Stanford Applied Cyber
Intro Security Workshop

Aditya Saligrama

Security and its relevance to Stanford students

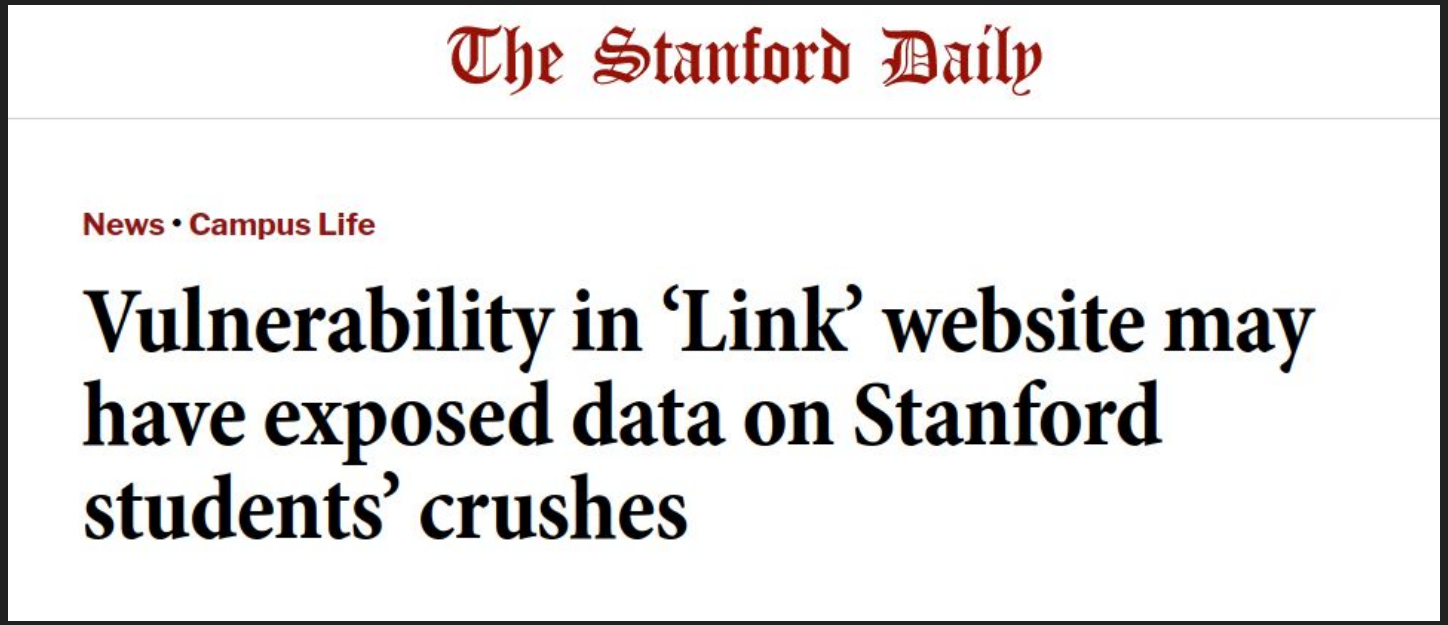
Case study: Stanford Link (2020)

- *Match with your crush if they like you back*
- *Website keeps you anonymous if they don't*

Case study: Stanford Link (2020)

- *Match with your crush if they like you back*
- *Website keeps you anonymous if they don't*
- ***What could go wrong?***

Case study: Stanford Link (2020)

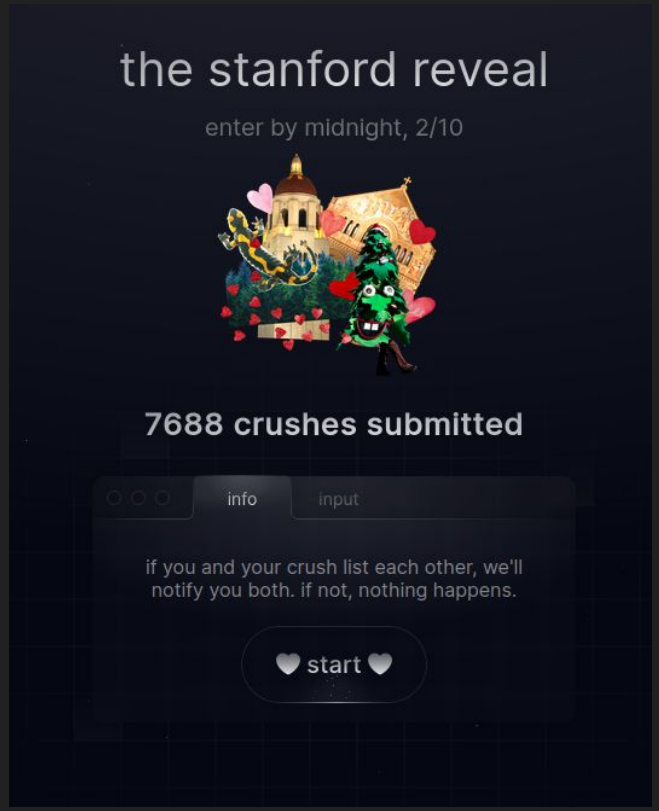


The Stanford Daily

News • Campus Life

Vulnerability in 'Link' website may have exposed data on Stanford students' crushes

What’s old is new again: Stanford Reveal (2023)

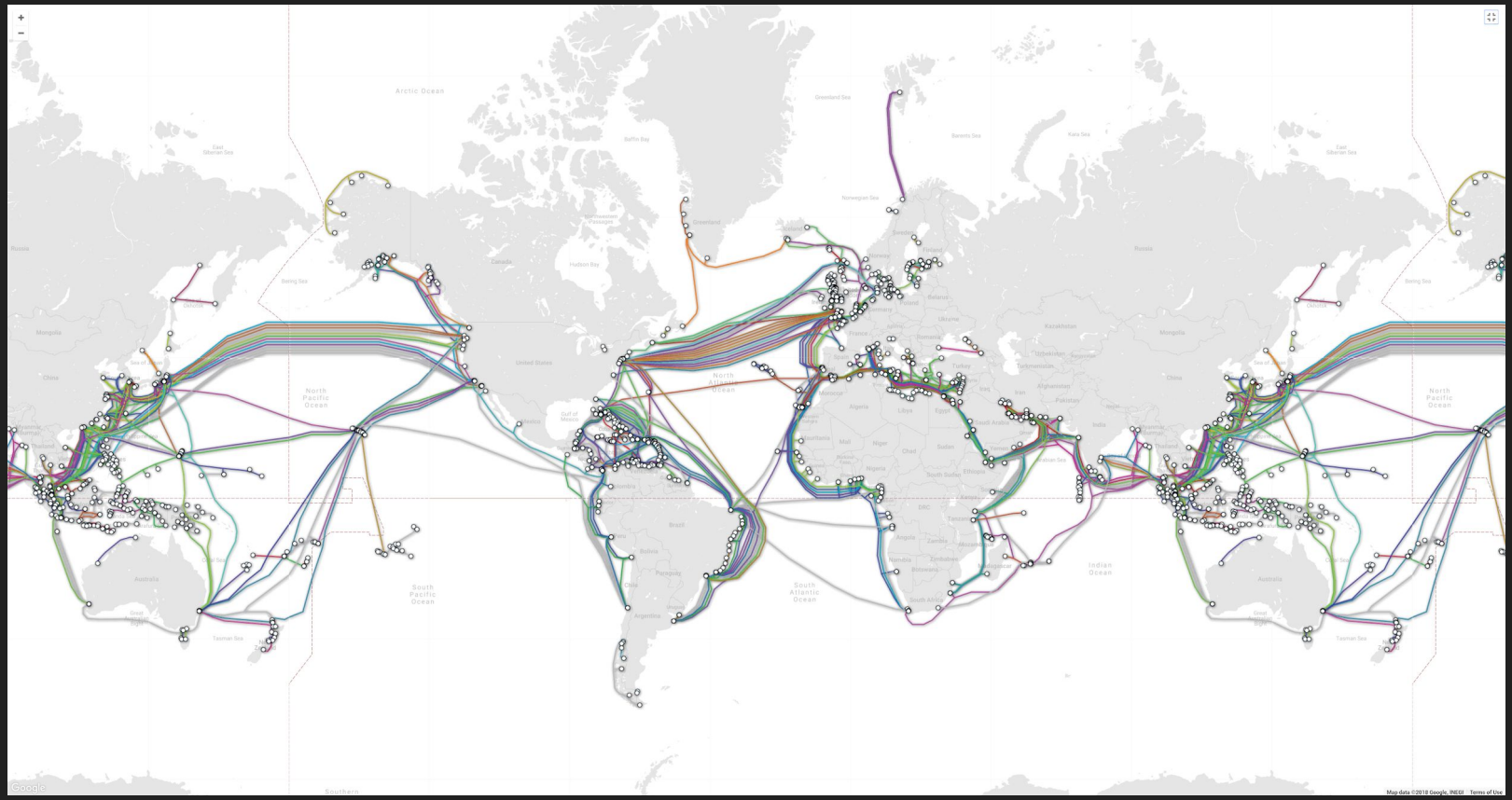



```
44    {
45      "submittingUserFullName": "Aditya Saligrama",
46      "user": "4yz2FPyYDgND8KhtVOQLCeeGsaq2",
47      "submittingUserEmail": "akps@stanford.edu",
48      "fullNames": []
49    },
```

