CSE 484 / CSE M 584: Computer Security and Privacy

Web security: Lab 2 and Context

Spring 2017

Jared Moore

jlcmoore@cs.uw.edu

Thanks to Franzi Roesner, Dan Boneh, Dieter Gollmann, Dan Halperin, Yoshi Kohno, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

Looking Forward

- Today: Introduction to Lab 2 + related concepts
- Wednesday & Friday: More web security

- Lab #2 out; due **5/19**
- Final Project Deadline #1 due **Friday**

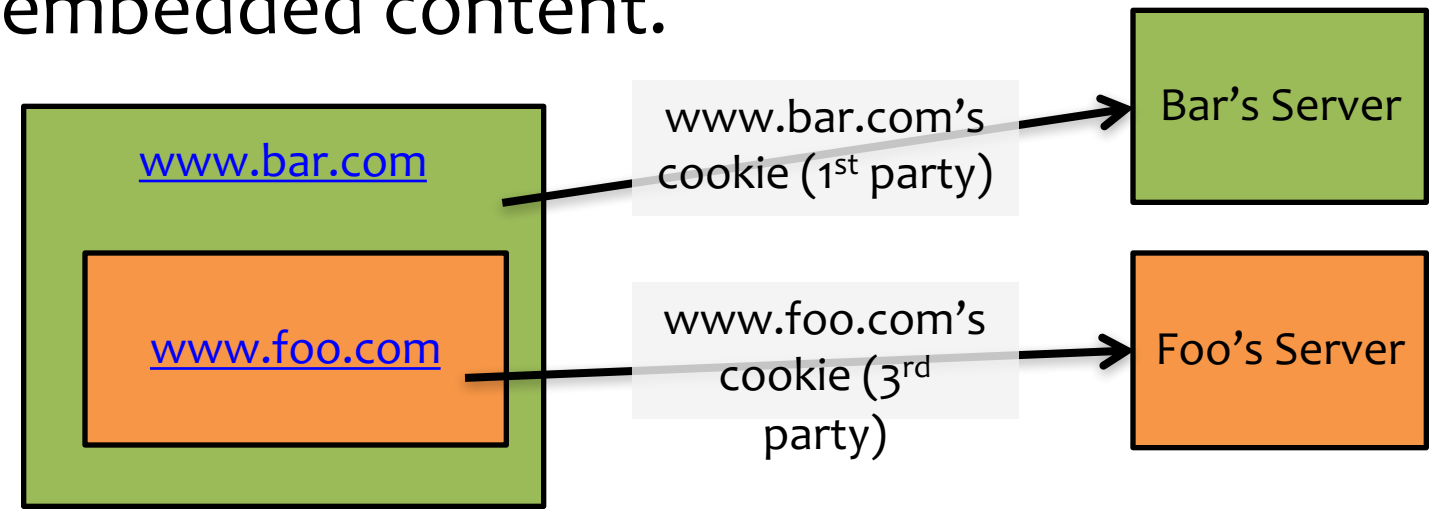
- Section this week: More lab 2 and clickjacking

Same-Origin Policy (Cookies)

- **For cookies:** Only code from same origin can read/write cookies associated with an origin.
  - Can be set via Javascript (`document.cookie=…`) or via `Set-Cookie` header in HTTP response.
  - Can narrow to subdomain/path (e.g., http://example.com can set cookie scoped to http://account.example.com/login.)
  - Secure cookie: send only via HTTPS.
  - HttpOnly cookie: can't access using JavaScript.

Same-Origin Policy (Cookies)

- Browsers automatically include cookies with HTTP requests.
- **First-party cookie:** belongs to top-level domain.
- **Third-party cookie:** belongs to domain of embedded content.



XSS: Cross-Site Scripting

- **Idea:** Place user-provided data in the page.
  - Makes page more interactive and personal.
- **Threat:** Improperly used data can be interpreted as code.
- **Solutions?**
  - Sanitize/validate input. (e.g., `htmlspecialchars()`)
  - Browser detection/prevention.

