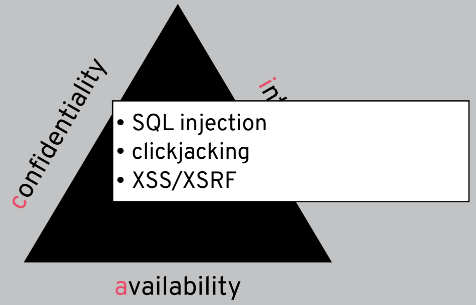
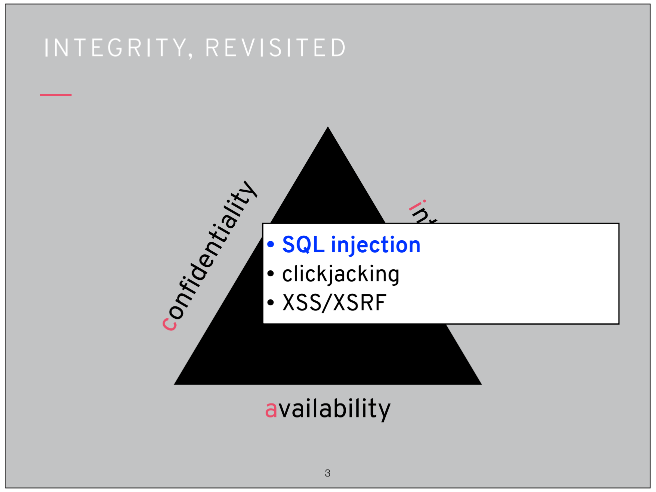
SECURITY (COMP0141): ATTACKS ON INTEGRITY

INTEGRITY, REVISITED



- SQL injection
- clickjacking
- XSS/XSRF

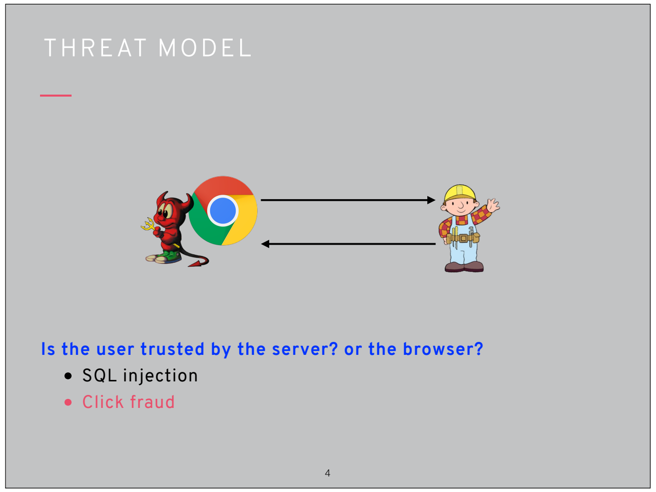


INTEGRITY, REVISITED

confidentiality
int
availability

- **SQL injection**
- clickjacking
- XSS/XSRF

3



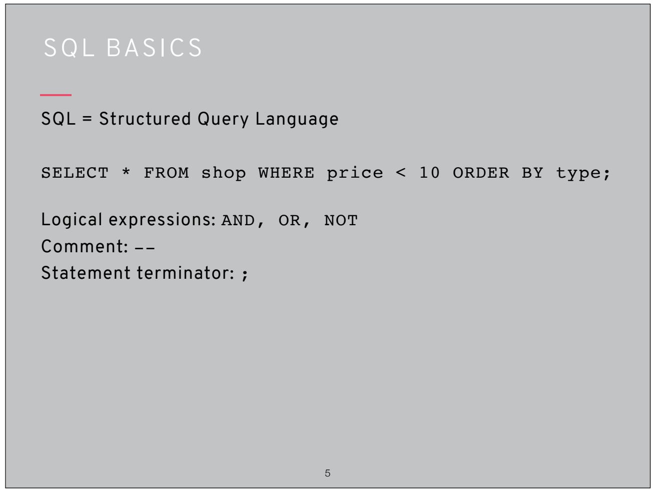
THREAT MODEL

Is the user trusted by the server? or the browser?