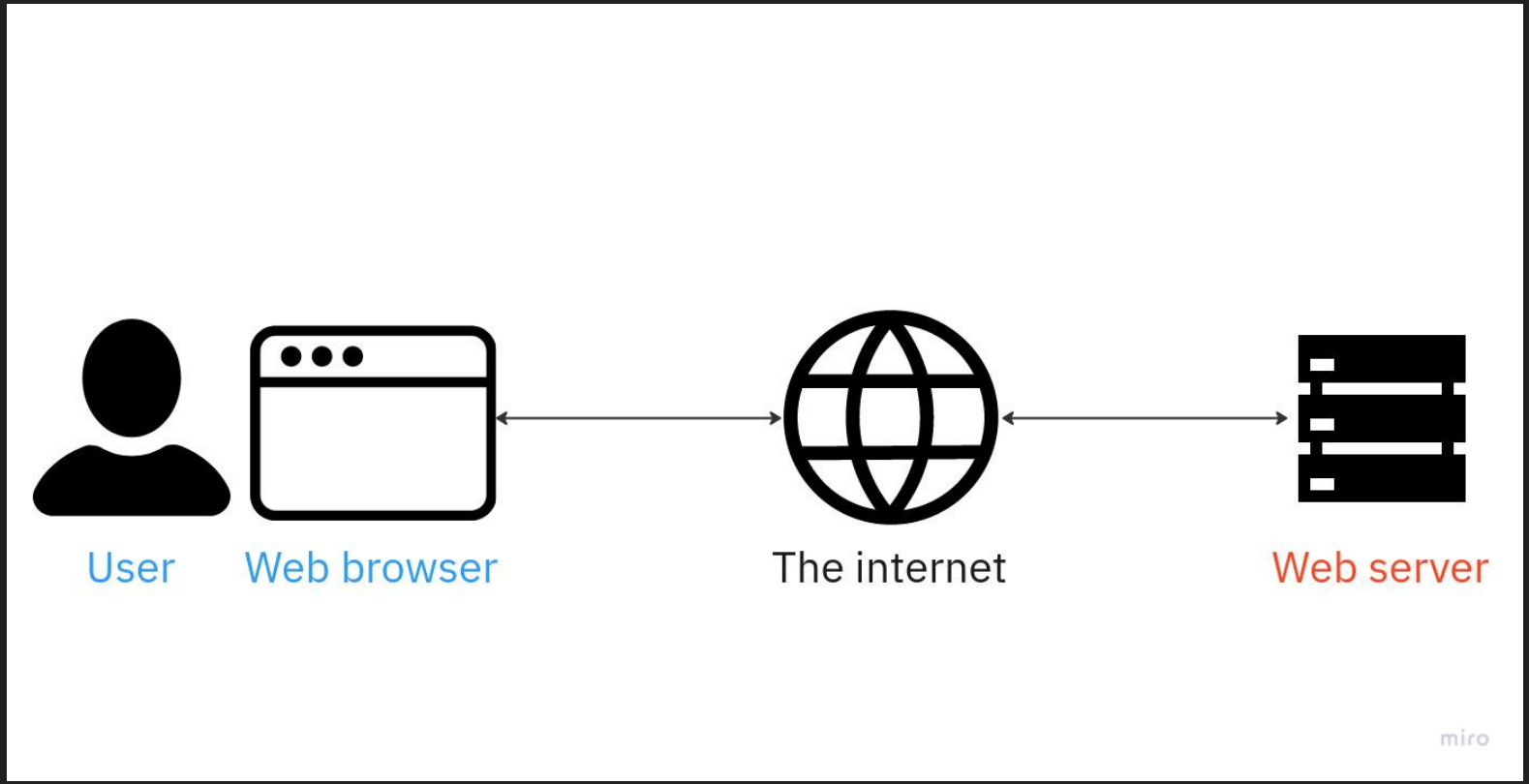
```
231   {
232     "submittingUserEmail": "mccain@stanford.edu",
233     "submittingUserFullName": "Robert Miles Redd McCain",
234     "user": "N3Q9CkeKeJfKQzOhqt7qFbpanat1",
235     "fullNames": [
236       "isabelle levent"
237     ]
238   },
```

The fastest web crash course ever

How does the Internet work?



Our Internet Abstraction



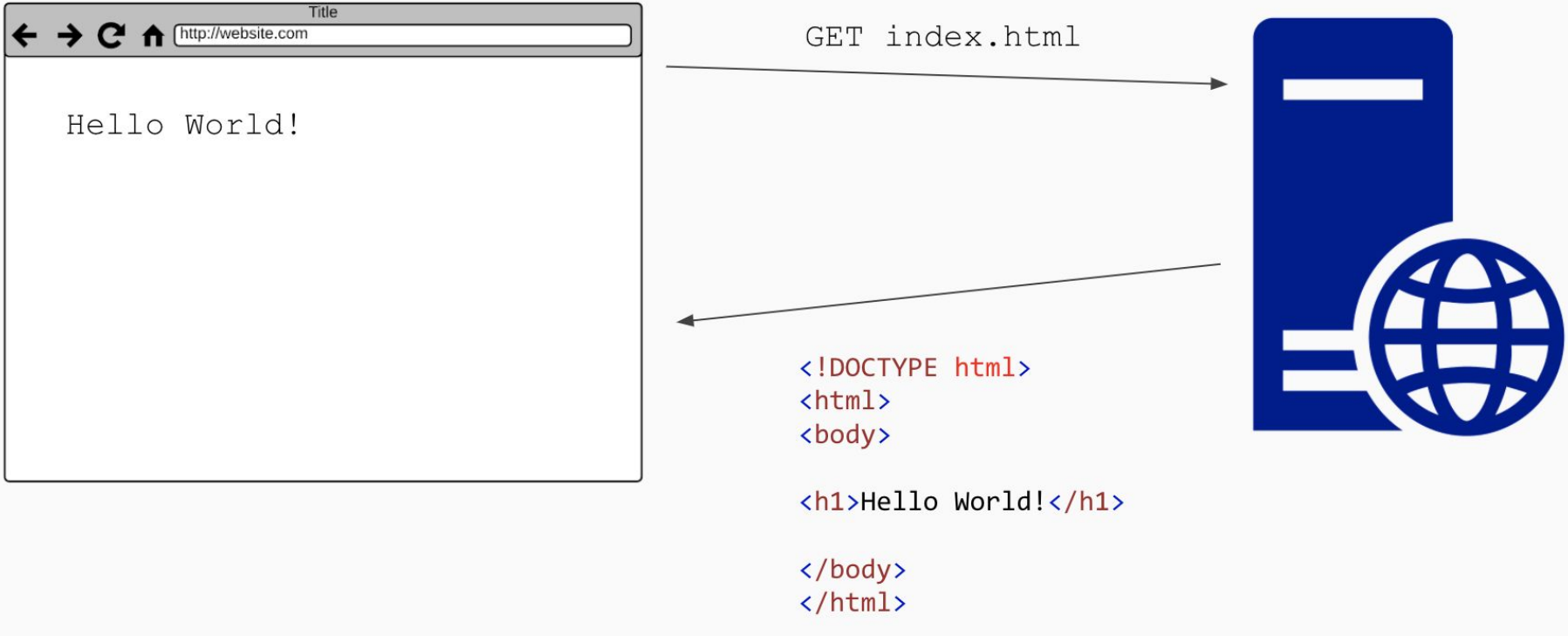
What language does the web speak?

How do we communicate with a web server?

