Computer System- B Security

Introduction to Web Security P3

Cookies, Phishing, SQL injections

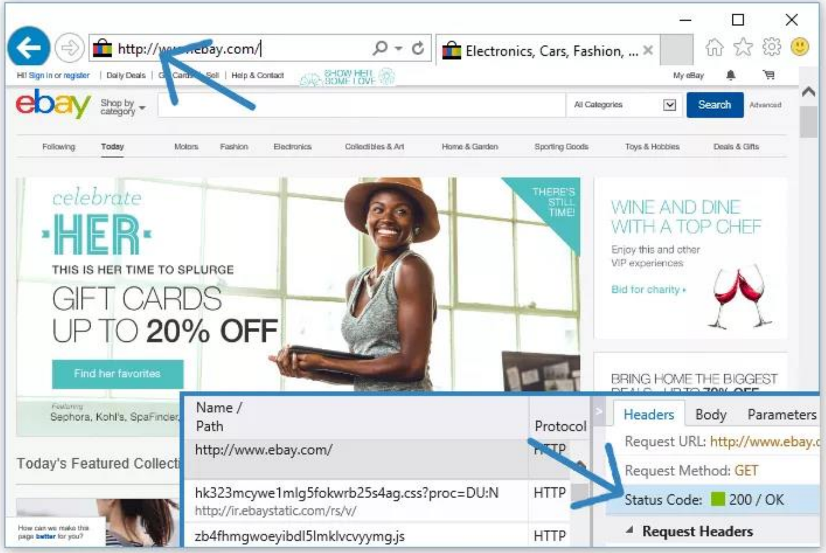
Alma Oracevic

bristol.ac.uk

HTTP vs HTTPS

- HTTP send request/response in clear text
  - Information can be sniffed (confidentiality is lost)
- We do not know if we are connected to the right server
  - Identity/authenticity is not variable.
- HTTPS (secure) solves this by using crypto.
  - Encryption
  - Signature
  - MAC
- Example: SSL/TSL (later in the unit)

HTTP vs HTTPS



http://www.ebay.com/
Electronics, Cars, Fashion, ...
Hi! Sign in or register | Daily Deals | Gift Cards | Sell | Help & Contact
My eBay
ebay
Shop by category
All Categories
Search
Advanced
Following
Today
Motors
Fashion
Electronics
Collectibles & Art
Home & Garden
Sporting Goods
Toys & Hobbies
Deals & Gifts
celebrate HER
THIS IS HER TIME TO SPLURGE
GIFT CARDS UP TO 20% OFF
Find her favorites
THERE'S STILL TIME!
WINE AND DINE WITH A TOP CHEF
Enjoy this and other VIP experiences
Bid for charity ›
BRING HOME THE BIGGEST
Featuring Sephora, Kohl's, SpaFinder
Today's Featured Collecti
How can we make this page better for you?

Name / Path	Protocol
http://www.ebay.com/	HTTP
hk323mcywe1mlg5fokwrb25s4ag.css?proc=DU:N http://ir.ebaystatic.com/rs/v/	HTTP
zb4fhmgwoeyibdl5lmklvcvyymg.js	HTTP

