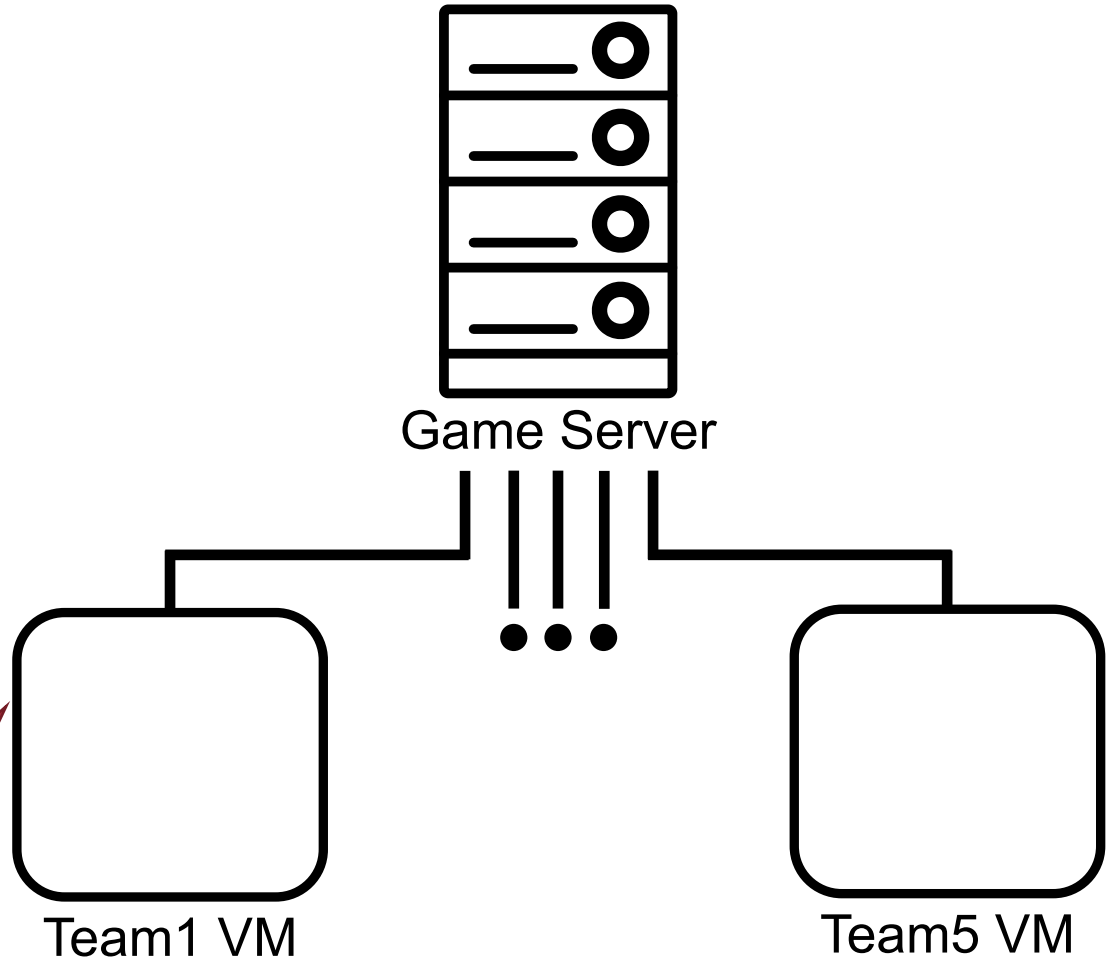
Repository as Scoreboard



Handle Configuration Challenge

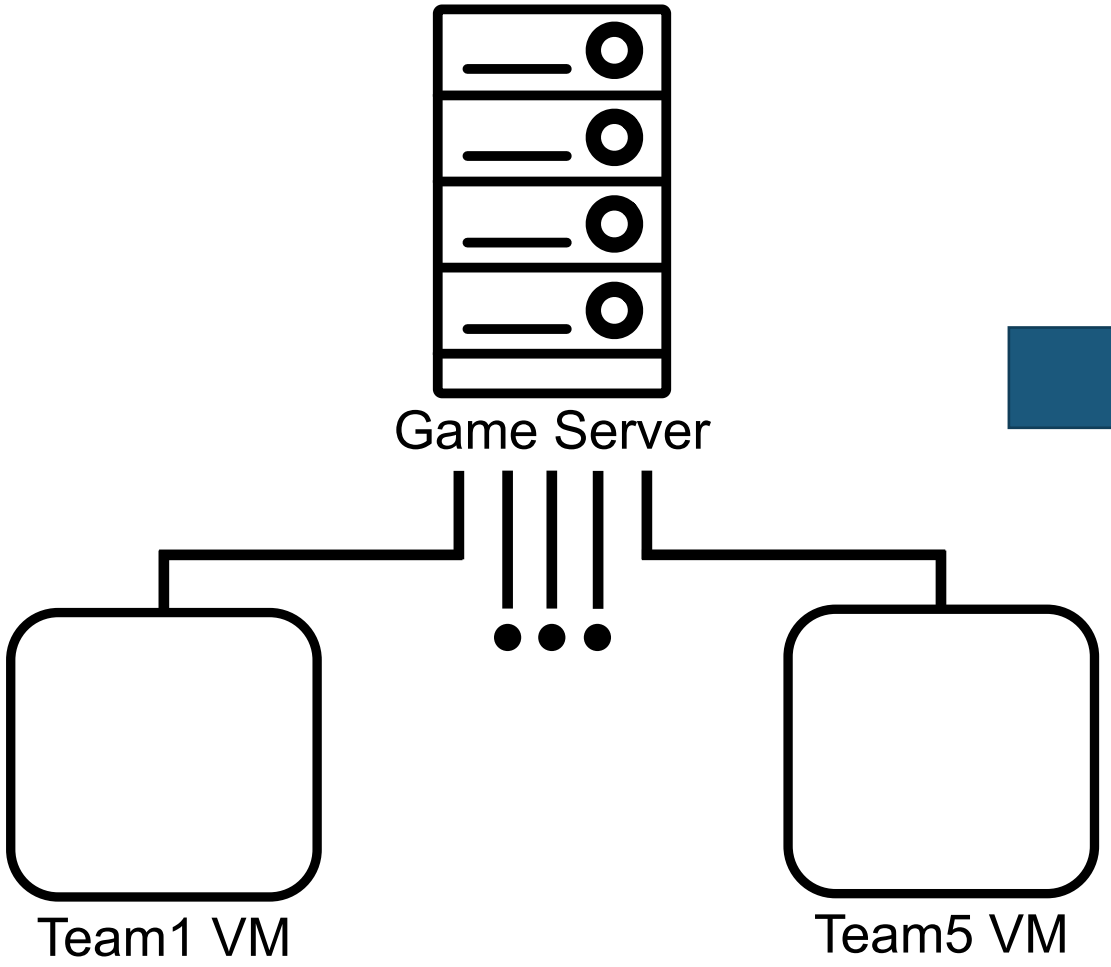


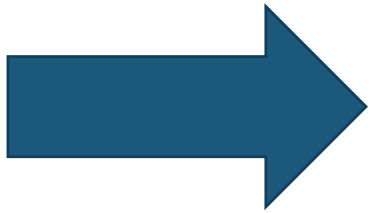
Git-based Infrastructure



Network, VM, DB,
Scoreboard, etc.