Server Side Scripts Review

- Before a webpage is sent to you, code is executed by the server
- Can be use to set and read cookies for authentication
- You will need a basic script to receive captured cookies
- We will use PHP

Lab 2

Overview

- Pikachu, Meowth, and Cookies
  - XSS; **Today**
- Jailbreak
  - SQL Injection; **Today** if time
- Hack your 4.0!
  - **Wednesday** or **Friday**

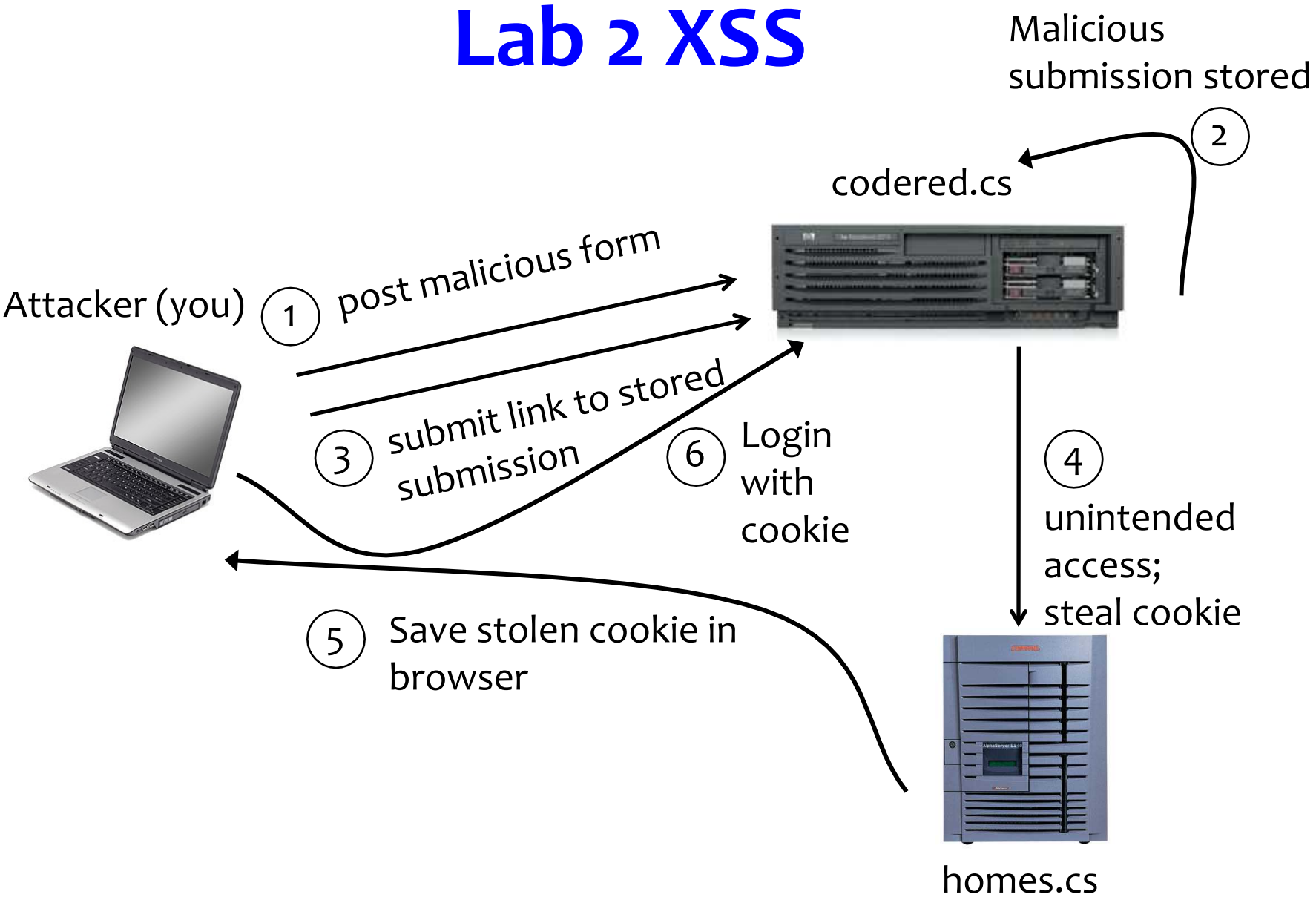
Lab 2 XSS

- Give the TAs (codered.cs) a link with a XSS vulnerability.
- TAs will 'visit' this link, and cookie will be stolen.
- The process of stealing cookie involves sending it to a place you control.
- Save the cookie, read it, and use it to log in

Tools

- Web browser (Firefox or Chrome)
- Cookie editing extension (Firebug for Firefox)
- A php script on homes.cs to capture cookies
- (see lab details)

Lab 2 XSS


Malicious submission stored
2
codered.cs
Attacker (you)
1
post malicious form
3
submit link to stored submission
6
Login with cookie
4
unintended access; steal cookie
5
Save stolen cookie in browser
homes.cs

Demo