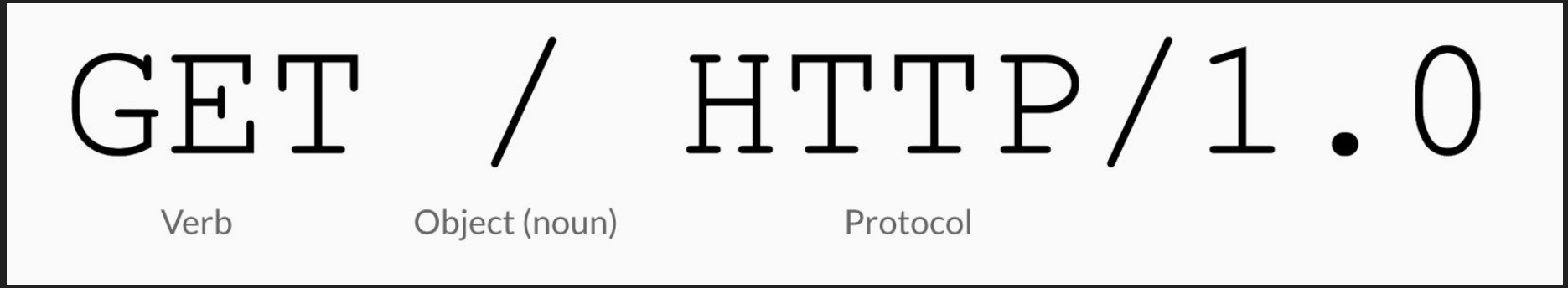
HTTP

Hypertext Transport Protocol

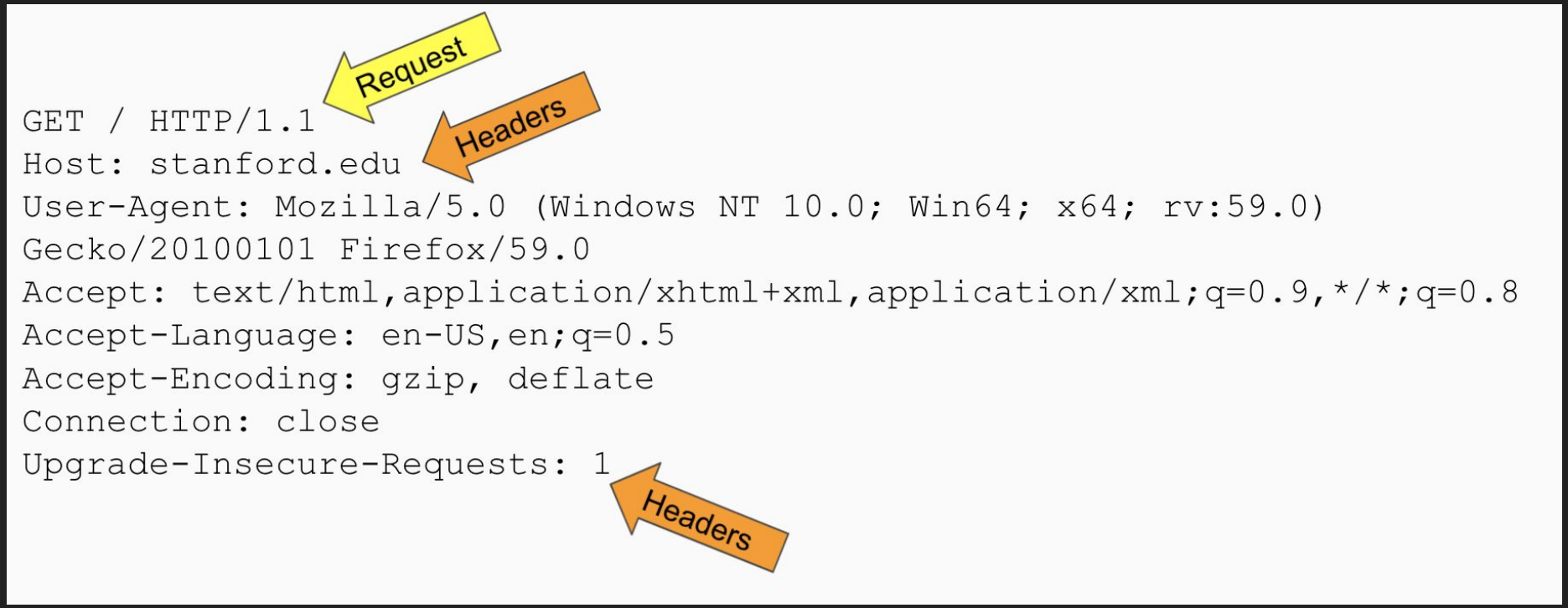
HTTP: the missing language of the web



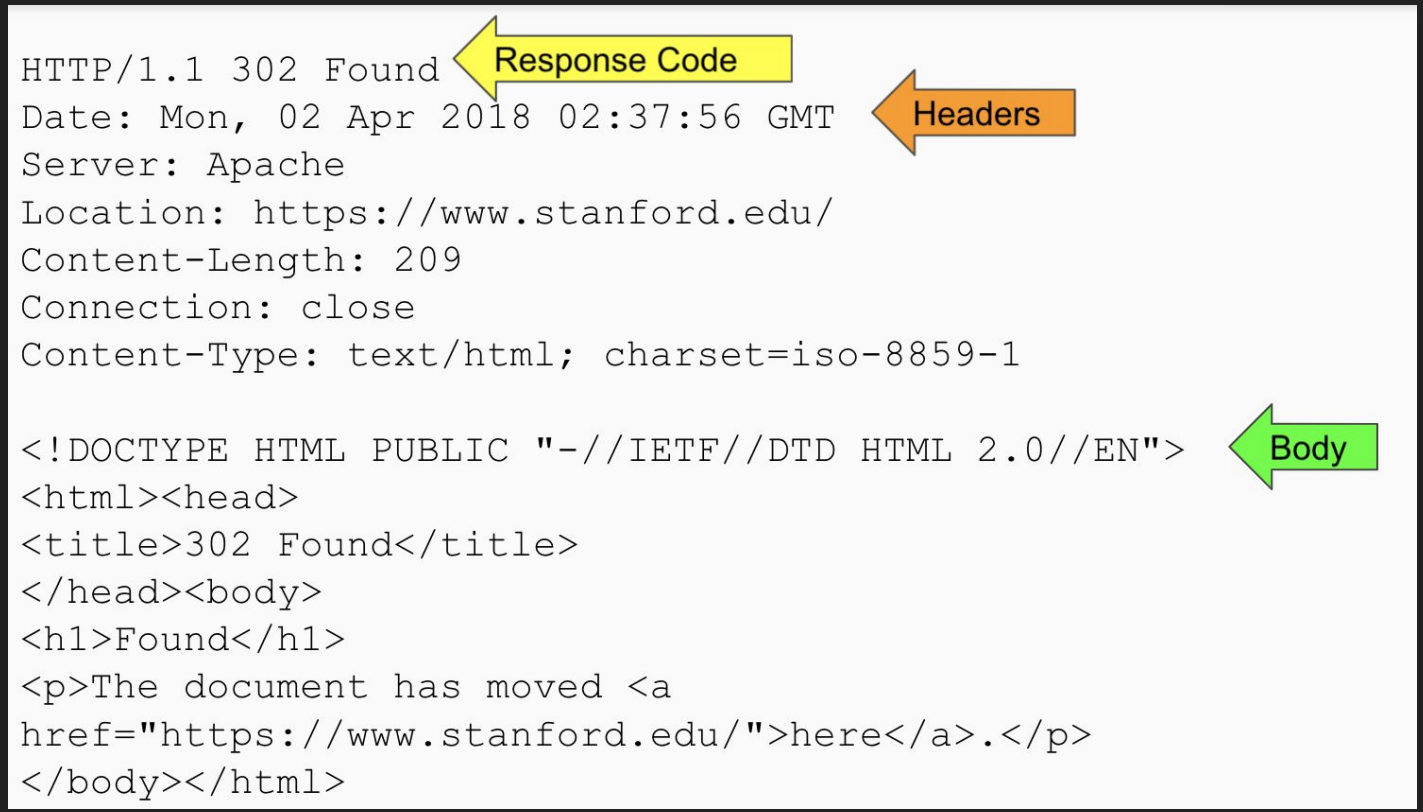
HTTP protocol



HTTP Requests



HTTP Responses

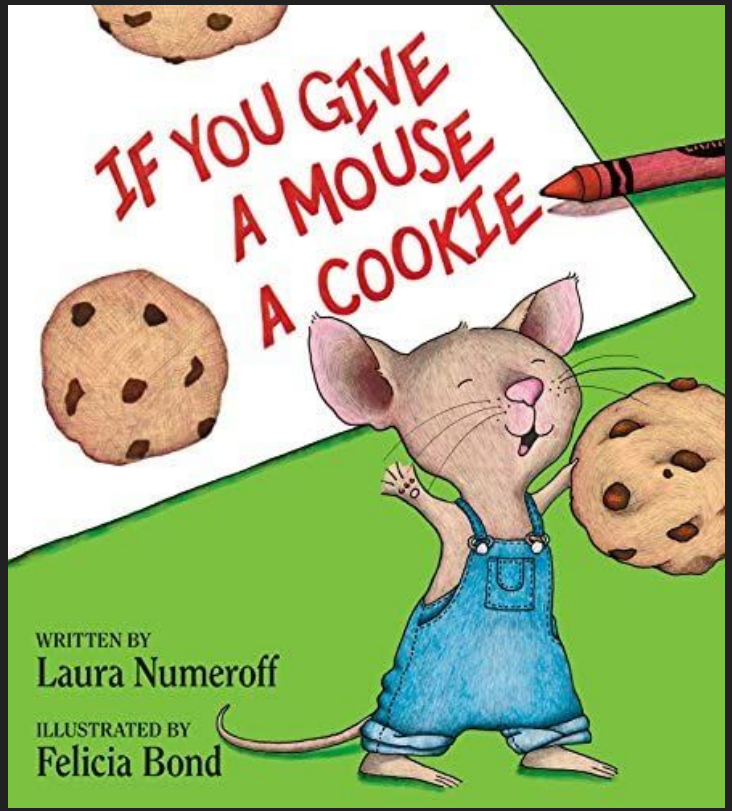


HTTP Requests: GET and POST

- **GET**: Requests a specified resource
  - Should only retrieve data, without changing server state

- **POST**: Submits data to the specified resource
  - Often causes changes in state or side effects on the server

Session Handling: *How does a website remember?*

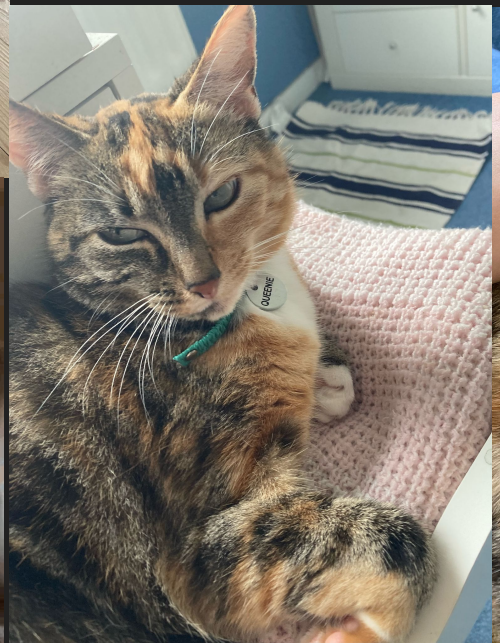


- ***Cookies!***
- Cookies enable web servers to store stateful information in your browser
