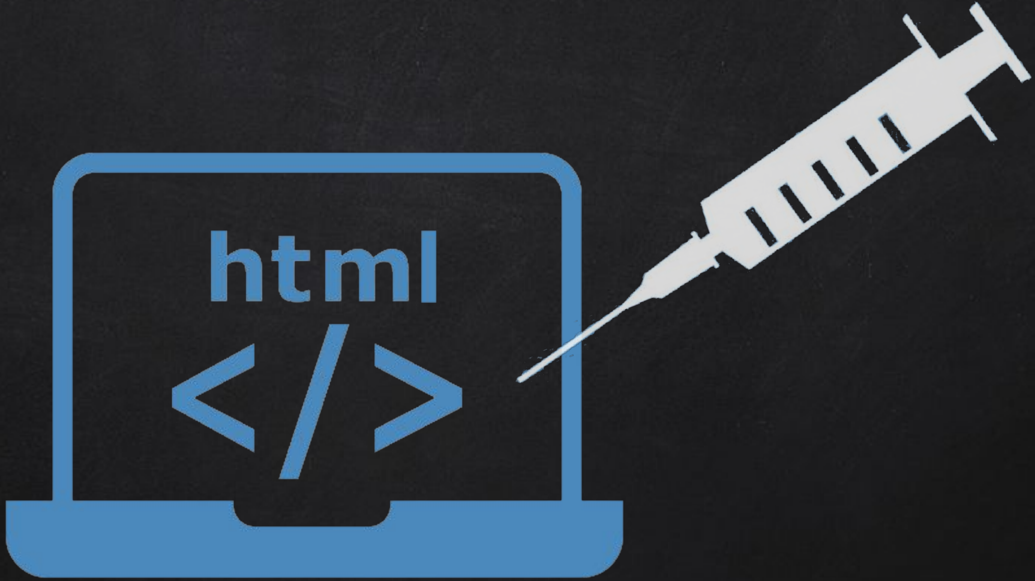
XSS Cross Site Scripting

- Allow an attacker to inject javascript code into the page.
- Code is executed when the page loads.
- Code is executed on the client machine not the server.

Three main types:

1. Reflected XSS
2. Persistent/Stored XSS
3. DOM based XSS

HTML Injection

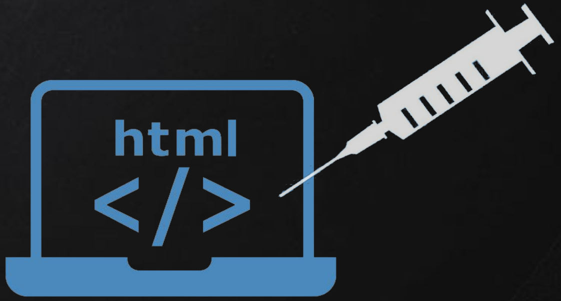


HTML INJECTION

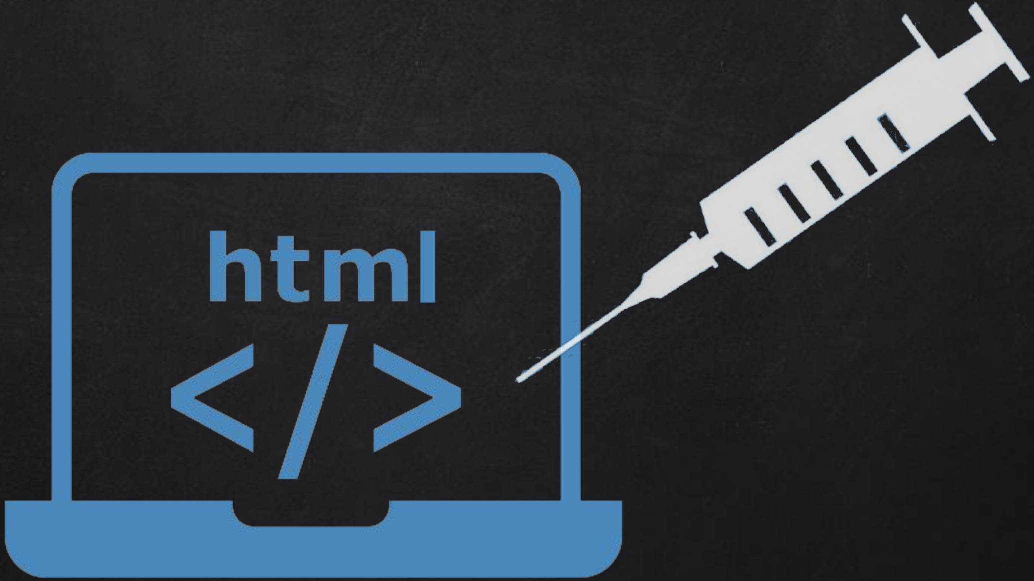
- Allow an attacker to inject HTML code into the page.
- Code is executed when the page loads.
- Code is executed on the client machine not the server.

→ Similar to XSS but simpler.

→ Hints at the existence of an XSS.



XSS Cross Site Scripting



XSS Cross Site Scripting

Discovering XSS

- Try to inject Javasript code into the pages.
- Test text boxes and url parameters on the form
  http://target.com/page.php?something=something

XSS Cross Site Scripting

Reflected XSS

- None persistent, not stored.
- Only work if the target visits a specially crafted URL
- EX
  http://target.com/page.php?something=<script>alert(“XSS”)</script>

XSS Cross Site Scripting

Stored XSS

- Persistent, stored on the page or DB.
- The injected code is executed everytime the page is loaded.