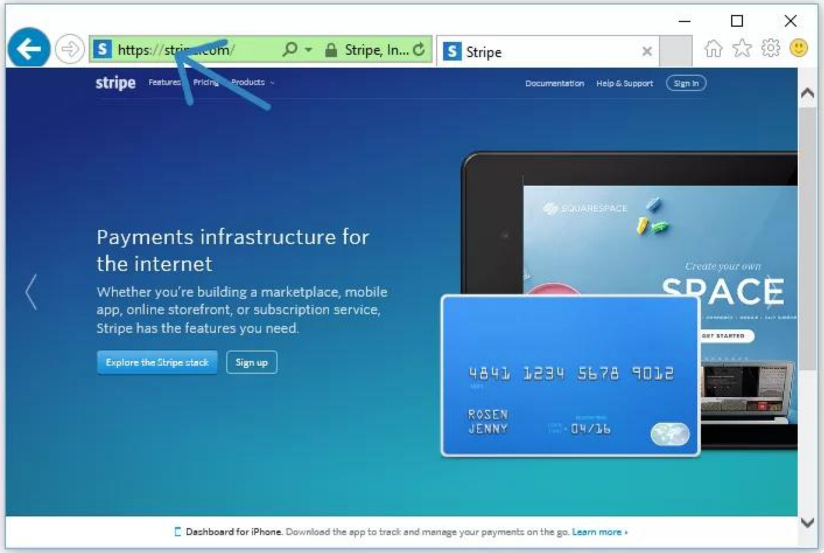
Headers | Body | Parameters

Request URL: http://www.ebay.c

Request Method: GET

Status Code: 🟩 200 / OK

Request Headers



Payments infrastructure for the internet

Whether you're building a marketplace, mobile app, online storefront, or subscription service, Stripe has the features you need.

Explore the Stripe stack | Sign up

Dashboard for iPhone. Download the app to track and manage your payments on the go. Learn more ›

HTTP is stateless

HTTP is stateless

- The notion of a **session**
  - encapsulates information about a visitor
  - allows user to relate multiple requests

HTTP is stateless

- The notion of a **session**
  - encapsulates information about a visitor
  - allows user to relate multiple requests
- Session information should be considered extremely sensitive

HTTP is stateless

- The notion of a **session**
  - encapsulates information about a visitor
  - Allows user to relate multiple requests
- Session information should be considered extremely sensitive
- Thus, a class of attacks known as session hijacking.

Session Hijacking

- Leakage of HTTP session information may lead to an attack called *session hijacking*.
  - Stealing of session ID/cookies allows an attacker to impersonate an ongoing session
  - Replay of a session to repeat some important action.

Cookies

Cookies

- Small packets of data, called cookies, which are sent to the client by the web server and stored on the client’s machine.

Cookies

- Small packets of data, called cookies, which are sent to the client by the web server and stored on the client’s machine.
- When the user revisits the web site, these cookies are returned, unchanged, to the server, which can then “remember” that user and access their session information.

Cookies

- Small packets of data, called cookies, which are sent to the client by the web server and stored on the client’s machine.
- When the user revisits the web site, these cookies are returned, unchanged, to the server, which can then “remember” that user and access their session information.
- SOP policy is applicable to who access the cookies

Cookies

- Small packets of data, called cookies, which are sent to the client by the web server and stored on the client's machine.
- When the user revisits the web site, these cookies are returned, unchanged, to the server, which can then "remember" that user and access their session information.
- SOP policy is applicable to who access the cookies
- Contains sensitive information!

Phishing

- Forged web pages created to fraudulently acquire sensitive information.

Click-Jacking

Click-Jacking

- Click-jacking is a form of web site exploitation.
- A user's mouse click on a page is used in a way that was not intended by the user
- For example
- <a onMouseUp=window.open("http://www.evilsite.com") href="http://www.trustedsite.com/">Trust me!</a>

SQL Injection Attack

- Web Security

SQL Injection Attack

- Many web applications take user input from a form
- Often this user input is used literally in the construction of a SQL query submitted to a database. For example:
  - SELECT user FROM table WHERE name = ‘user_input’;

SQL Injection Attack

- Many web applications take user input from a form
- Often this user input is used literally in the construction of a SQL query submitted to a database. For example:
  - SELECT   user FROM table WHERE name = ‘user_input’;

SQL Injection Attack

- Many web applications take user input from a form
- Often this user input is used literally in the construction of a SQL query submitted to a database. For example:
  - SELECT user FROM table WHERE name = ‘user_input’;
- An SQL injection attack involves placing SQL statements in the user input (again, data-code confusion!)

SQL:
Standard
Query
Language

SQL lets you access and manage (Query) databases

A database is a large collection of data organized in tables for rapid search and retrieval, with row and columns

SQL: Standard Query Language

A database is a large collection of data organized in tables for rapid search and retrieval, with row and columns.