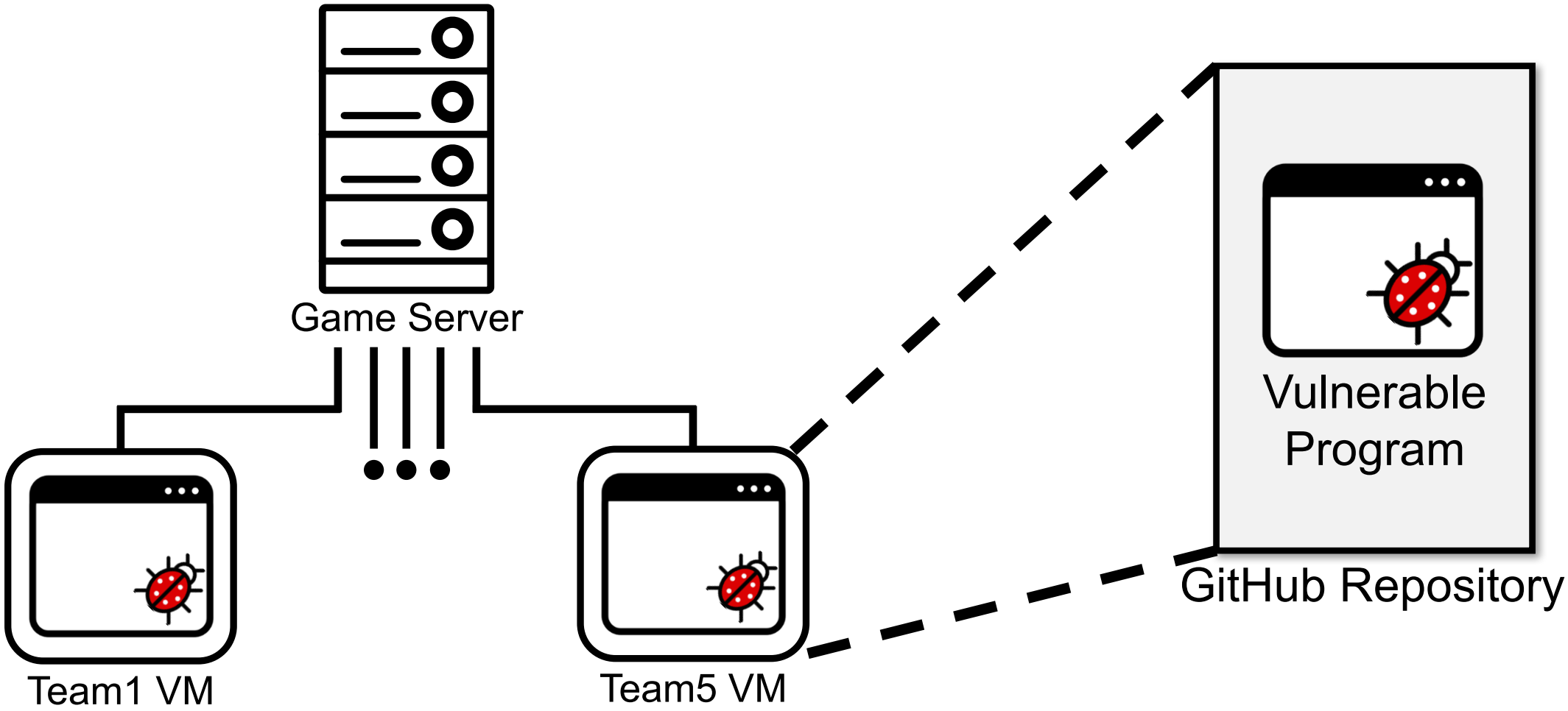
Git-based Infrastructure



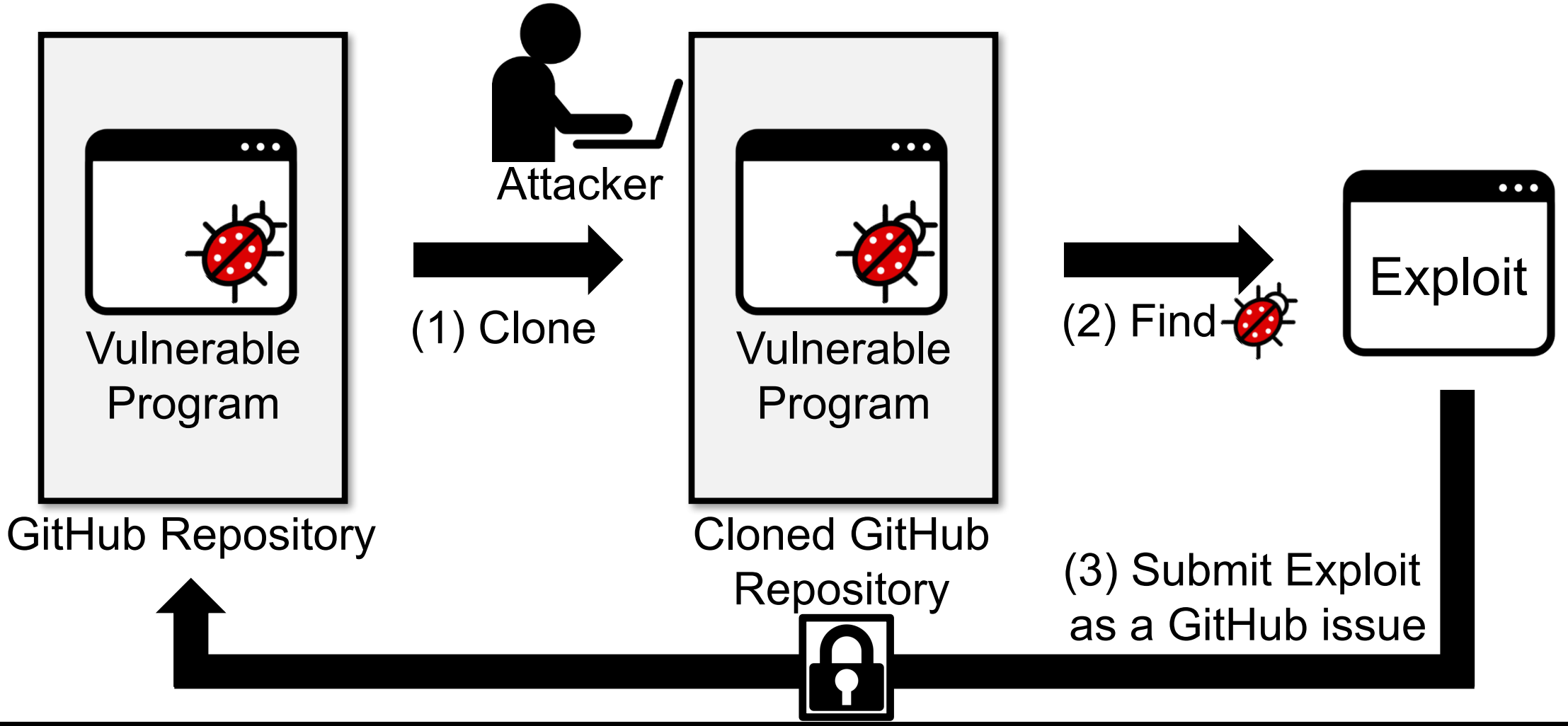




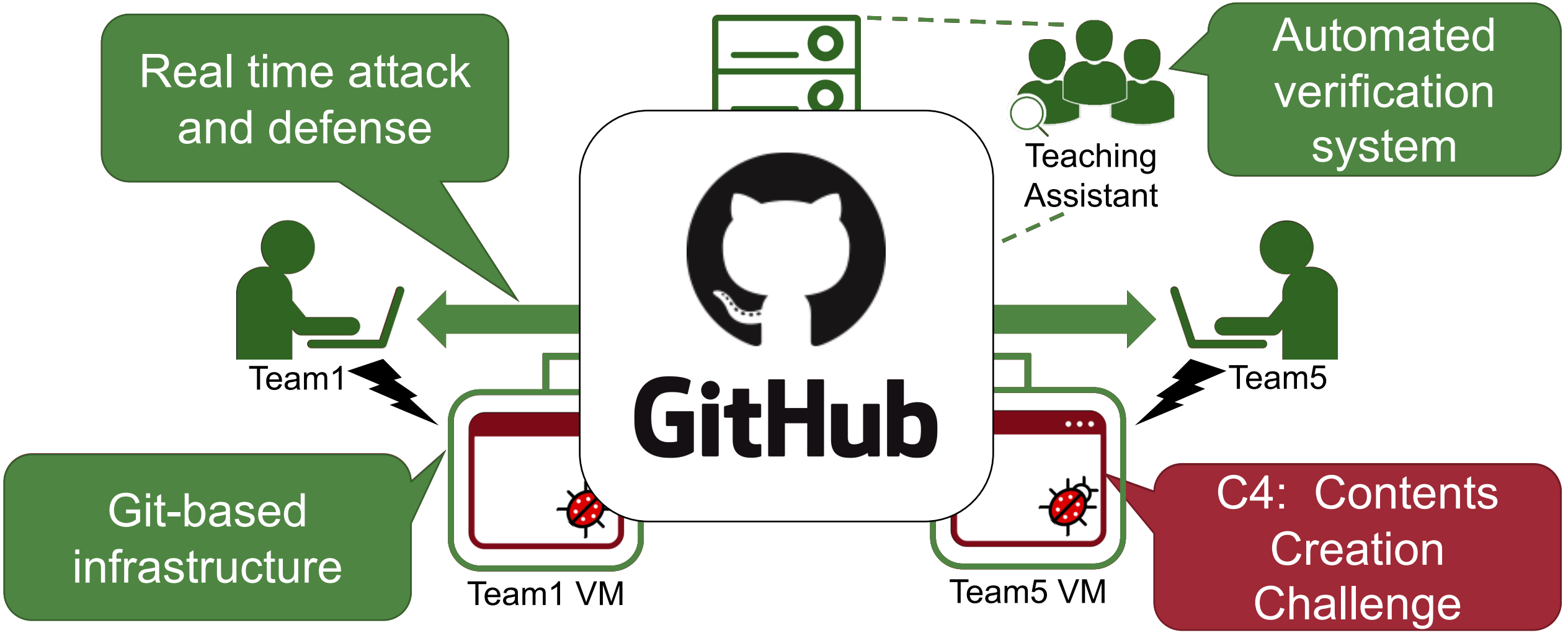
Git Repository as Service



Distributed System



Handle Monitoring Challenge

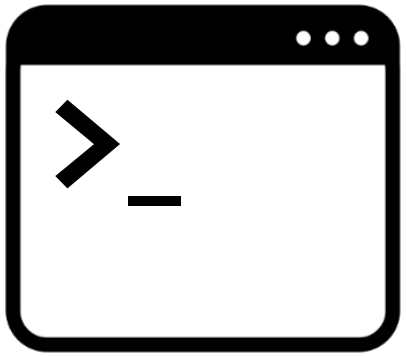