XSS Cross Site Scripting

Dom Based XSS

- Similar to reflected and stored XSS.
- Can be discovered and exploited similarly.
- Main difference is that it occurs entirely on the client side.
- Payload is never sent to the server.
  → No logs, no filters, no server side protection

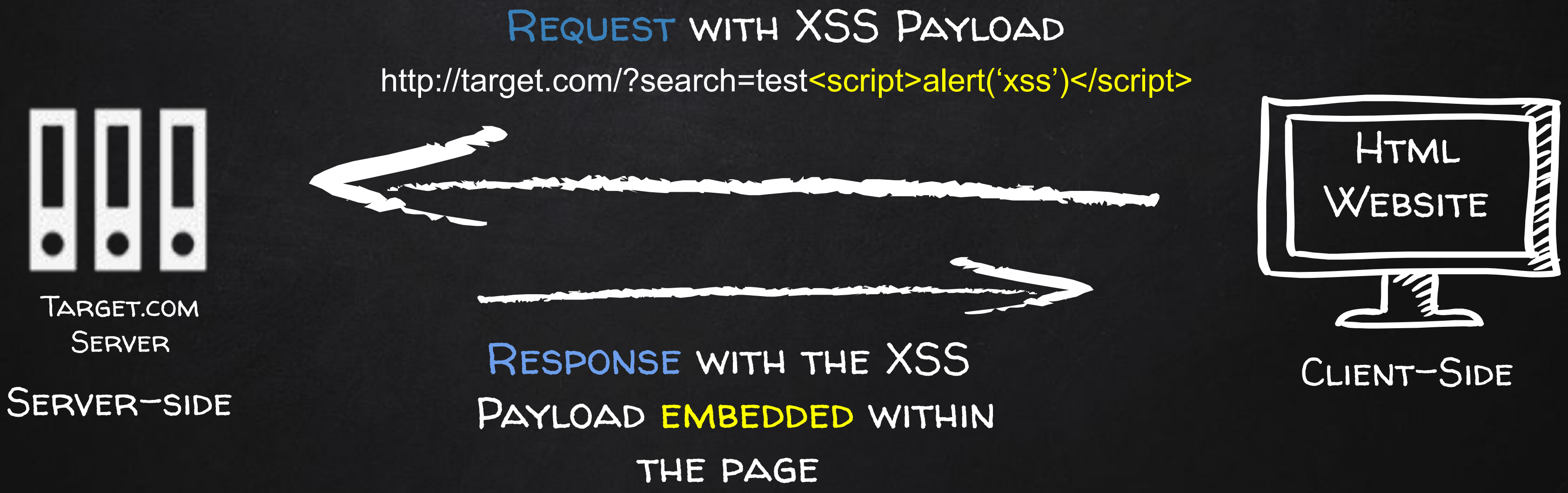
XSS Cross Site Scripting

Bypassing Security

Filters & Firewalls

XSS

Cross Site Scripting