- Authentication cookies are used to authenticate that a user is logged in, and with which account
  - On login: `Set-Cookie: session=session-id`
  - Future requests: `Cookie: session=session-id`

Common insecure design patterns

CatShare

https://catshare.saligrama.io



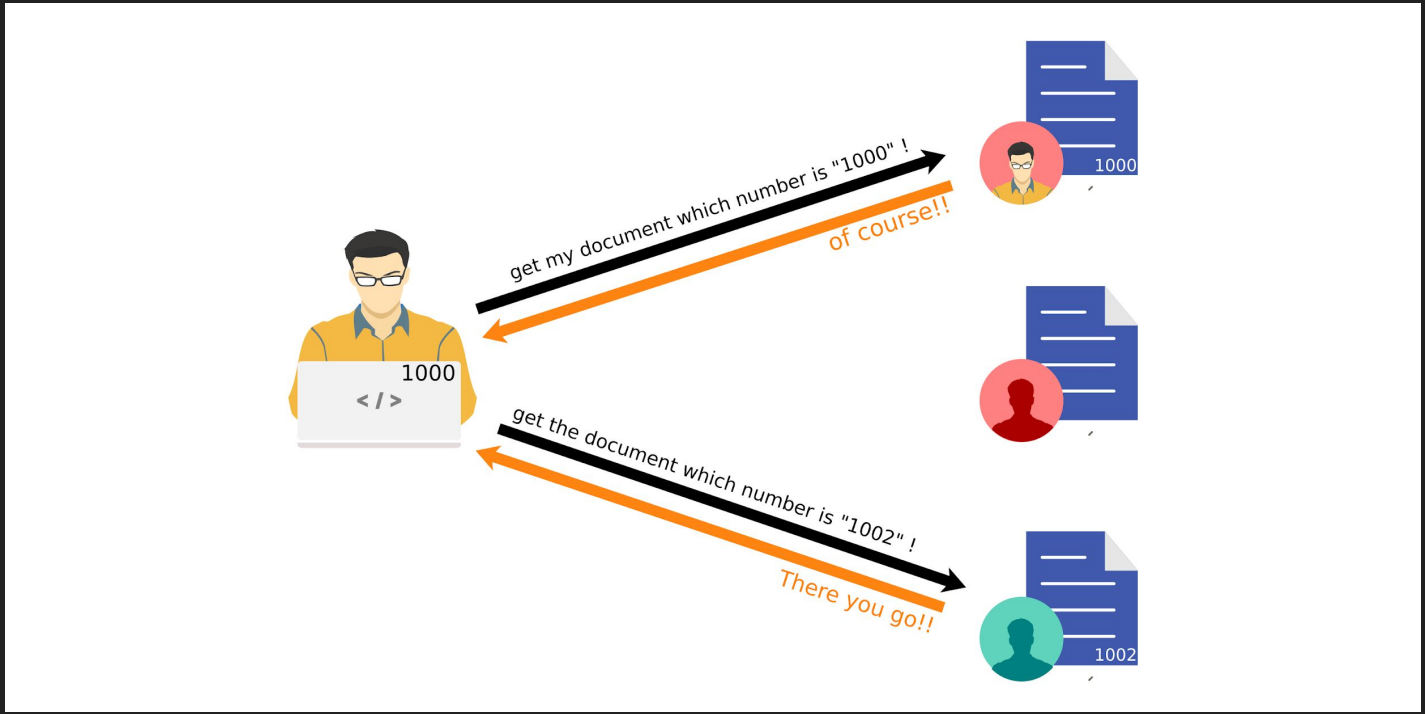
We're a real startup!

Vulnerabilities

- Insecure Direct Object Reference (IDOR)
- Cross Site Scripting (XSS)
- Improper Session Handling

Insecure Direct Object Reference (IDOR)

Or: asking the server for the resources you want



IDOR case study I: Parler

- IDOR vulnerability leads to leakage of 70TB of user data

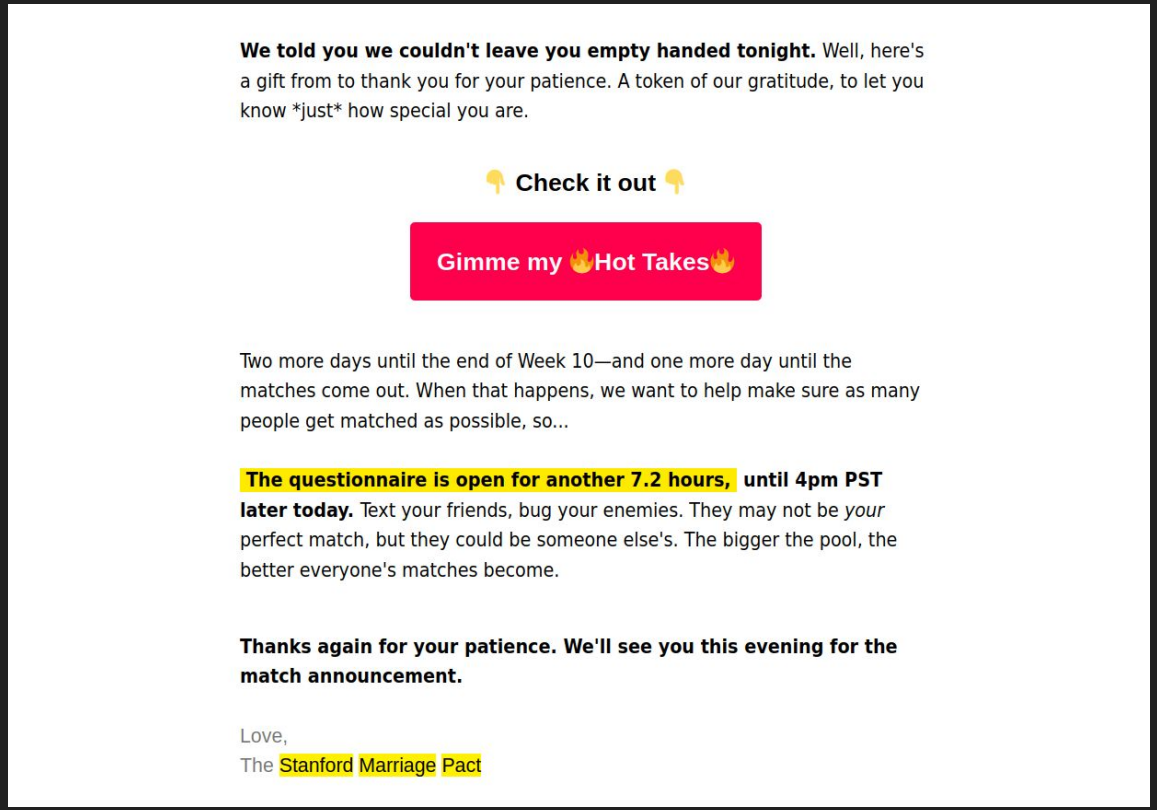
- Why?
  - Poor engineering
  - Lack of testing