Automated Verification System



Teaching Assistant

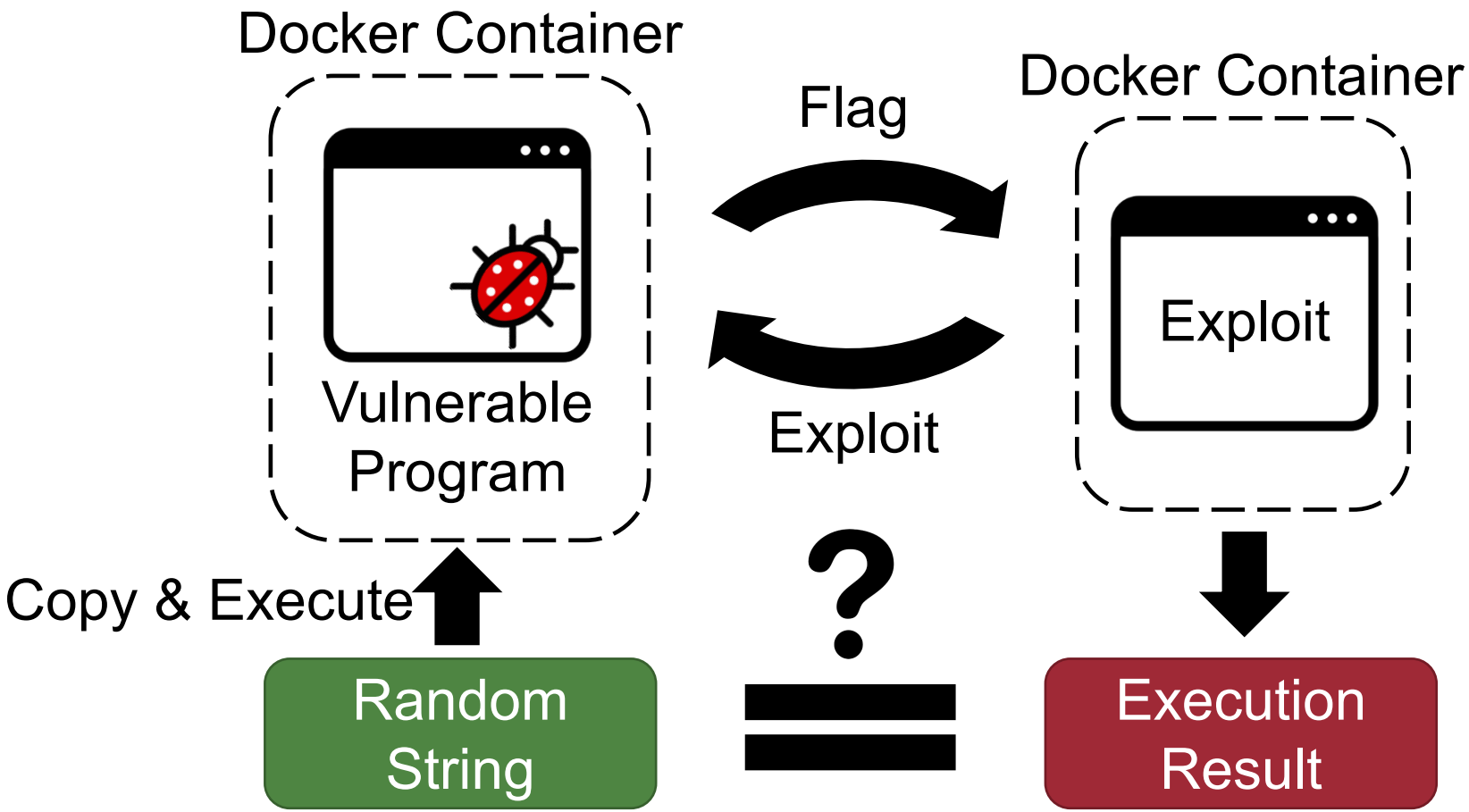
Execute



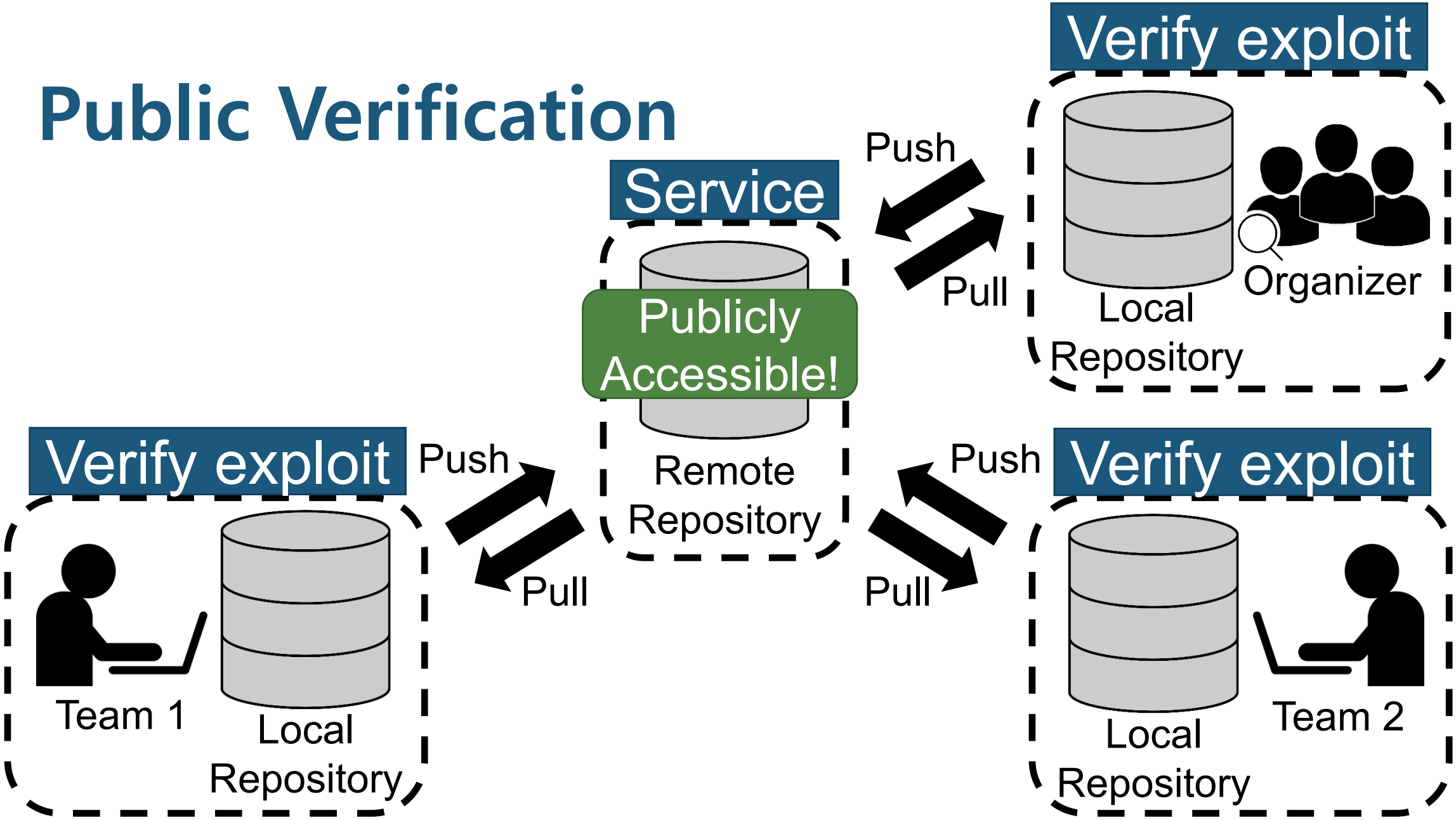
gitctf.py

- Verify exploit in each round
- Manage the game score

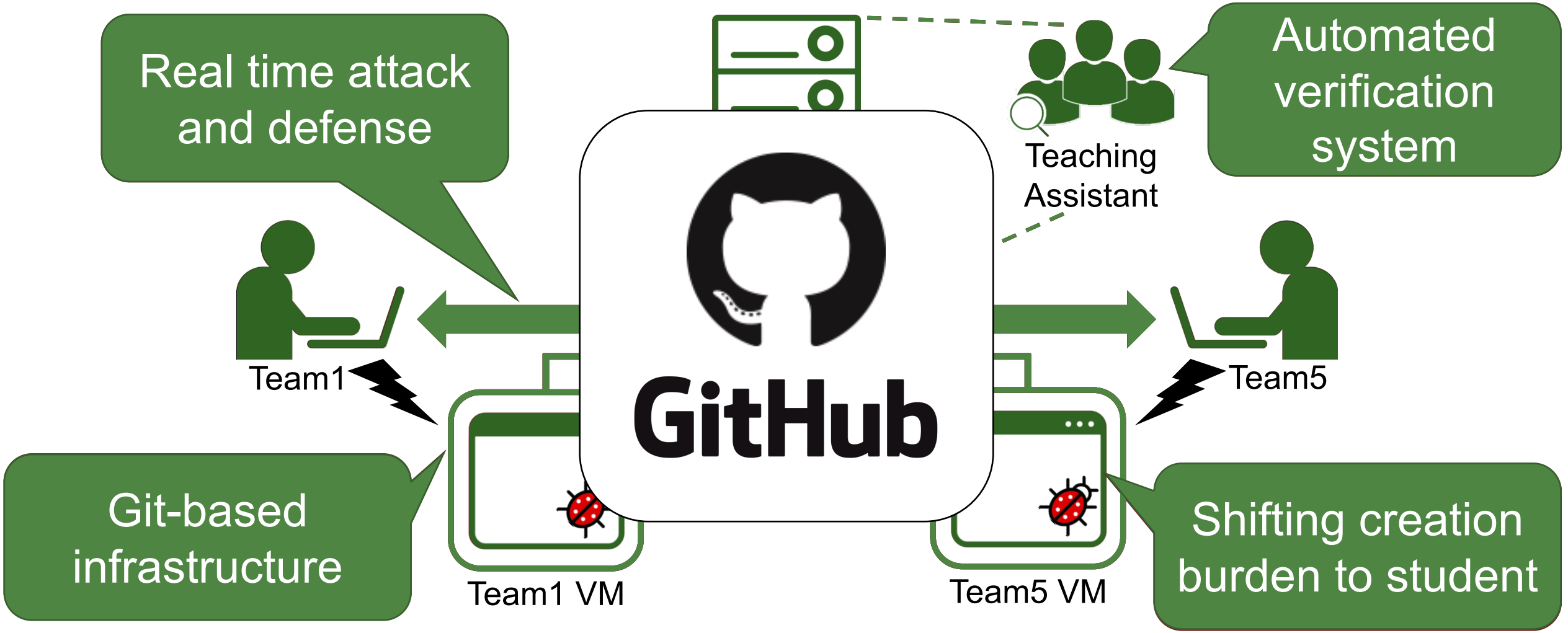