Request with XSS Payload

http://target.com/?search=test<script>alert(‘xss’)</script>

Target.com Server

Server-side

Response with the XSS Payload embedded within the page

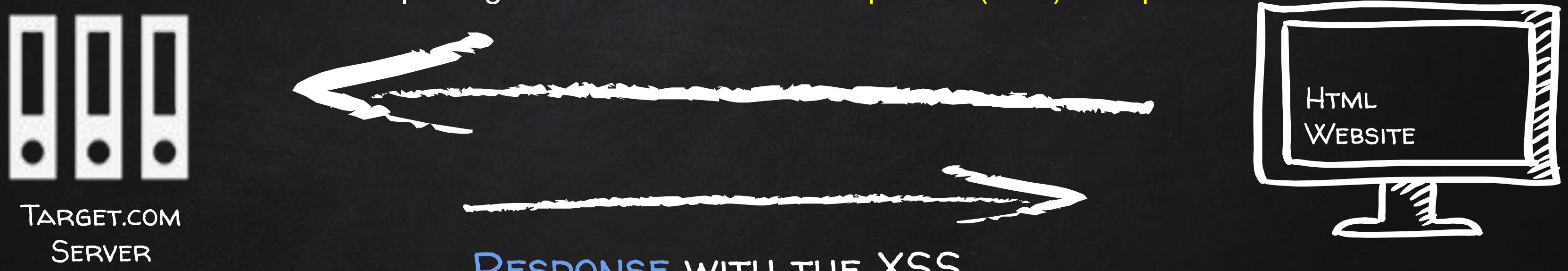
HTML Website

Client-Side

Reflected / Stored XSS

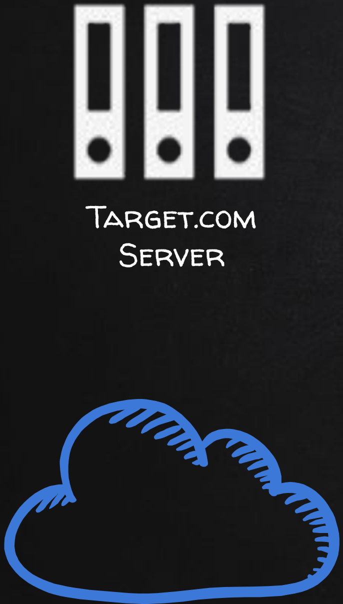
Request with XSS Payload

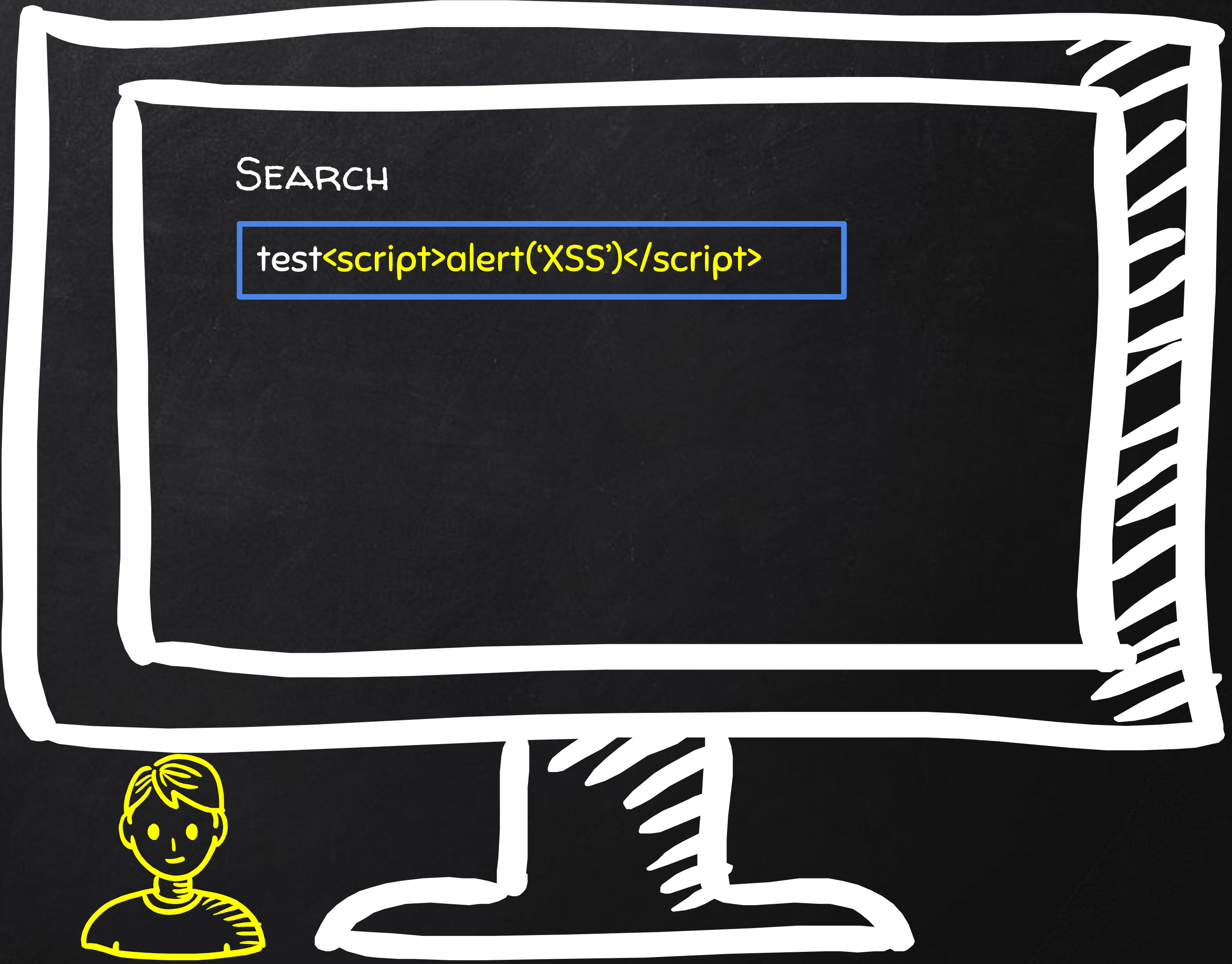
http://target.com/?search=test<script>alert(‘xss’)</script>