Source: https://cybernews.com/news/70tb-of-parler-users-messages-videos-and-posts-leaked-by-security-researchers/

TRY IT!

- The CatShare team has a website https://catshare.saligrama.io/ that **stores personal information**
- There's an endpoint https://catshare.saligrama.io/user to access this info
  - e.g. https://catshare.saligrama.io/user?id=test
- CatShare claims this is secure and only accessible to admins
- **Show us otherwise**

IDOR case study II: Stanford Marriage Pact (2020)



We told you we couldn't leave you empty handed tonight. Well, here's a gift from to thank you for your patience. A token of our gratitude, to let you know *just* how special you are.

👇 **Check it out** 👇

Gimme my 🔥Hot Takes🔥

Two more days until the end of Week 10—and one more day until the matches come out. When that happens, we want to help make sure as many people get matched as possible, so...

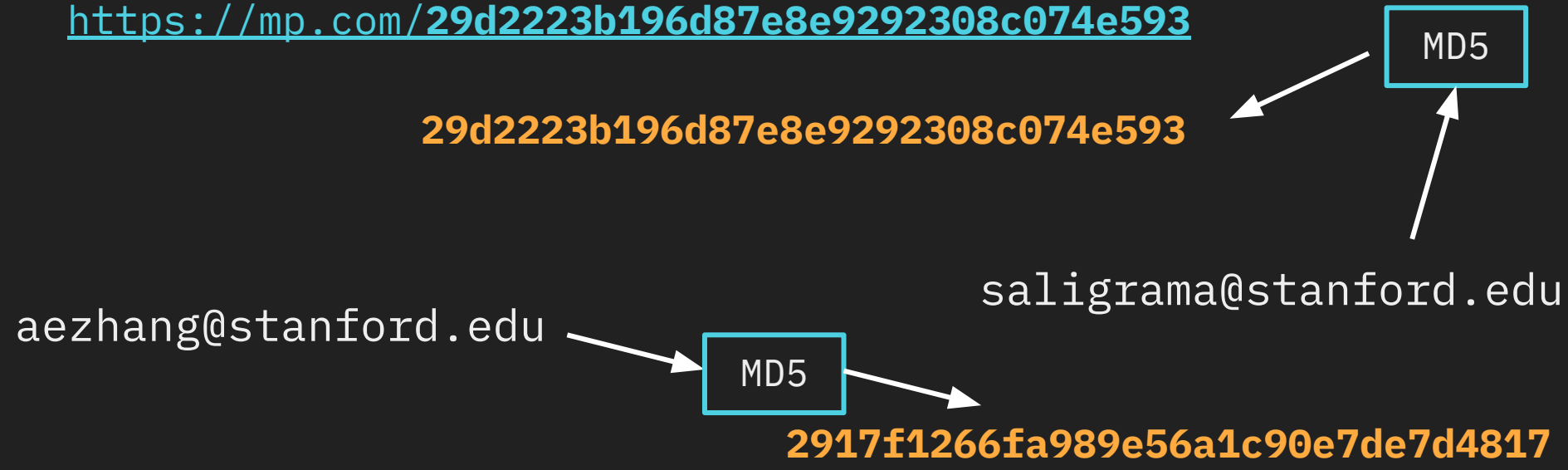
The questionnaire is open for another 7.2 hours, until 4pm PST later today. Text your friends, bug your enemies. They may not be *your* perfect match, but they could be someone else's. The bigger the pool, the better everyone's matches become.

Thanks again for your patience. We'll see you this evening for the match announcement.

Love,
The Stanford Marriage Pact

IDOR case study II: Stanford Marriage Pact (2020)

https://mp.com/**29d2223b196d87e8e9292308c074e593**



https://mp.com/2917f1266fa989e56a1c90e7de7d4817

Avoiding IDOR

- Ensure that a user is allowed to access a resource before returning it

- If not possible (e.g. cloud storage buckets), then make resource URIs random and unpredictable. Avoid:
  - Automatically incrementing resource IDs
  - Hashing a guessable property such as usernames, phone numbers, or emails

- Instead: use random identifiers such as UUIDs

Cross Site Scripting (XSS)

- XSS attacks enable attackers to hijack your website to **run JavaScript code** on other users' browsers

- They occur when **user input is not properly sanitized and displayed**, allowing it to execute as code

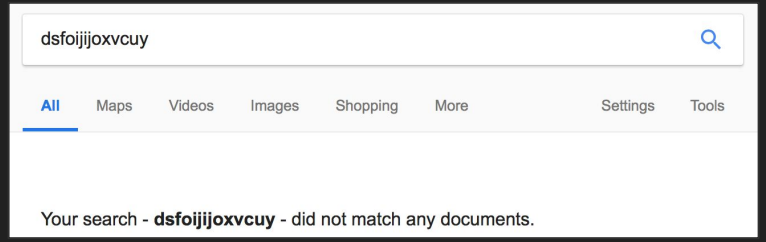
Cross-Site Scripting (XSS)

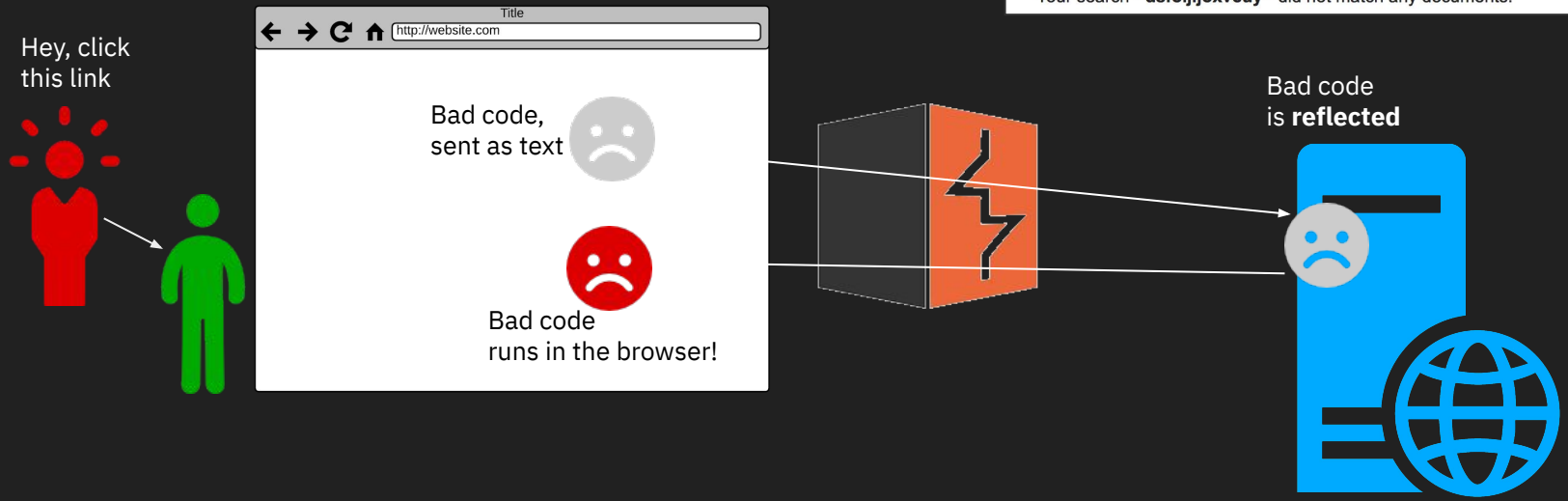
GET /myfeed

```
<!DOCTYPE html>
<html>
<body>

<b>Teddy says:</b>
<script>
     alert("You got hacked!");
</script>

</body>
</html>
```