- SQL injection
- Click fraud

4

We won't go over click fraud again but this also falls into this category of attack



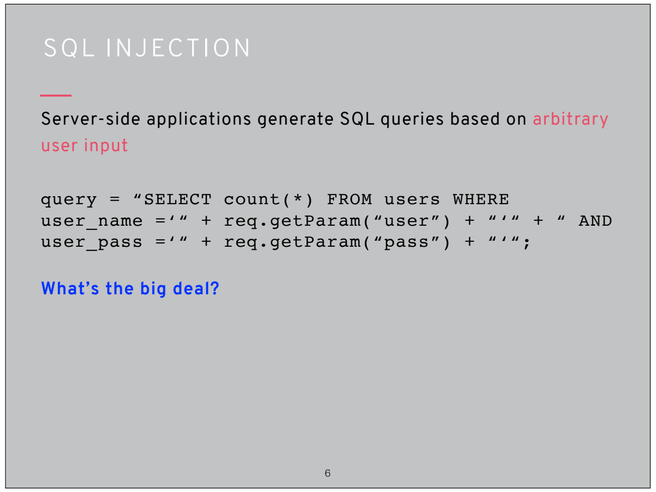
SQL BASICS

SQL = Structured Query Language

```
SELECT * FROM shop WHERE price < 10 ORDER BY type;
```

Logical expressions: `AND, OR, NOT`
Comment: `--`
Statement terminator: `;`

5



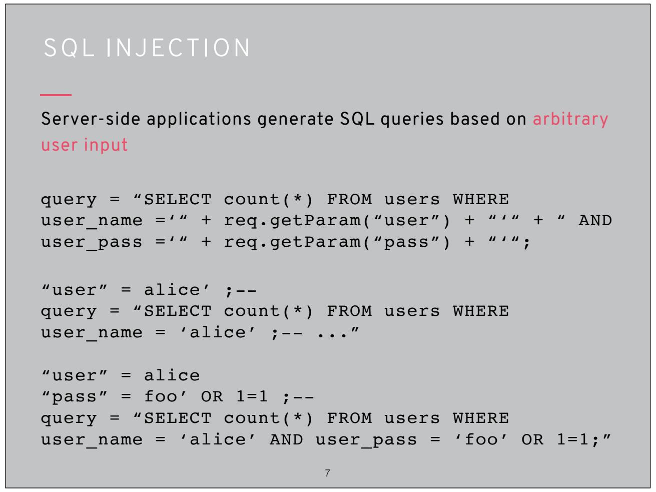
SQL INJECTION

Server-side applications generate SQL queries based on arbitrary user input

```
query = “SELECT count(*) FROM users WHERE
user_name =‘“ + req.getParam(“user”) + “‘“ + “ AND
user_pass =‘“ + req.getParam(“pass”) + “‘“;
```

What’s the big deal?