Response with the XSS Payload embedded within the page

Dom Based XSS



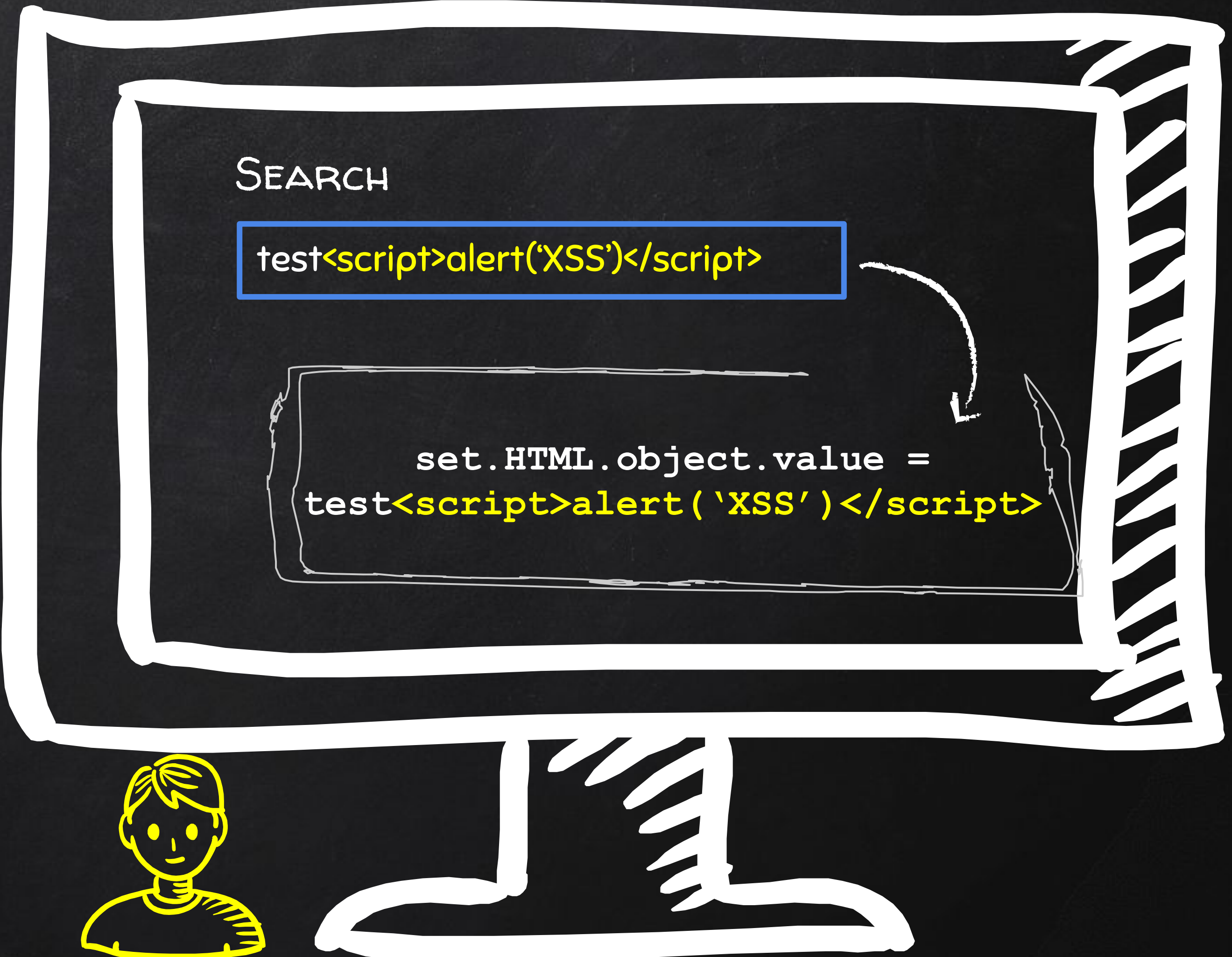


Search

test<script>alert(‘XSS’)</script>

Dom Based XSS

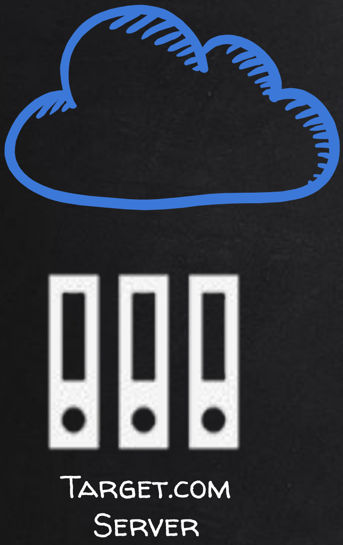
Target.com
Server



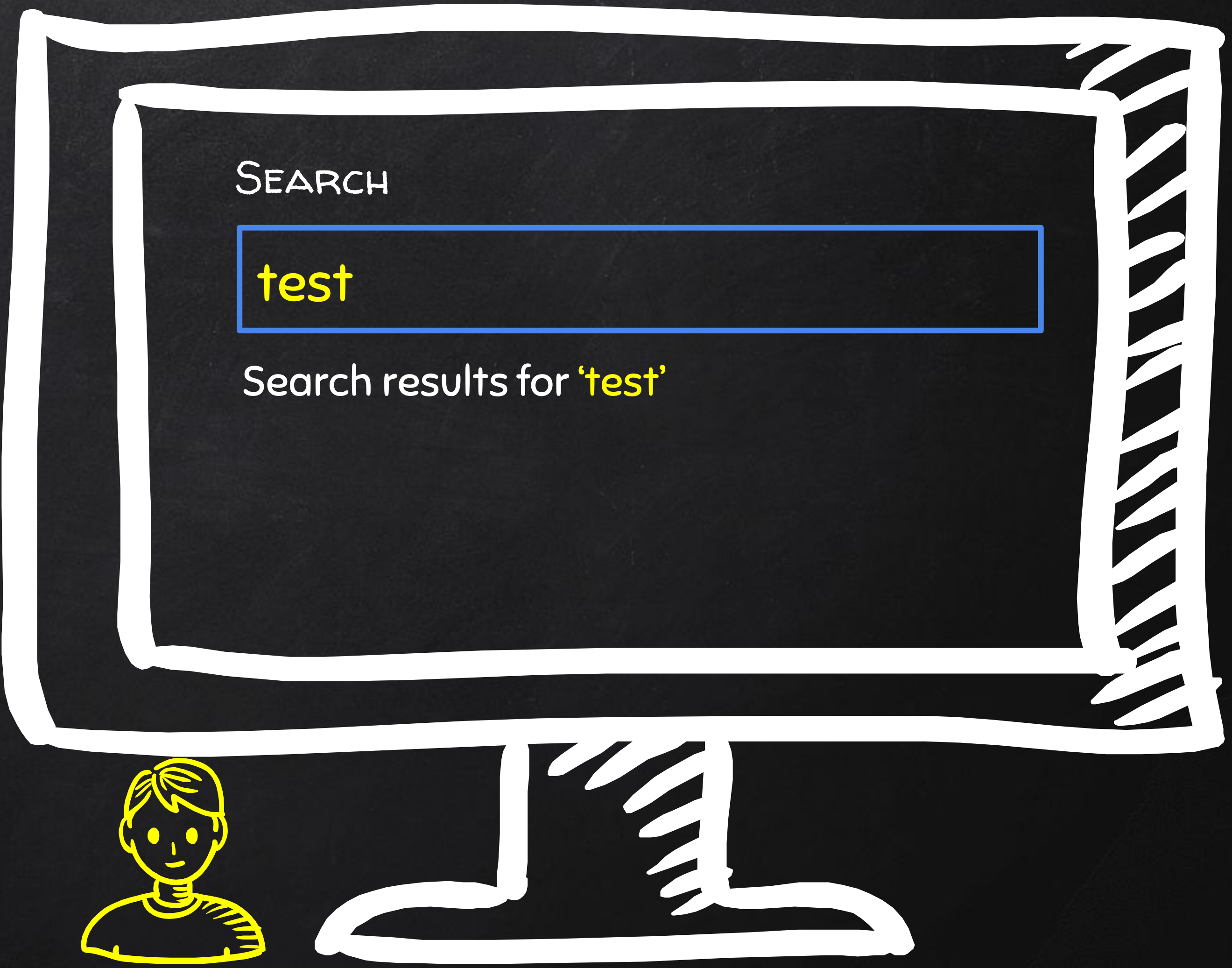
Search

```
test<script>alert(‘XSS’)</script>
```

```
set.HTML.object.value =
test<script>alert(‘XSS’)</script>
```