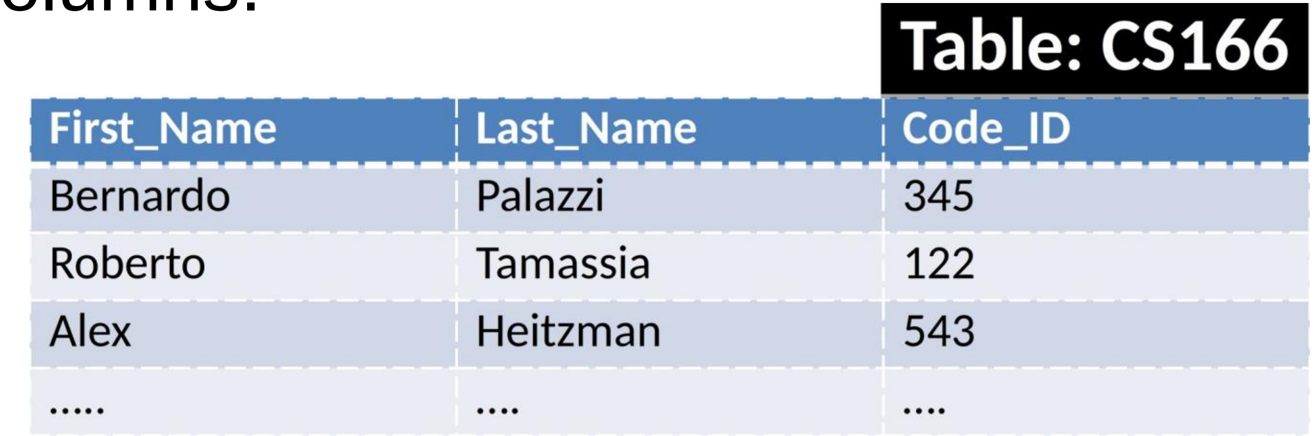
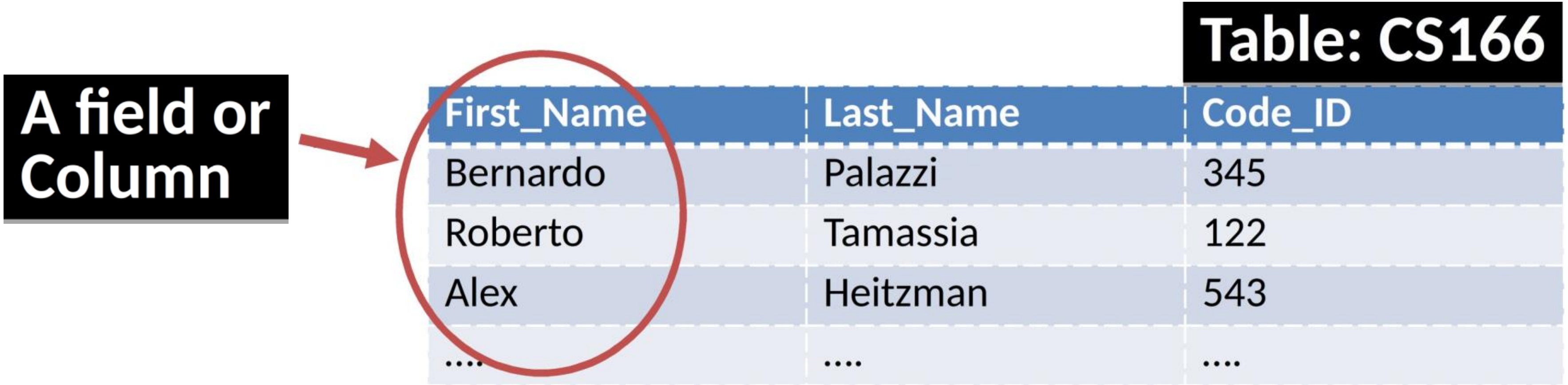


Table: CS166

First_Name	Last_Name	Code_ID
Bernardo	Palazzi	345
Roberto	Tamassia	122
Alex	Heitzman	543
…..	….	….