Automated Exploit Verification



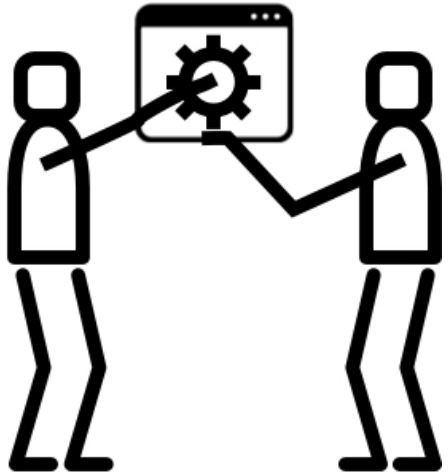
Public Verification



Handle Contents Creation Challenge

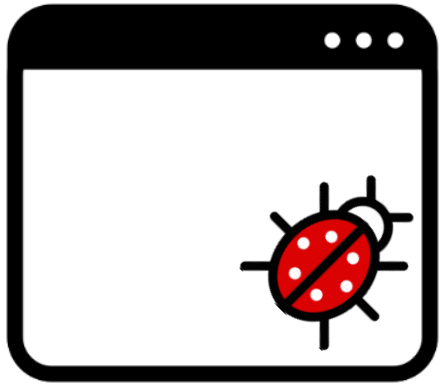


Shifting Creation Burden to Student (BIBIFI, *ACM CCS '16*)