Dom
Stored
Reflected
XSS



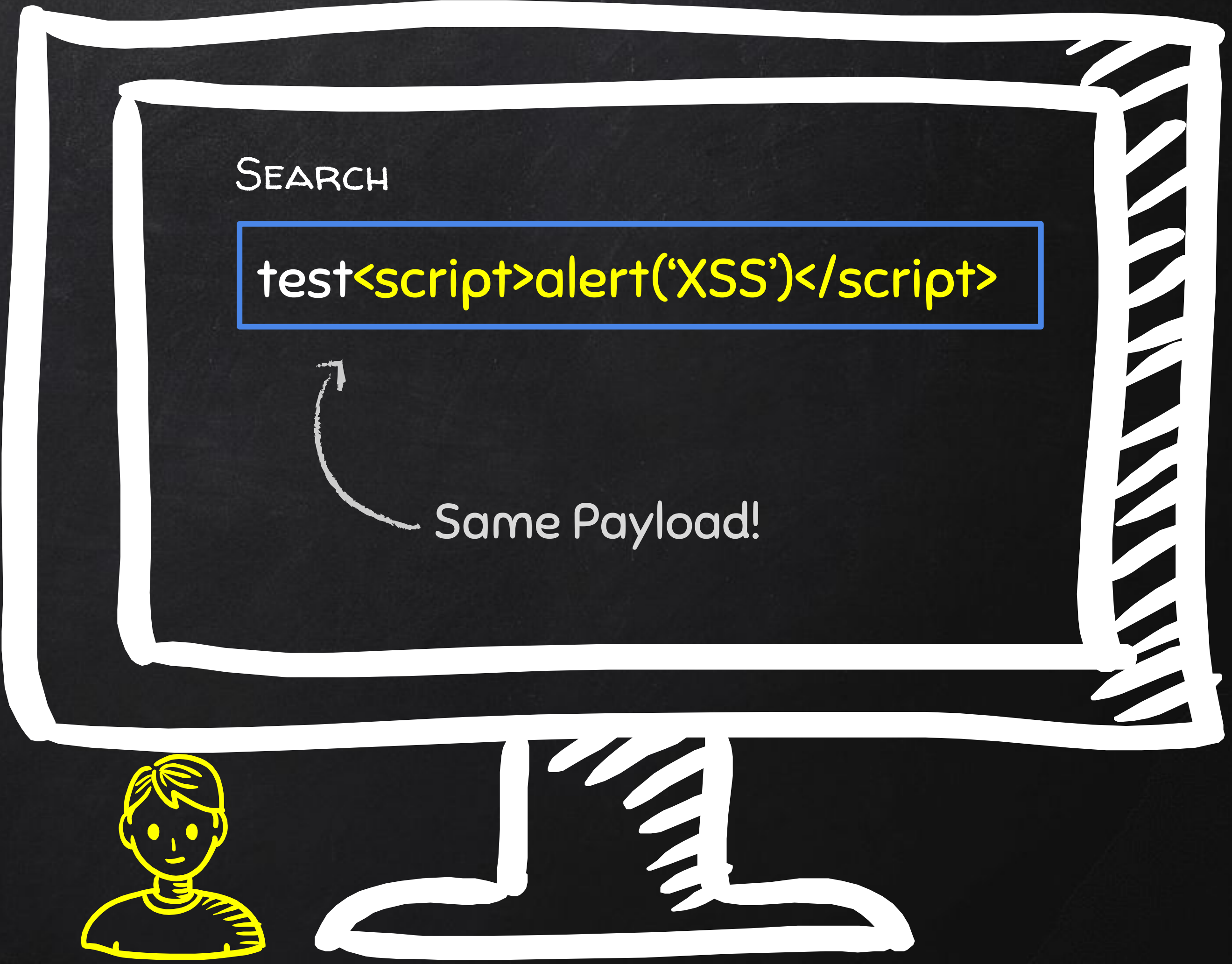
Search

test

Search results for ‘test’

Target.com
Server

Dom
Stored
Reflected
XSS



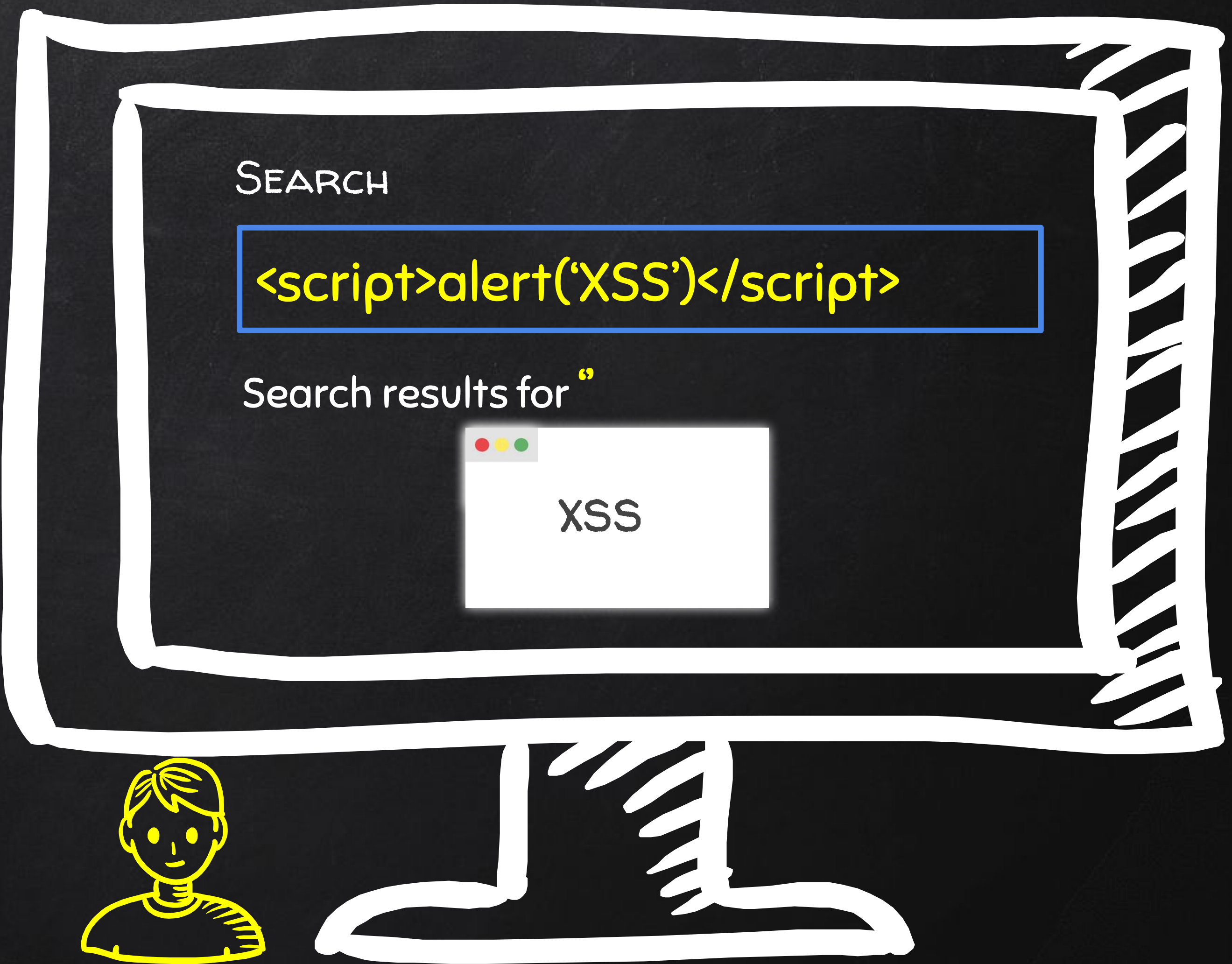
Search

test<script>alert(‘XSS’)</script>

Same Payload!