6



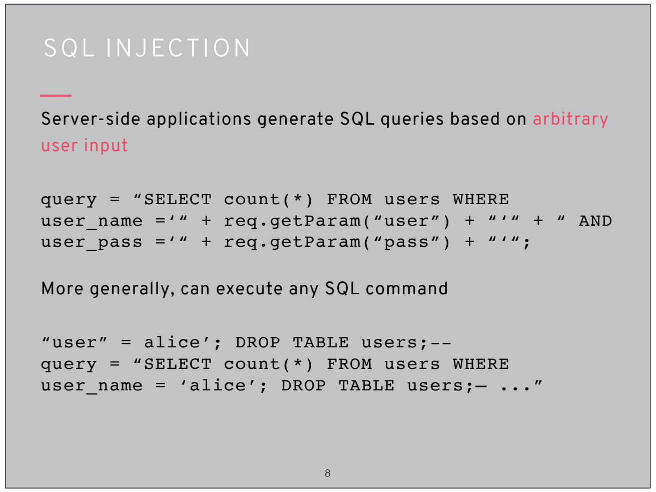
SQL INJECTION

Server-side applications generate SQL queries based on arbitrary user input

```
query = “SELECT count(*) FROM users WHERE
user_name =‘“ + req.getParam(“user”) + “‘“ + “ AND
user_pass =‘“ + req.getParam(“pass”) + “‘“;

“user” = alice’ ;--
query = “SELECT count(*) FROM users WHERE
user_name = ‘alice’ ;-- ...”

“user” = alice
“pass” = foo’ OR 1=1 ;--
query = “SELECT count(*) FROM users WHERE
user_name = ‘alice’ AND user_pass = ‘foo’ OR 1=1;”
```



SQL INJECTION

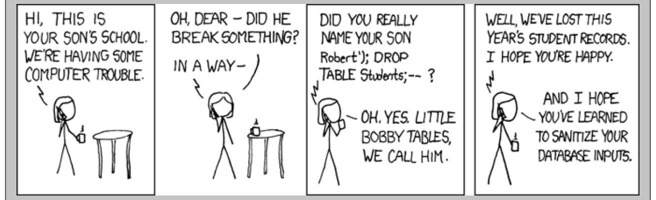
Server-side applications generate SQL queries based on arbitrary user input

```
query = “SELECT count(*) FROM users WHERE
user_name =‘“ + req.getParam(“user”) + “‘“ + “ AND
user_pass =‘“ + req.getParam(“pass”) + “‘“;
```

More generally, can execute any SQL command

```
“user” = alice’; DROP TABLE users;--
query = “SELECT count(*) FROM users WHERE
user_name = ‘alice’; DROP TABLE users;– ...”
```

LITTLE BOBBY TABLES


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH, YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

SQL INJECTION MITIGATIONS

Server-side applications generate SQL queries based on arbitrary user input

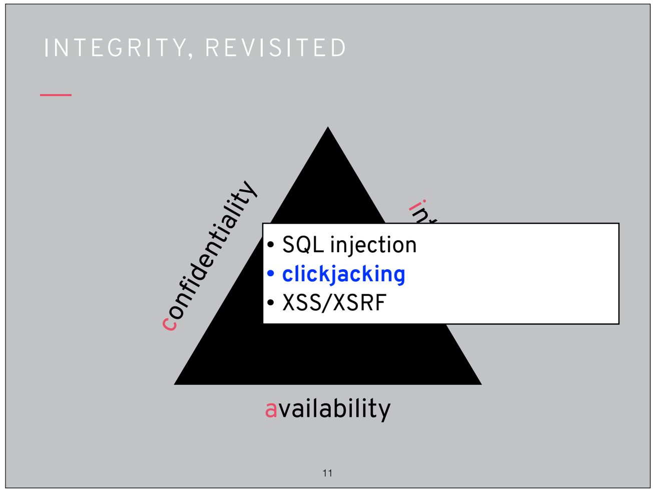
Solution? Don't accept arbitrary user input!

Parameterised queries: pre-compiled queries that separate commands from input

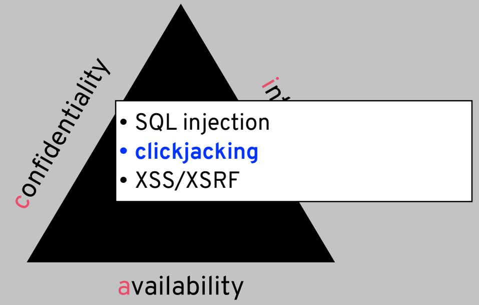
Input sanitisation: make sure only safe input is accepted
What is unsafe?