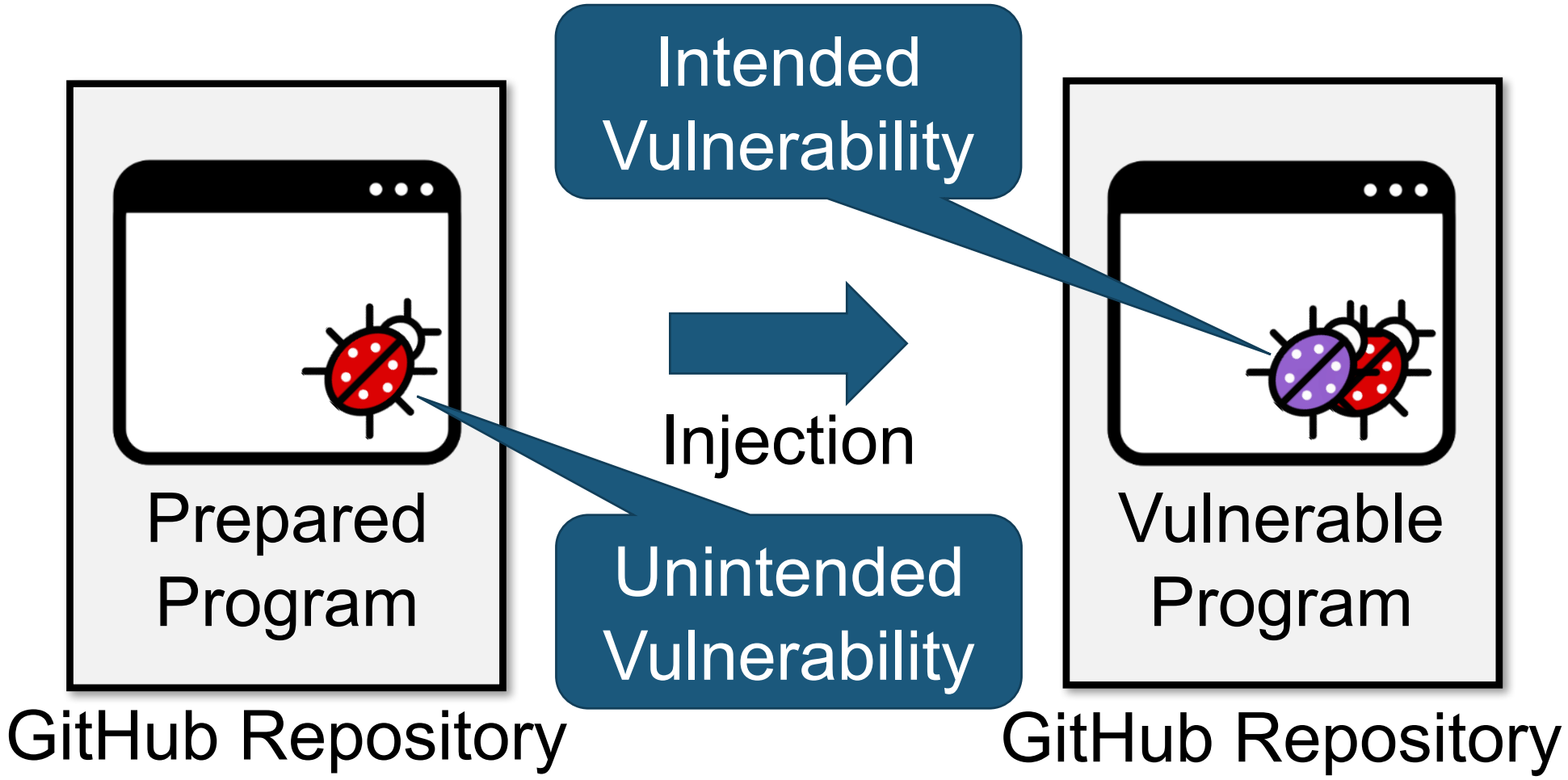
Hands-on Development



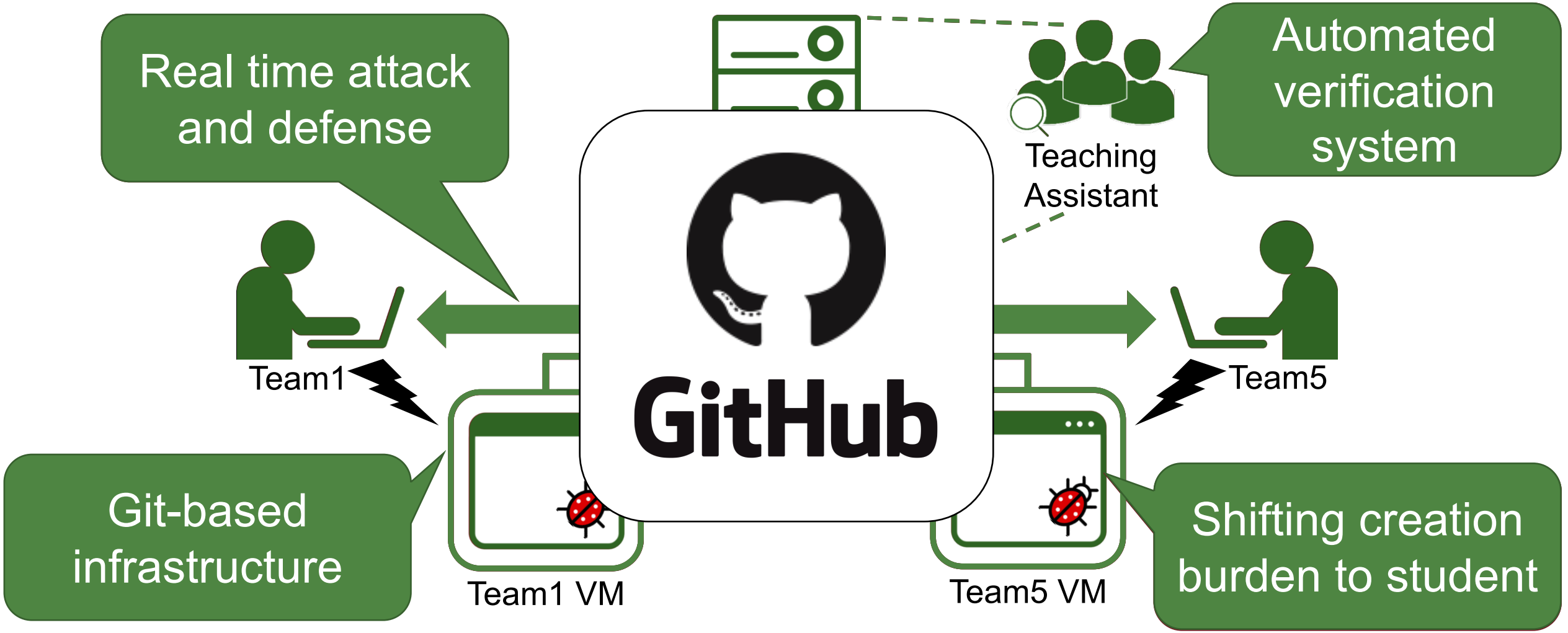


Prepared Program

Injecting Vulnerabilities



Our Goal: Handle All the Challenges



Evaluation Setup (Preliminary)

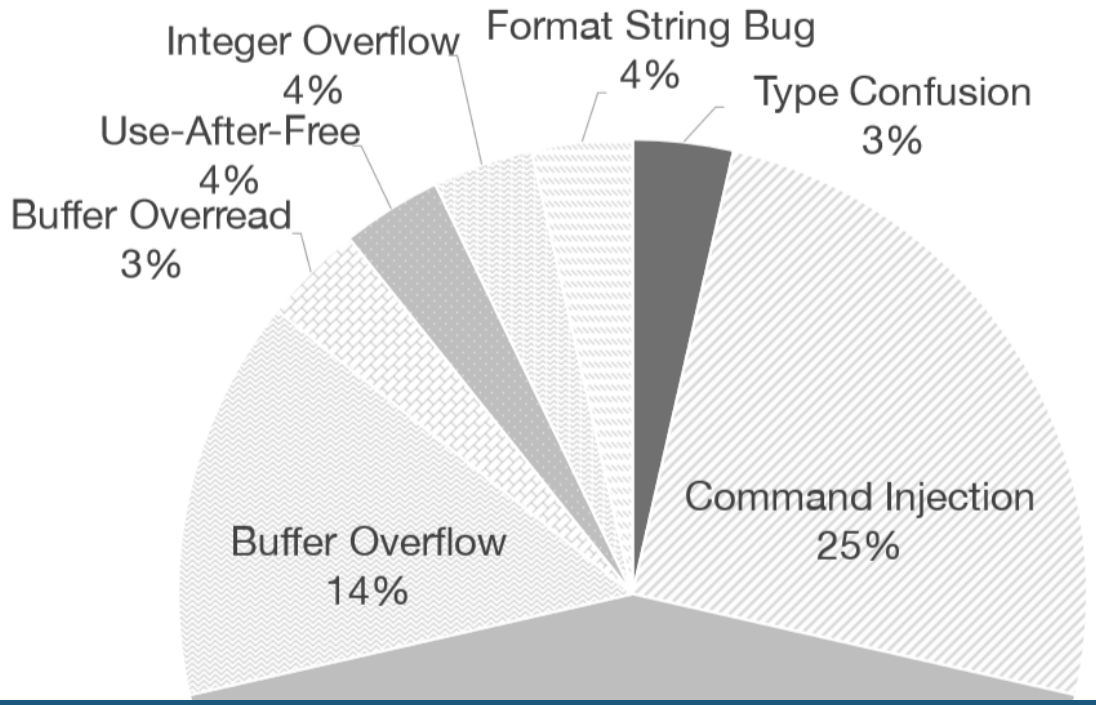
- ***IS521 Information Security Laboratory*** 2018 in *KAIST*, Korea

- 21 students (11 of them had no experience in CTF), 6 teams

- Preparation: Develop a simple secure messaging application (use either C or C++)
- Injection (Individual): Inject at least one vulnerability
- Exercise (Individual): Report unintended vulnerabilities or functionality bugs