Target.com
Server

Dom
Stored
Reflected
XSS



Search

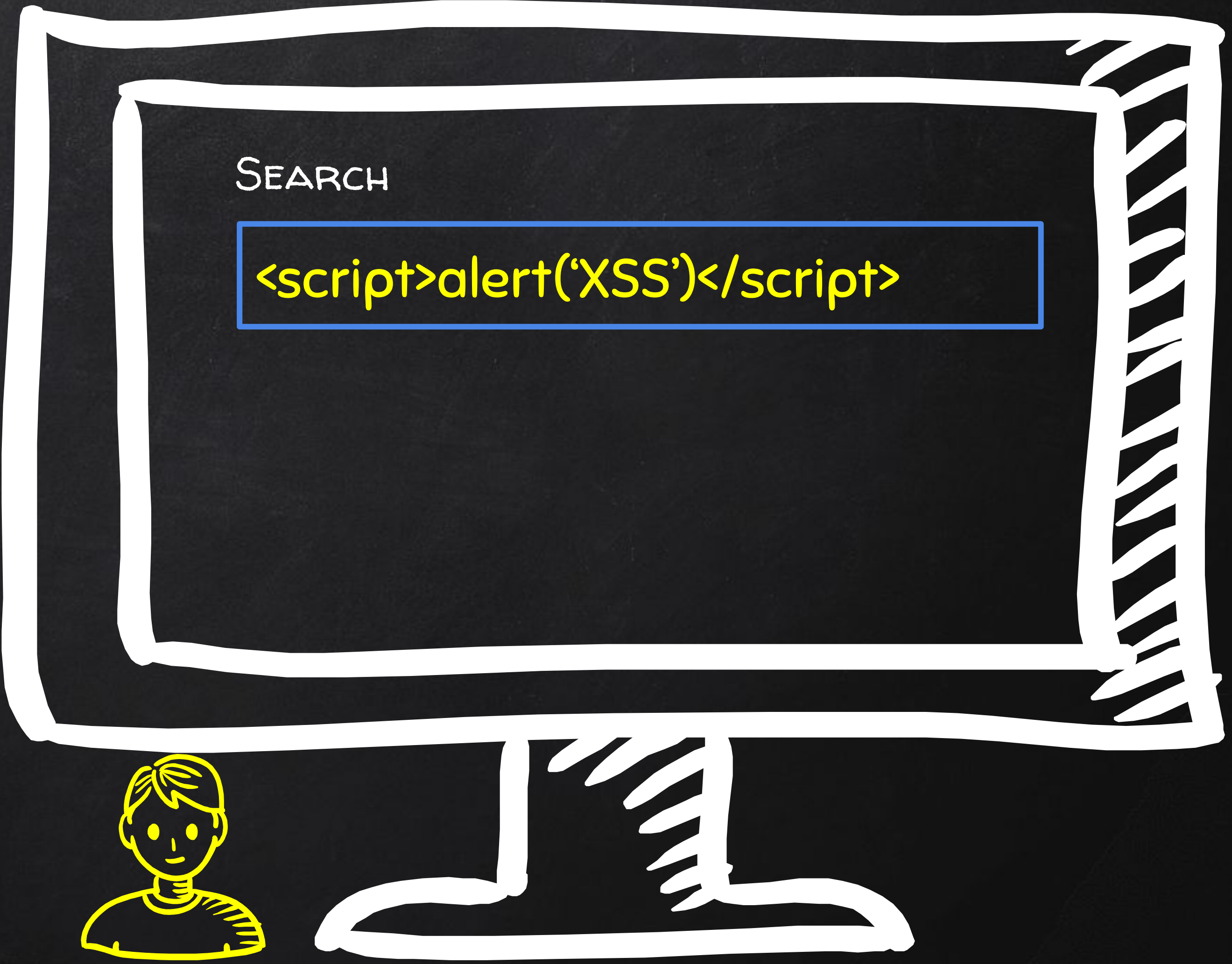
```
<script>alert(‘XSS’)</script>
```

Search results for “

XSS

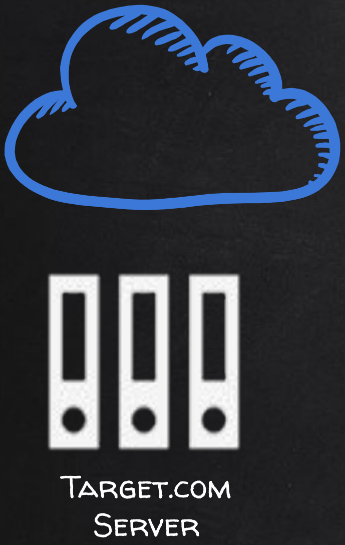
Target.com
Server

Dom
Stored
Reflected
XSS