www.yourwebsite.com/law/<?StealAllTheData.js>/supersecretdata

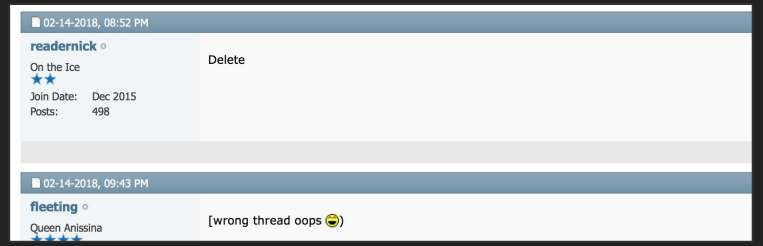
Reflected XSS

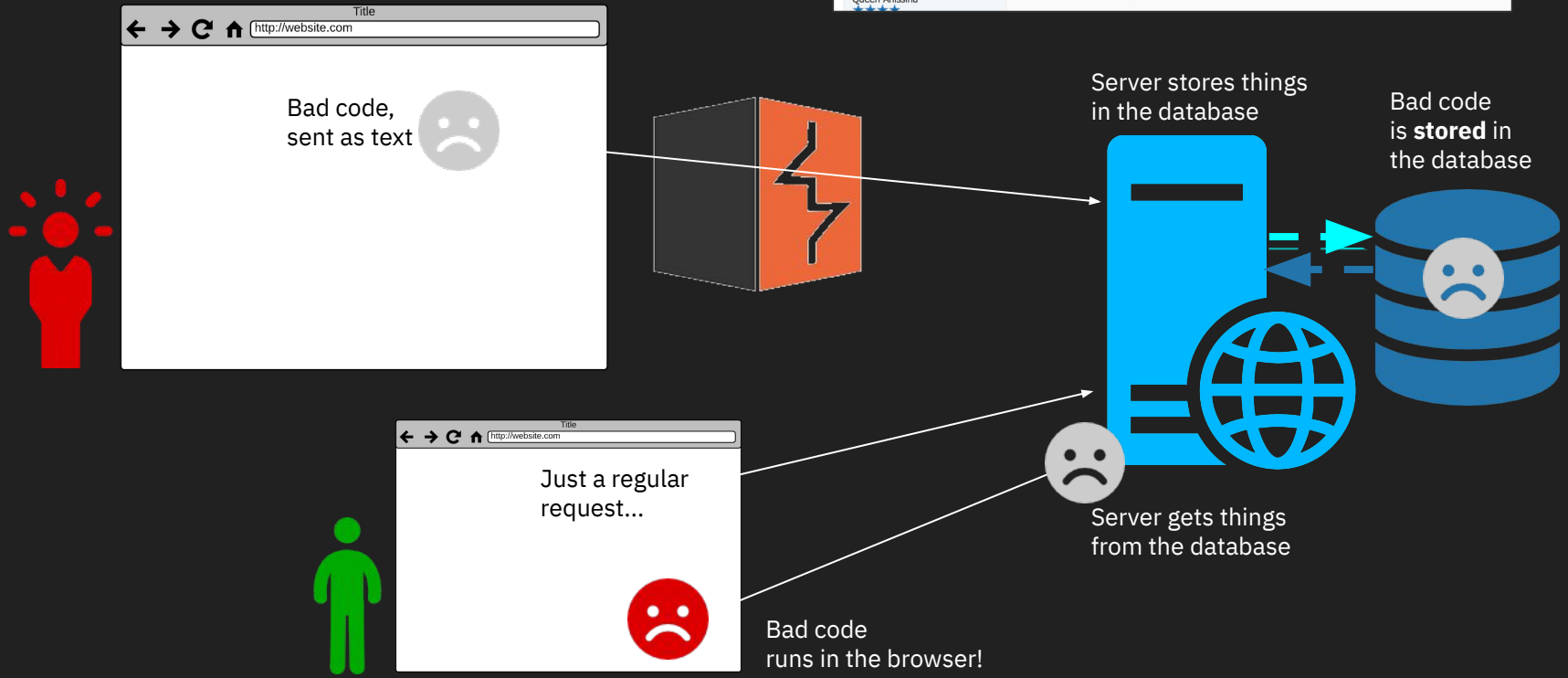




https://vulnerable.website/search?query=<script>alert("pwned")</script>

Stored XSS


02-14-2018, 08:52 PM
readernick
On the Ice
Join Date: Dec 2015
Posts: 498
Delete
02-14-2018, 09:43 PM
fleeting
Queen Anissina
[wrong thread oops 😎]


Title
http://website.com
Bad code,
sent as text
Server stores things
in the database
Bad code
is **stored** in
the database
Title
http://website.com
Just a regular
request...
Server gets things
from the database
Bad code
runs in the browser!

TRY IT!

- After our last data breach, we at CatShare want to make our customers feel like we care about them

- We added an endpoint https://catshare.saligrama.io/hello that takes a user's name and greets them kindly. Ya' know, to show we care
  - e.g. https://catshare.saligrama.io/hello?name=User1

- We think this is harmless and will only build customer trust. **Show us our mistake.**

Improper session handling

Cookie itself is insecure

- Can modify cookie to access another's account
  - e.g. become admin

Cookie not checked for authorization

- Use your own account to
  - Impersonate someone else
  - Escalate privileges to admin

Consequences are IDOR-like, even when resource IDs are randomized

TRY IT!

- CatShare added an admin view to https://catshare.saligrama.io/login for admins to view user data

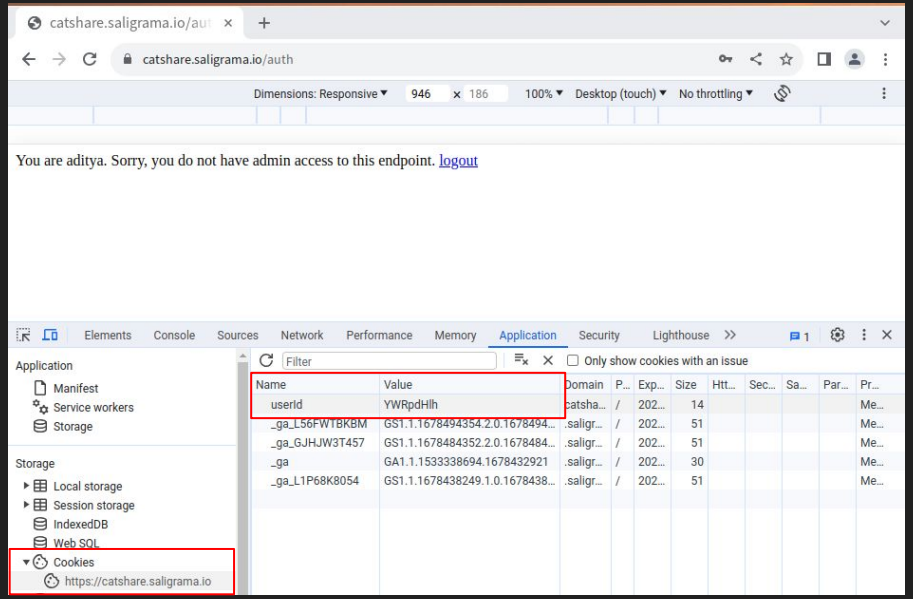
- Log in using `cooper:cooper`

- Can you become `admin` and view the user data?

TRY IT!