Diversity of Injected Vulnerabilities

The students introduced 28 vulnerabilities in the 6 distinct applications

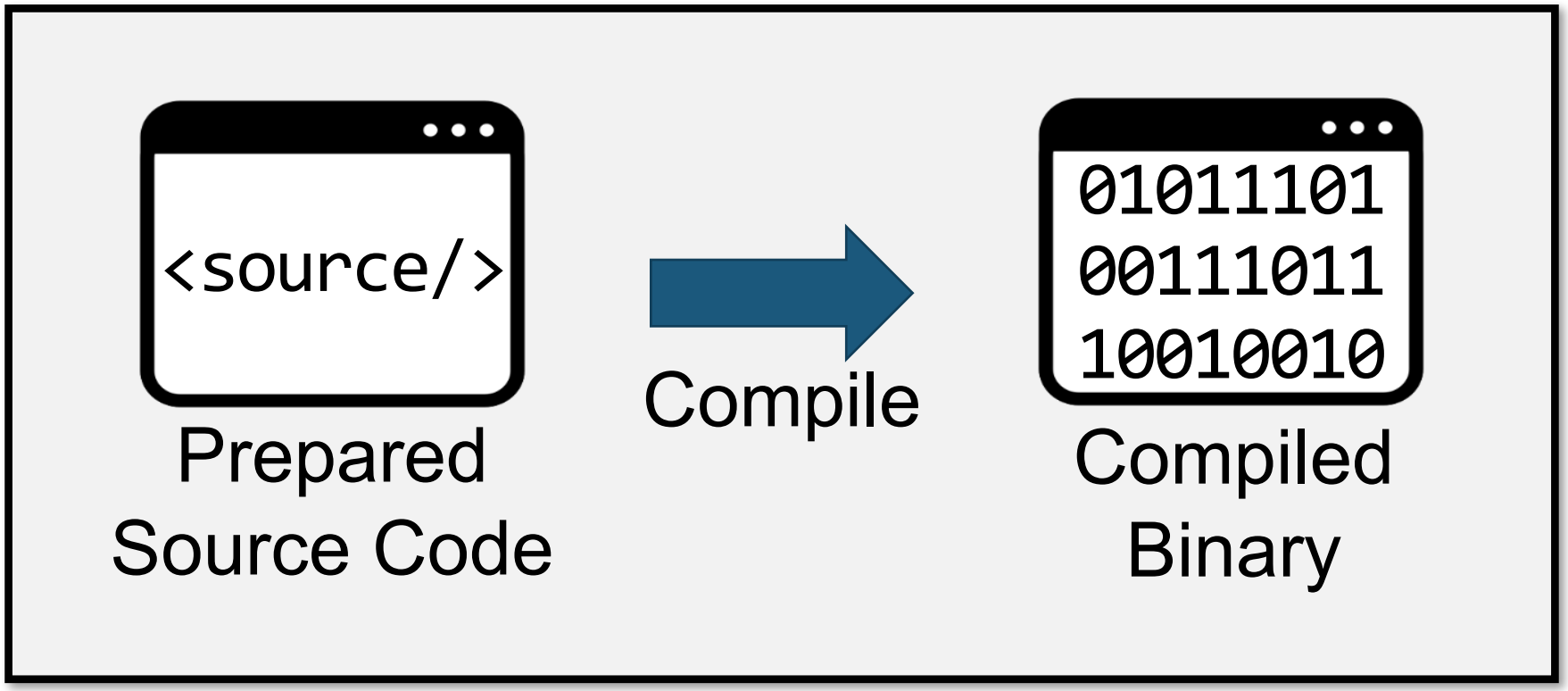


Help the instructors prepare a diverse set of CTF challenges

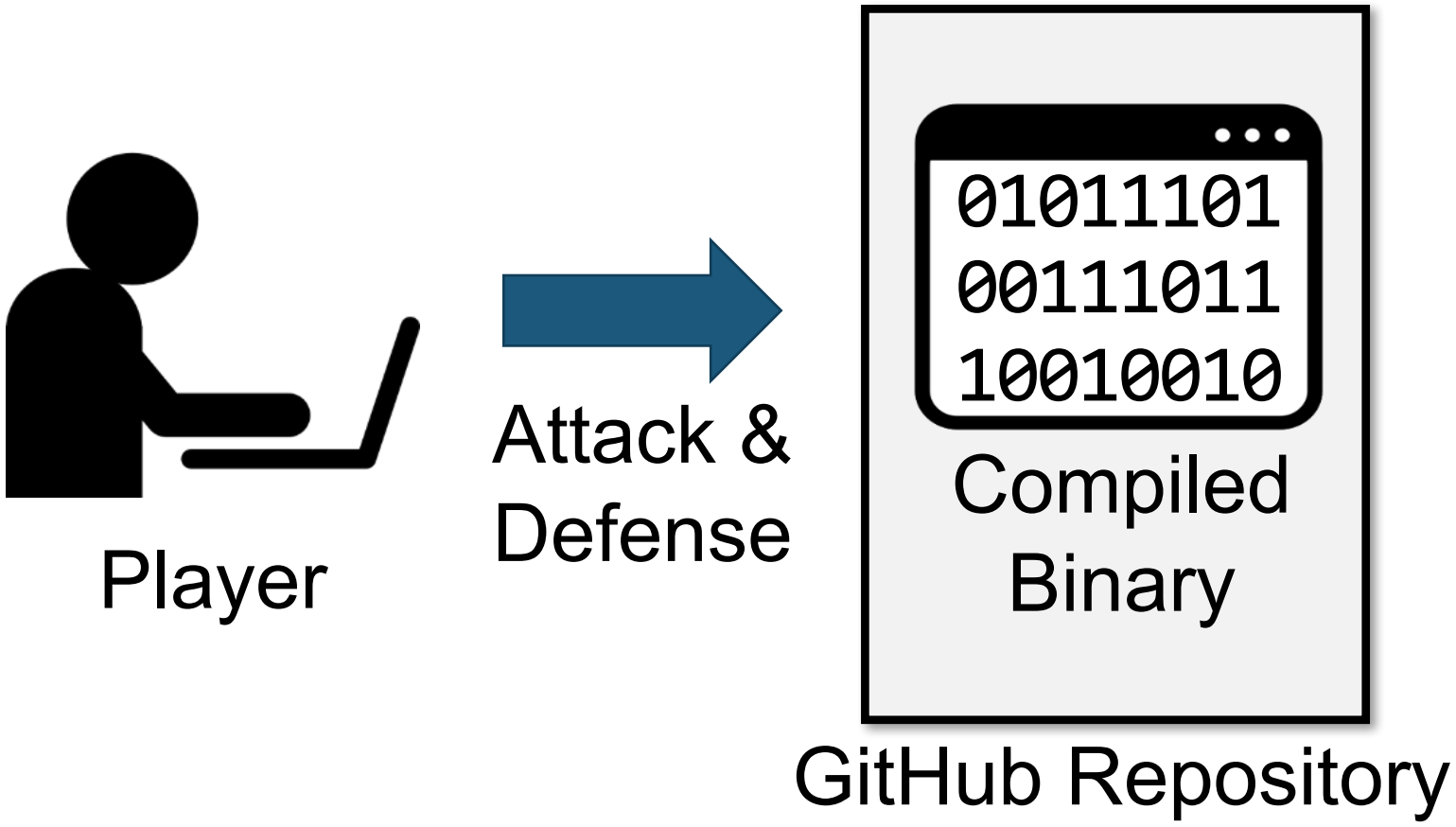
Exercise with Unintended Vulnerability

- 14 vulnerabilities and 18 functionality bugs were reported
- Each team had at least one unintended vulnerability
- Unintended vulnerabilities are found mostly by experienced students