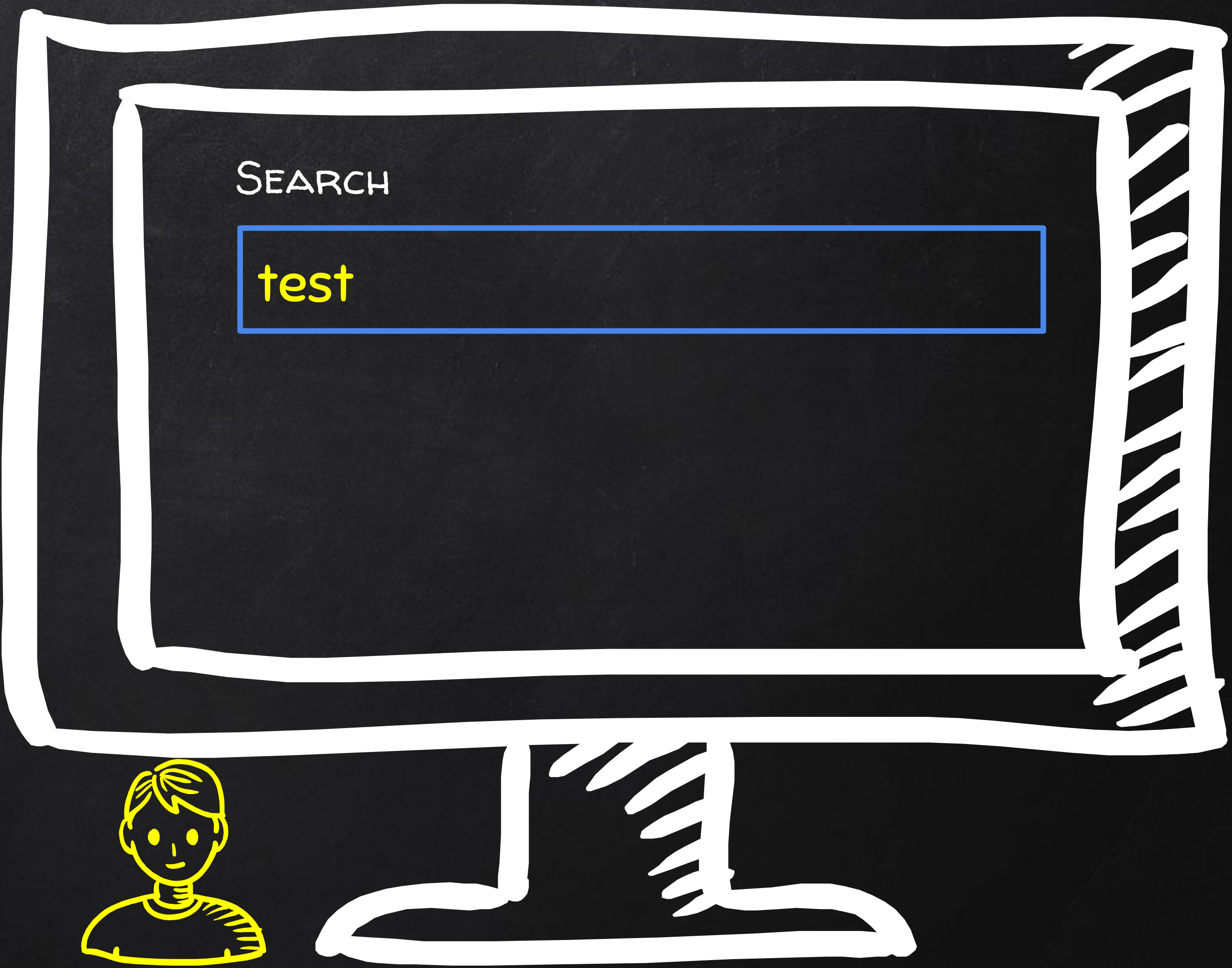


Search

```
<script>alert('XSS')</script>
```



Dom
Stored
Reflected
XSS

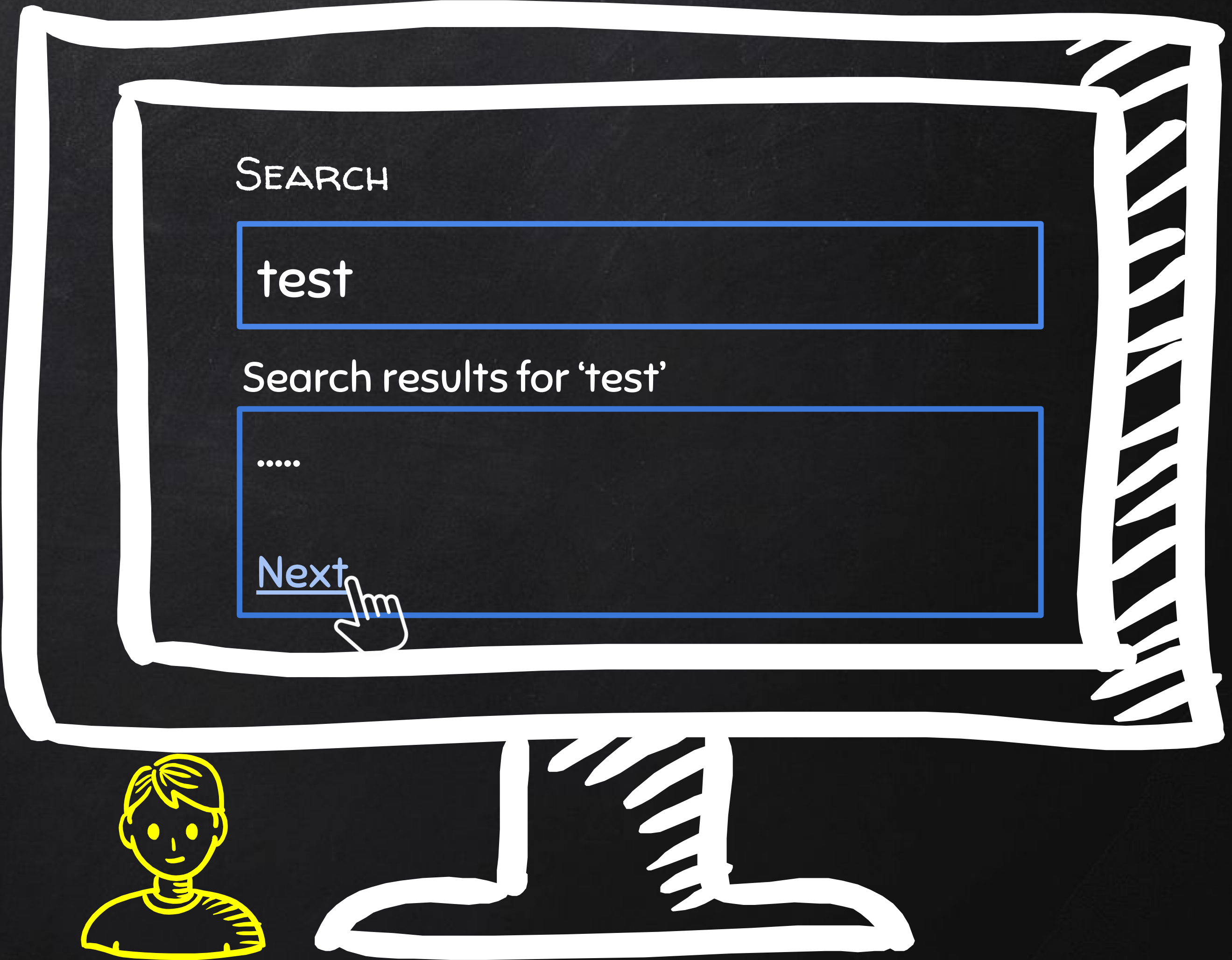


Dom

Stored

Reflected

XSS

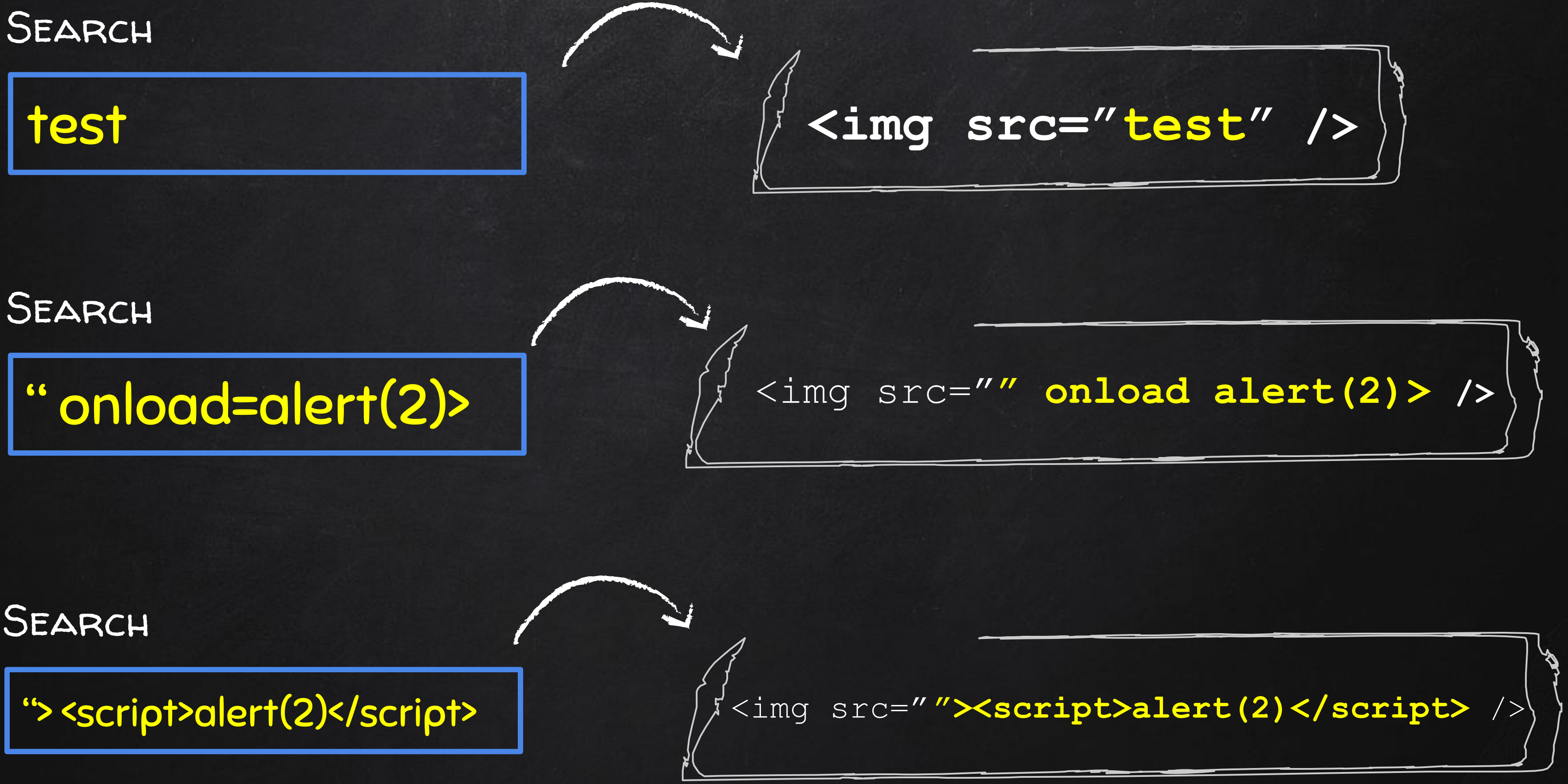


Search

test

Search results for ‘test’

·····



Search