TOOLS/REFERENCE

- Cookie is in Base64 format
  - Transforms data into a mix of letters and numbers.
  - Doesn't actually secure or encrypt data; it's just a different way to show it.
  - Use https://kk.lol to encode/decode
- Your browser's Developer Tools
  - Accessible from Inspect Element

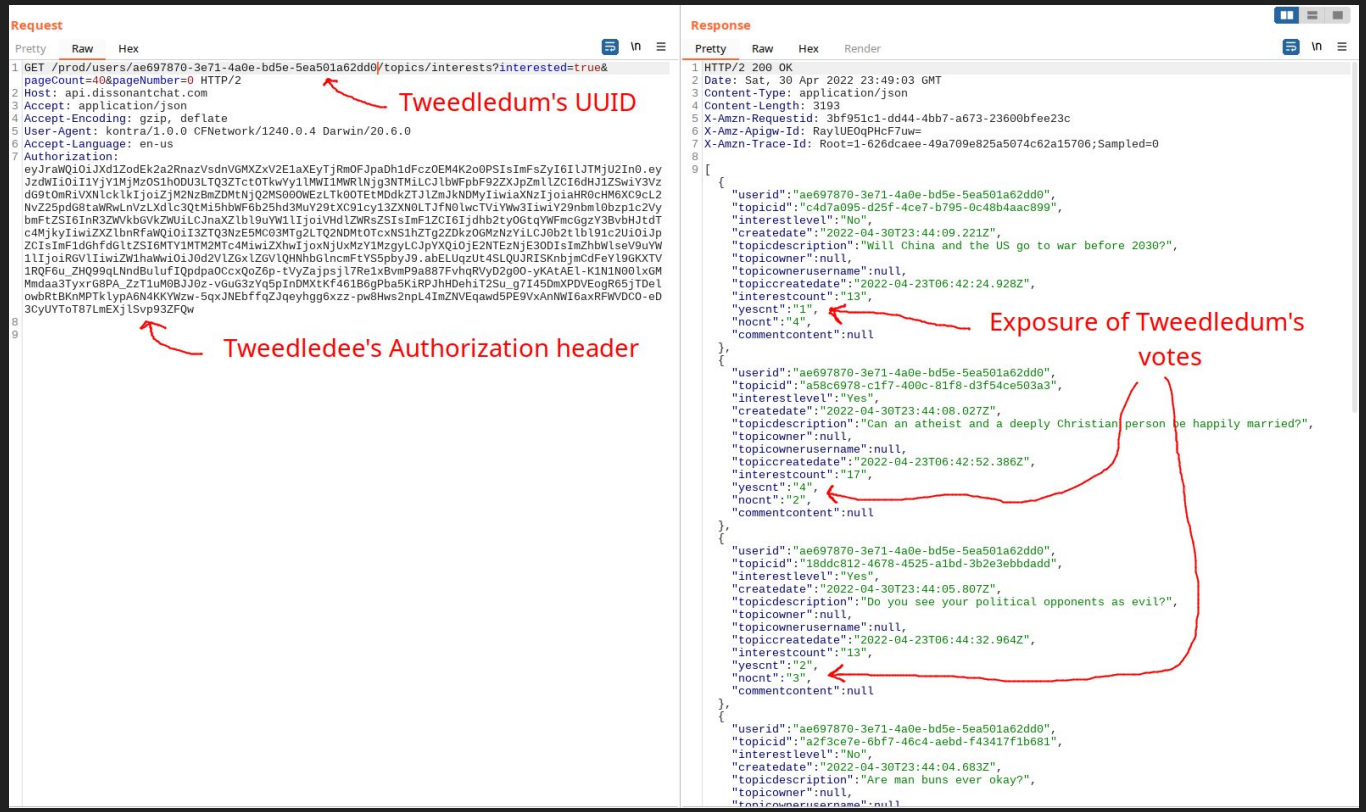


What to look for is in red (logged in as `aditya` here)

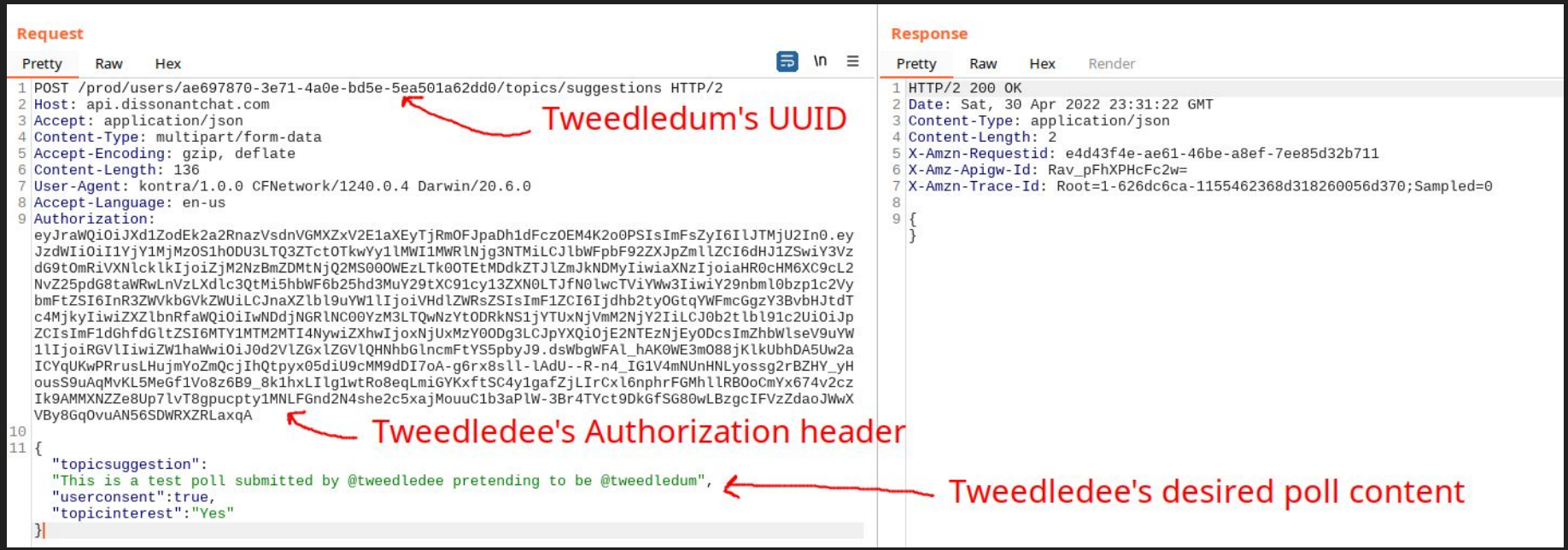
- https://catshare.saligrama.io/login
  - Login with `cooper:cooper`

Session handling case study: Kontra (2022)

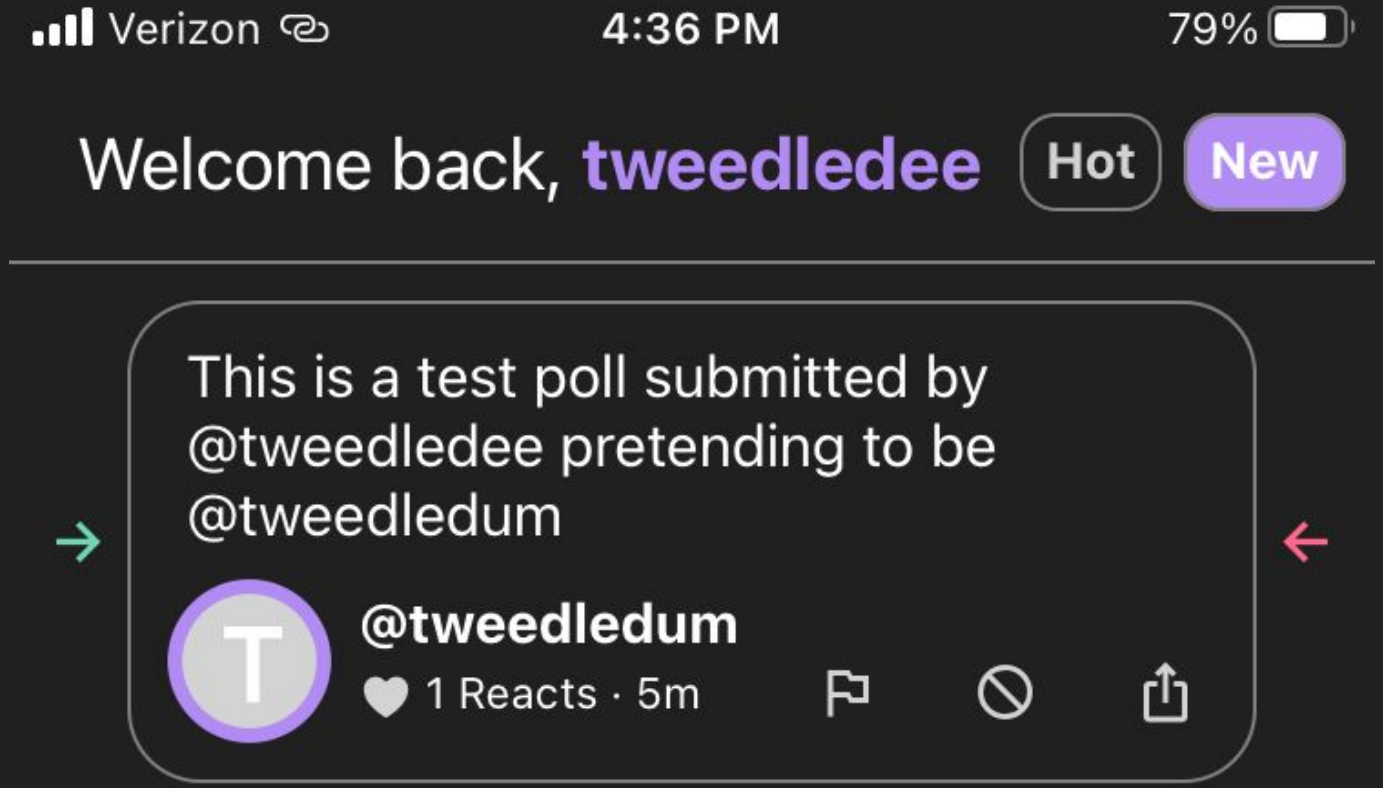
Session handling case study: Kontra (2022)