- Single quote? Dashes? But these could be legitimate

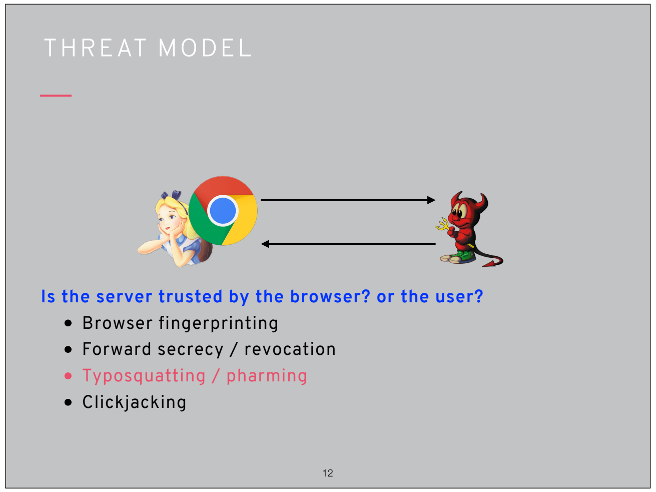
Sanitise on the client or the server?
Use proper escaping/encoding?



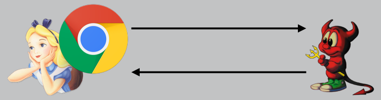
INTEGRITY, REVISITED



- SQL injection
- **clickjacking**
- XSS/XSRF



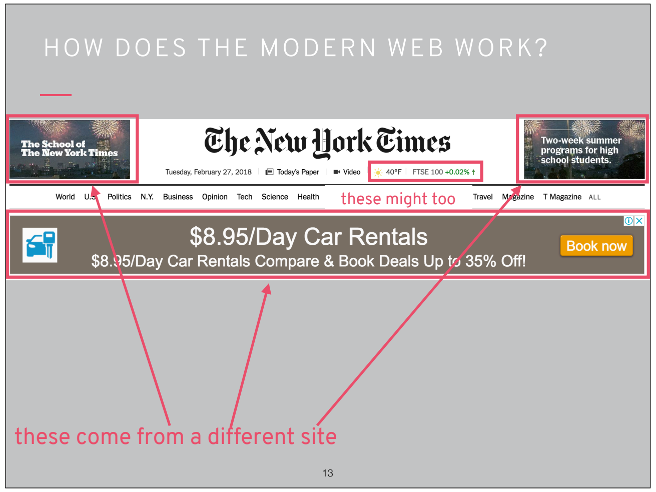
THREAT MODEL



Is the server trusted by the browser? or the user?

- Browser fingerprinting
- Forward secrecy / revocation
- Typosquatting / pharming
- Clickjacking

We won’t go over pharming again but this is also an attack on integrity



HOW DOES THE MODERN WEB WORK?

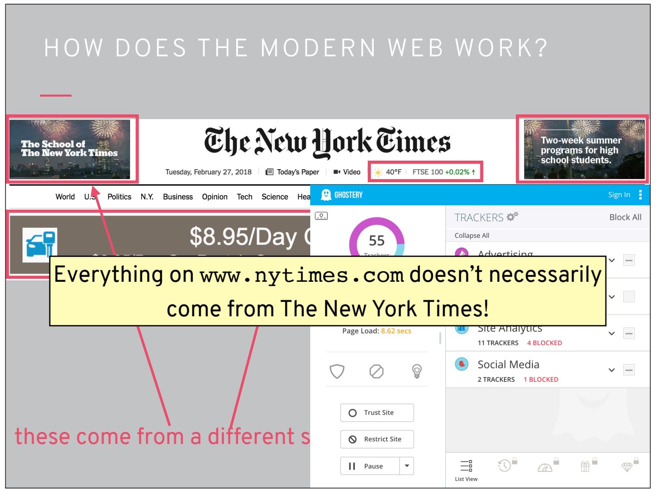
The School of The New York Times
The New York Times
Tuesday, February 27, 2018
Today's Paper
Video
40°F
FTSE 100 +0.02%
Two-week summer programs for high school students.
World
U.S.
Politics
N.Y.
Business
Opinion
Tech
Science
Health
Travel
Magazine
T Magazine
ALL
$8.95/Day Car Rentals
$8.95/Day Car Rentals Compare & Book Deals Up to 35% Off!
Book now

these might too

these come from a different site

13

The content on a big modern website comes from many different sources, not just from the host itself



