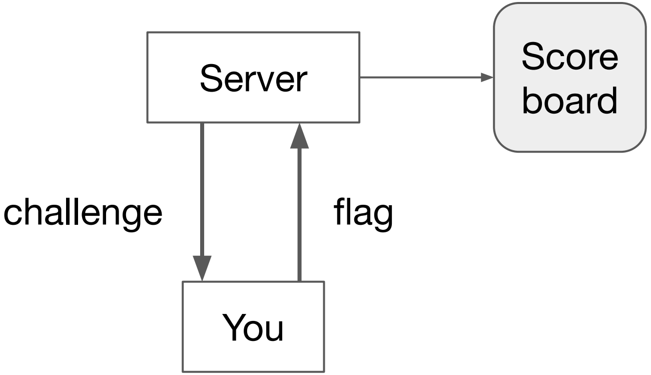
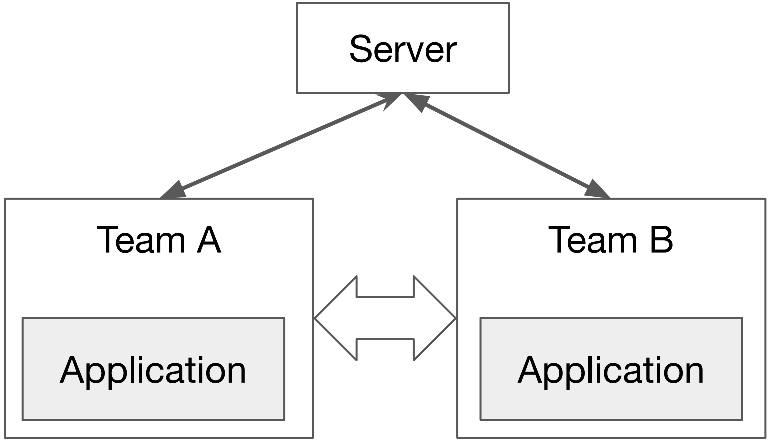
Capture The Flag

Lab 8-10 briefing

Introduction

- Capture The Flag
  - **Jeopardy style**
  - Attack-defense style
- DEFCON
- DARPA’s Cyber Grand Challenge