Session handling case study: Kontra (2022)



Session handling case study: Kontra (2022)



Avoiding improper session handling

Before taking a sensitive action:

Check the user is who they say they are

And that they are allowed to perform the action

Mitigating risk as a startup

Nothing is 100% secure

It happens to the best of us

Aditya's Blog

Thoughts, guides and fun from a security/systems enthusiast @ Stanford

Flipping the script: when a hacking class gets hacked

October 12, 2022 1316 words No tag

This morning, an EternalBlue-vulnerable machine used for testing for Stanford's Hack Lab course accidentally given a public IP address on Google Cloud was unsurprisingly pwned and used to launch further EternalBlue scanning against other public web hosts.

This blog post describes our course's infrastructure setup (including why we had that testing box in the first place), how we discovered and remediated the incident, and how we used the incident as a way to teach students about incident response and public disclosure.

Let the community help you

A vulnerability disclosure policy is intended to give ethical hackers clear guidelines for submitting potentially unknown and harmful security vulnerabilities to organizations.

Vulnerability Disclosure Policy Resources

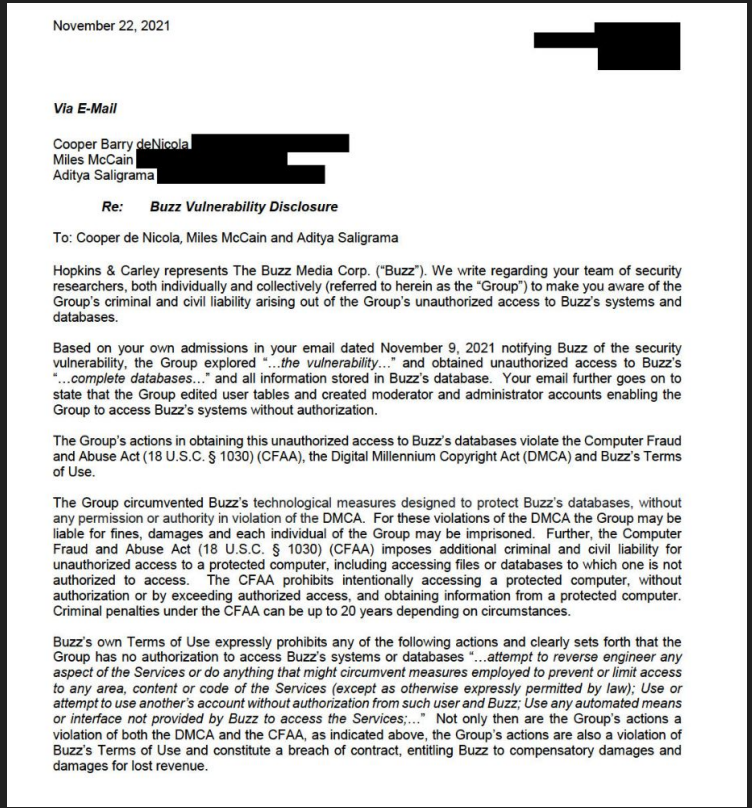
DHS Template: https://cyber.dhs.gov/bod/20-01/vdp-template/

DoJ Framework: https://www.justice.gov/criminal-ccips/page/file/983996/download

HackerOne: https://www.hackerone.com/blog/What-Vulnerability-Disclosure-Policy-and-Why-You-Need-One

Example Safe Harbor: https://github.com/cybertransparency/vdp-terms

Please don’t do this



November 22, 2021

[redacted]

Via E-Mail

Cooper Barry deNicola [redacted]
Miles McCain
Aditya Saligrama

Re: Buzz Vulnerability Disclosure

To: Cooper de Nicola, Miles McCain and Aditya Saligrama

Hopkins & Carley represents The Buzz Media Corp. (“Buzz”). We write regarding your team of security researchers, both individually and collectively (referred to herein as the “Group”) to make you aware of the Group’s criminal and civil liability arising out of the Group’s unauthorized access to Buzz’s systems and databases.

Based on your own admissions in your email dated November 9, 2021 notifying Buzz of the security vulnerability, the Group explored “*…the vulnerability…*” and obtained unauthorized access to Buzz’s “*…complete databases…*” and all information stored in Buzz’s database. Your email further goes on to state that the Group edited user tables and created moderator and administrator accounts enabling the Group to access Buzz’s systems without authorization.

The Group’s actions in obtaining this unauthorized access to Buzz’s databases violate the Computer Fraud and Abuse Act (18 U.S.C. § 1030) (CFAA), the Digital Millennium Copyright Act (DMCA) and Buzz’s Terms of Use.

The Group circumvented Buzz’s technological measures designed to protect Buzz’s databases, without any permission or authority in violation of the DMCA. For these violations of the DMCA the Group may be liable for fines, damages and each individual of the Group may be imprisoned. Further, the Computer Fraud and Abuse Act (18 U.S.C. § 1030) (CFAA) imposes additional criminal and civil liability for unauthorized access to a protected computer, including accessing files or databases to which one is not authorized to access. The CFAA prohibits intentionally accessing a protected computer, without authorization or by exceeding authorized access, and obtaining information from a protected computer. Criminal penalties under the CFAA can be up to 20 years depending on circumstances.

Buzz’s own Terms of Use expressly prohibits any of the following actions and clearly sets forth that the Group has no authorization to access Buzz’s systems or databases “*…attempt to reverse engineer any aspect of the Services or do anything that might circumvent measures employed to prevent or limit access to any area, content or code of the Services (except as otherwise expressly permitted by law); Use or attempt to use another’s account without authorization from such user and Buzz; Use any automated means or interface not provided by Buzz to access the Services;…*” Not only then are the Group’s actions a violation of both the DMCA and the CFAA, as indicated above, the Group’s actions are also a violation of Buzz’s Terms of Use and constitute a breach of contract, entitling Buzz to compensatory damages and damages for lost revenue.

More Security at Stanford

Courses

- *INTLPOL 268*: Hack Lab
- *CS 155*: Computer and Network Security
- *CS 152*: Trust and Safety Engineering
- *CS 255*: Cryptography
- *CS 153*: Applied Security at Scale
- *INTLPOL 268D*: Online Open Source Investigation
- *CS (TBD): Cloud Infrastructure and Scalable Application Deployment*

More AC war stories

Opinion | Fizz previously compromised its users' privacy. It may do so again.

Fizz had a large data vulnerability discovered last fall. Their response raises questions about the app today.
(Graphic: JOYCE CHEN/The Stanford Daily)

Aditya's Blog

Thoughts, guides and fun from a security/systems enthusiast @ Stanford

A student's dream: hacking (then fixing) Gradescope's autograder

February 28, 2023 2591 words No tag

Hacking GraphQL for fun ~~and profit~~

Aditya Saligrama

Credits

Source Code for Vulnerable Web App

https://github.com/saligrama/catshare

Other materials

- *Web Crash Course* – Alex Stamos, INTLPOL 268 Hack Lab
- *Web Crash Course, IDOR/XSS/Session Handling Slides, Marriage Pact IDOR Case Study* – Cooper de Nicola
- *Stanford Link, Fizz articles* – The Stanford Daily
- *CatShare* – Cooper de Nicola, Aditya Saligrama, George Hosono