HOW DOES THE MODERN WEB WORK?

Everything on `www.nytimes.com` doesn't necessarily come from The New York Times!

these come from a different s

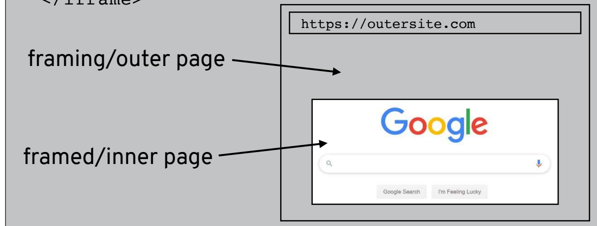
This means that there can be a lot of trackers, but also that in general scripts are running that you don't know about

IFRAMES

Content from one site is embedded into another using **iframes**

Example:

```
<iframe src="https://www.google.com">
</iframe>
```



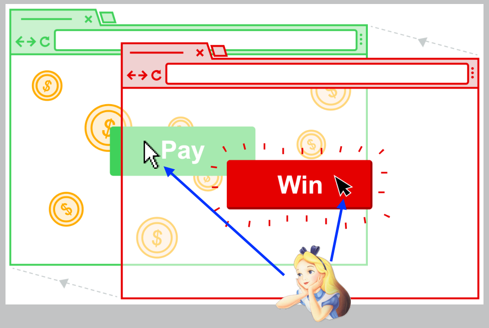
IFRAMES

Content from one site is embedded into another using **iframes**

Outer page can set the width and height of the frame

Only inner page can draw within its frame

CLICKJACKING



17

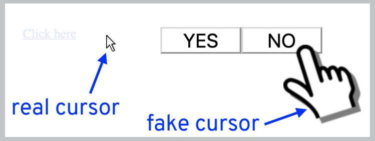
Clickjacking relies on overlay frames that make a user think they're clicking on one thing but really it's another (think how clicking on almost anything on a streaming site causes an ad to pop up)

CLICKJACKING

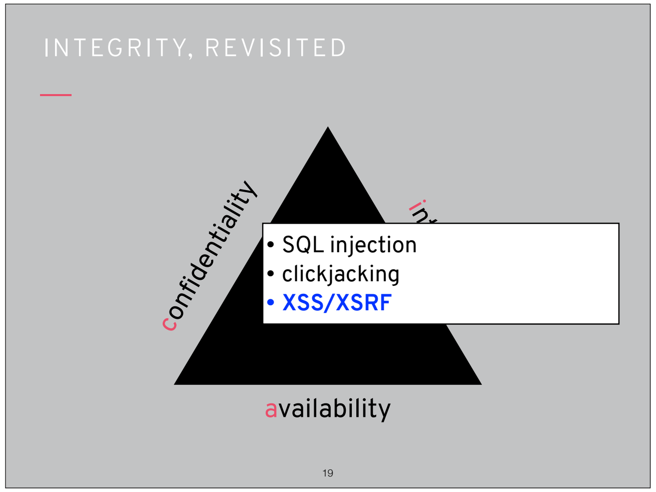
Attacker site frames good site and covers part of it to look different / create unintended interactions

Likejacking is when Facebook users are tricked into clicking the 'like' button

This is often achieved using cursorjacking



18



INTEGRITY, REVISITED

- SQL injection
- clickjacking
- **XSS/XSRF**