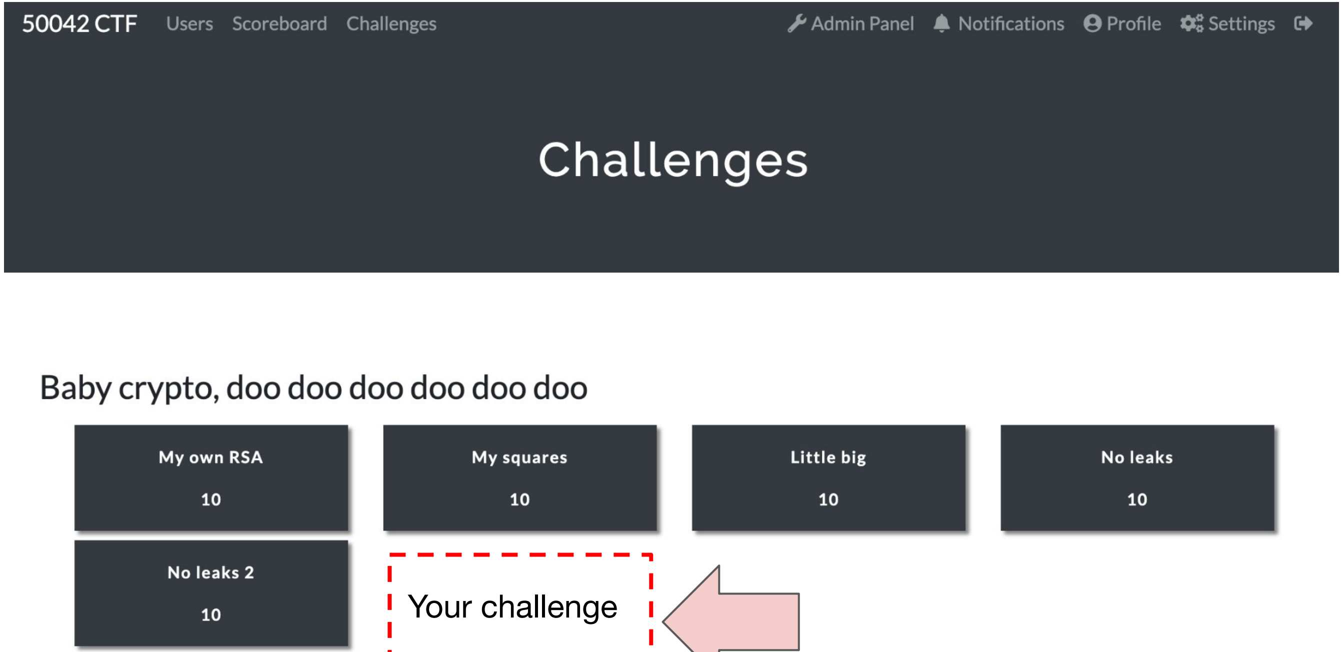


Introduction

- Each group (3-5 students) writes a challenge
  - For other groups to solve
- Deadlines:
  - **17/7, 23.59pm**: email TAs about your group
    - wei_lin@mymail.sutd.edu.sg
    - ganesh_subramanian@sutd.edu.sg
  - **29/7, 23.59pm**: challenge draft
  - **05/8, 23.59pm**: final submission
  - **08/8, 0.00am: CTF starts**
  - **12/8, 23.59pm: CTF ends**

CTF server

http://ec2-52-220-242-33.ap-southeast-1.compute.amazonaws.com:4000/



Challenges

Baby crypto, doo doo doo doo doo doo

My own RSA
10

My squares
10

Little big
10

No leaks
10

No leaks 2
10

Your challenge

Grading

- 12 points
  - **7 points** for your challenges
    - Technical aspects
    - Novelty
    - Fairness (to other groups)
    - Fun
  - **5 points** for solving other groups’ challenges
    - 0.5 points per solve
- The first 7 points are judged by us!
  - Our discretion
  - You can discuss with us before submission

Requirements on the challenges

- The intended flag has this format:
  - fcs22{XXXXXXXXXX….}
- Topic: **cryptography only**
  - Things that are covered in the class
  - Or outside of the class
    - Not too exotic
- Hardness
  - Must be solvable in reasonable time
    - If bruteforce is applicable, should only take minutes
  - **Does not require extensive readings on new concept**
  - Solutions do not require using special software
  - Hint: if the instructors and TA cannot solve it in reasonable time, maybe it’s too hard.

Requirements on the challenges

- Your challenge will be run on the CTF server
- Approach 1:
  - Write some program **P** that
    - Computes some output **o**
    - Based on a true flag **f**
  - Publish **o**
  - Publish a **stripped version of P**, called **P’**
    - With a fake flag **f’**
  - **(o,P’)** are uploaded to the server
  - Participant download it and works out the true flag

Requirements on the challenges

- Your challenge will be run on the CTF server
- Approach 2:
    - Write some program **P** that actually **runs on the CTF server**
        - Has a true flag **f**
        - Receives a user request (JSON)
        - Computes the response (JSON)
        - ***Can be stateful***
    - Publish the **network port** where the P is running
    - You can also publish the stripped down version of P
        - With a fake flag **f’**

Requirements on the challenges

- Question: do I have to publish **P’**
  - Not always, especially when
    - **P’** reveals too much about the technique to solve it
    - The logic is too well known
  - Check with us first!
- Question: why would people do approach 2?
  - Those programs model “cryptographic services”
  - Often, attacking them requires multiple interaction (outputs)

Requirements on the challenges

- How to make your P runs on the server (Approach 2)
  - Checkout **framework.zip**
  - See example on **`daemons/13370.py`**
  - You should be able to re-use most of it
    - Only need to write your Challenge object
  - Test it on your local machine:
    - Change the path in daemons/*.py
    - **`python3 daemon_manager.py -a`**
  - Connect to it and send requests
    - **`nc localhost <port>`**

Logistics

- **Office hours:**
  - Our offices, first come first serve
  - Thursday: 11.30am-1.30pm
  - Friday: 9-11am
- Challenge draft submission:
  - Simple text file describe your challenge (and intended solution)
- Final submission:
  - The code for the challenge
  - A write-up of the intended solution
    - You can find many CTF write-ups online

Advice

- Advice:
  - Google for CTF writeup to see examples
    - **DO NOT USE** existing challenges that have online write-up
  - Practice with the example challenges on our CTF server (***please do not DoS it***)
  - Think about some cool attacks (old or new) on crypto
  - Write your code that contains that cool attacks
- More advice:
  - Python **pycryptodome** is very useful
  - **Sage** is extremely good for breaking crypto, and has Python-like syntax

Good luck and have funs

- Email us to register your group by end of this week!