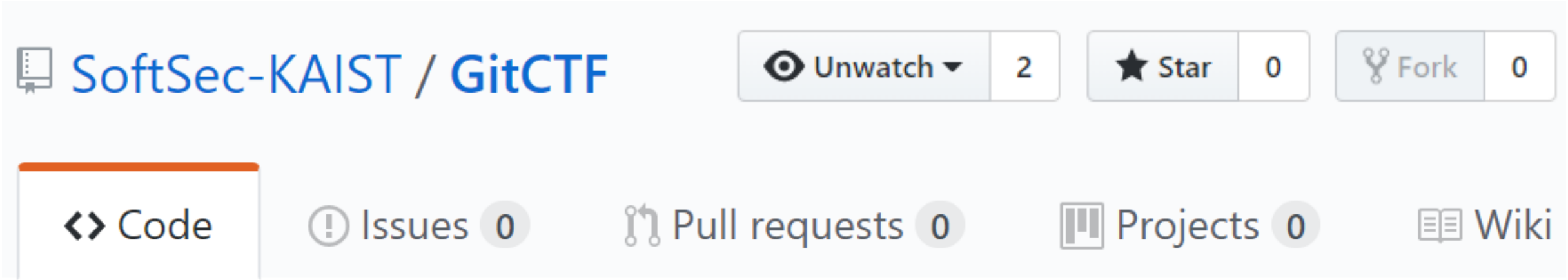
Discussion: Binary-Only CTF



Discussion: Binary-Only CTF



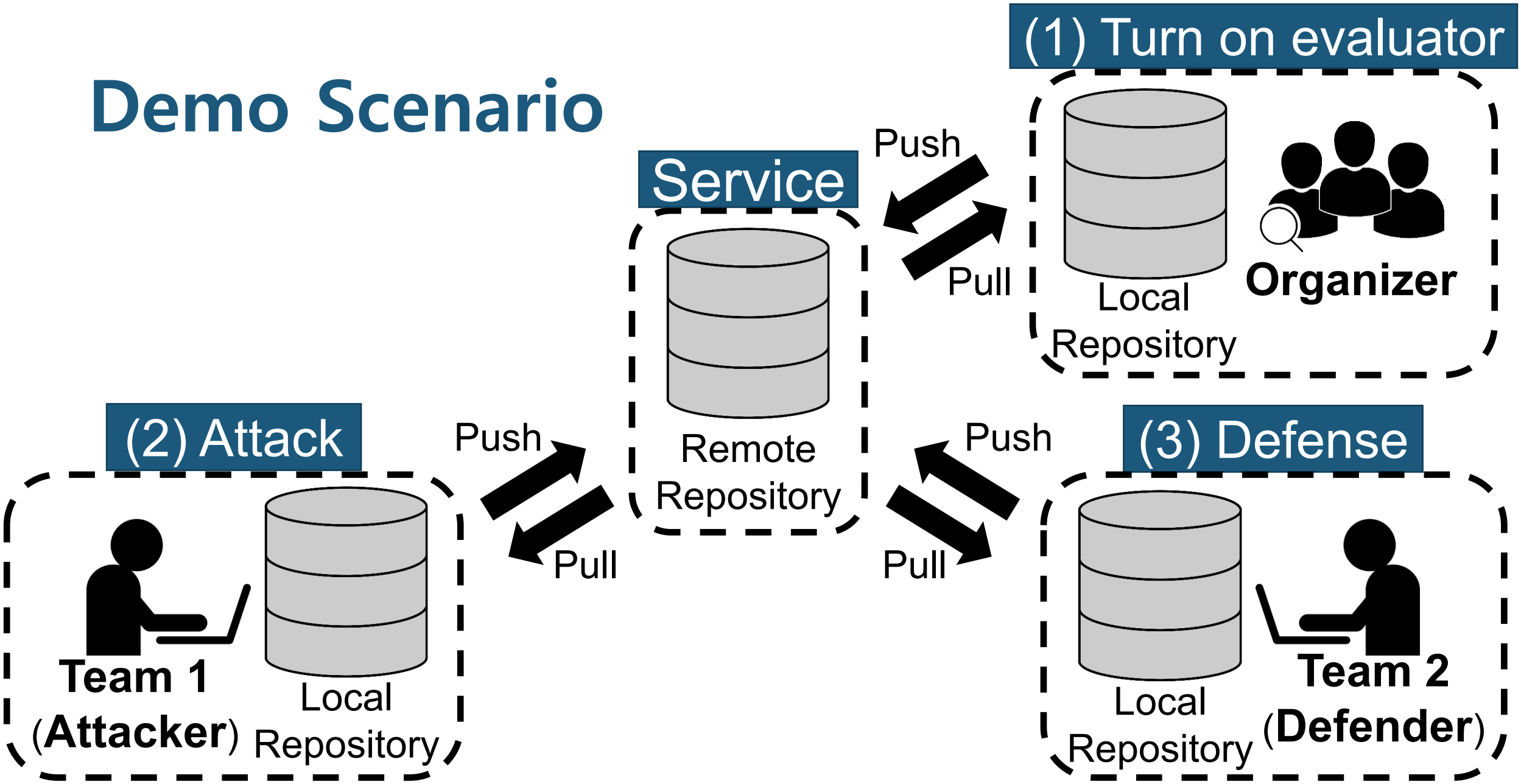
Open Science



https://github.com/SoftSec-KAIST/GitCTF

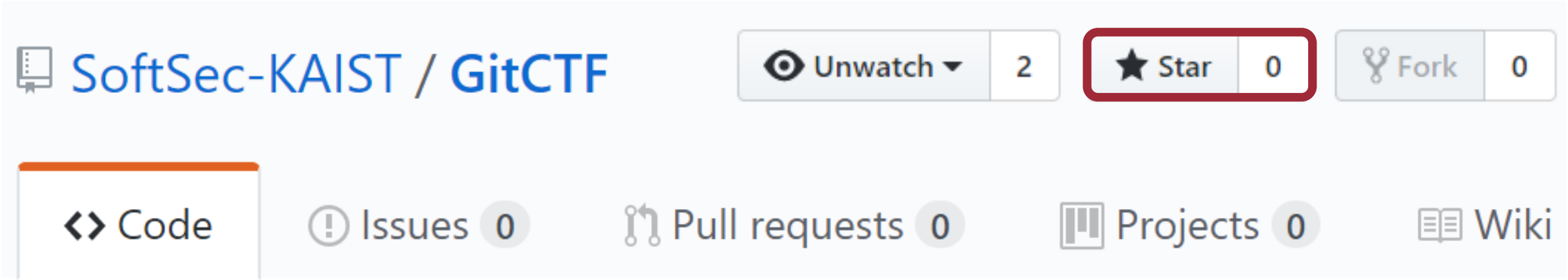
Demo

Demo Scenario



```
su3604@ubuntu:~/demo/evaluator$ ls
```

Open Science



https://github.com/SoftSec-KAIST/GitCTF

Question?