SQL: Standard Query Language

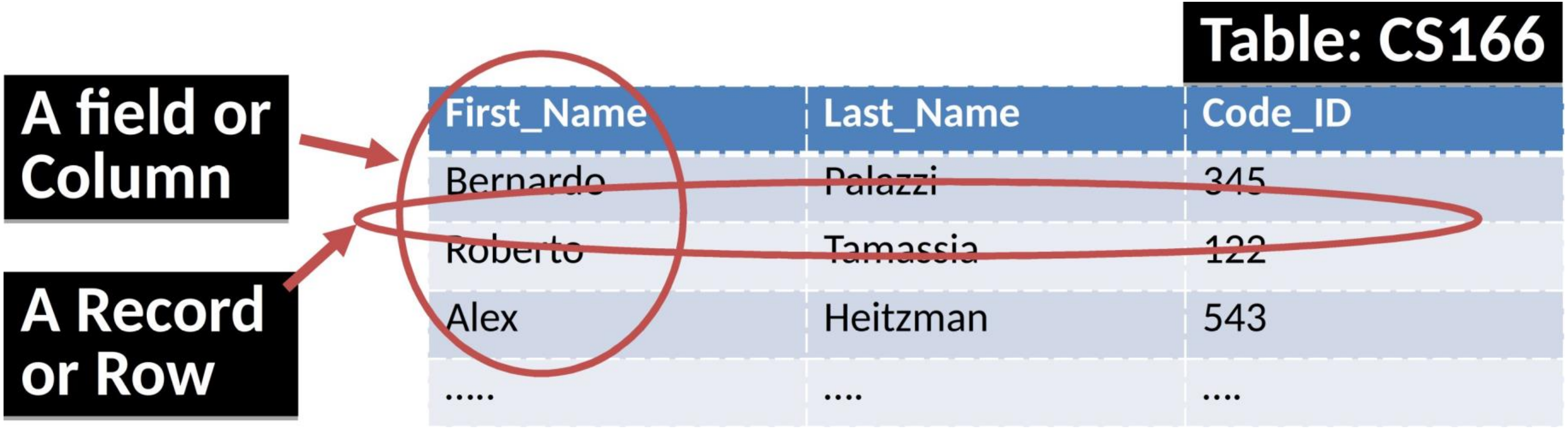


A field or Column

Table: CS166

First_Name	Last_Name	Code_ID
Bernardo	Palazzi	345
Roberto	Tamassia	122
Alex	Heitzman	543
….	….	….

SQL: Standard Query Language



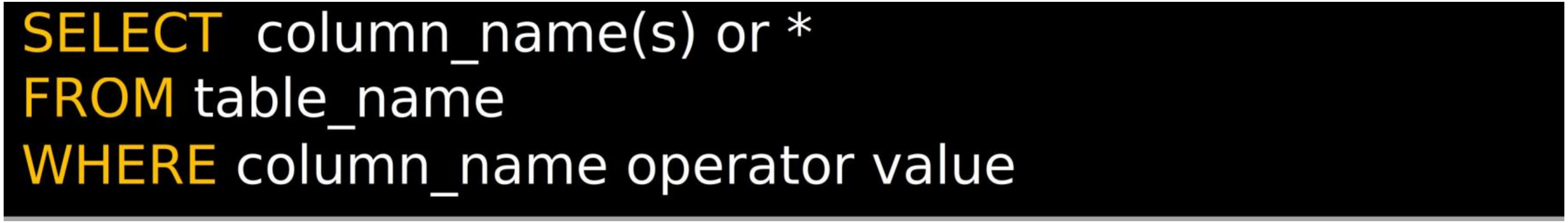
A field or Column

A Record or Row

Table: CS166

First_Name	Last_Name	Code_ID
Bernardo	Palazzi	345
Roberto	Tamassia	122
Alex	Heitzman	543
.....	....	....

SQL Syntax



```
SELECT  column_name(s) or *
FROM table_name
WHERE column_name operator value
```

- SELECT statement is used to select data FROM one or more tables in a database

•WHERE clause is used to filter records

Result-set is stored in a result table

- ;is statement terminator and --is remark beginning

What did we learn today?