```
test
```

```
<img src=”test” />
```

Search

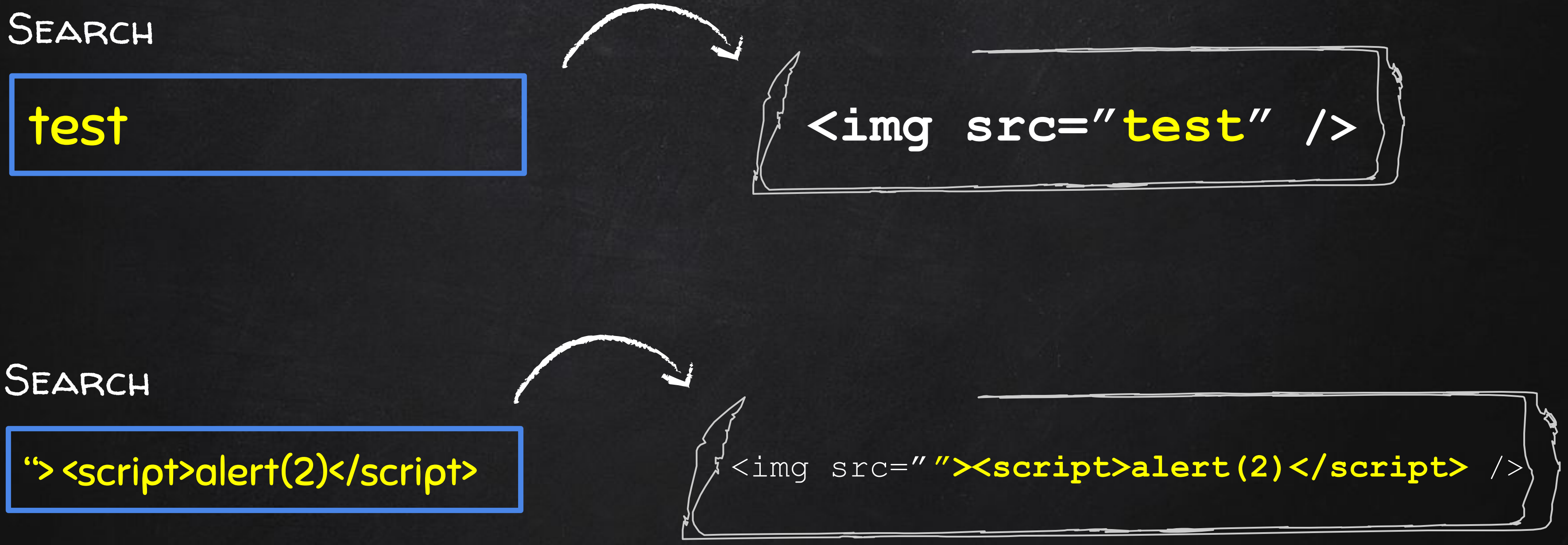
```
“ onload=alert(2)>
```

```
<img src=”” onload alert(2)> />
```

Search

```
“> <script>alert(2)</script>
```

```
<img src=””><script>alert(2)</script> />
```



Search

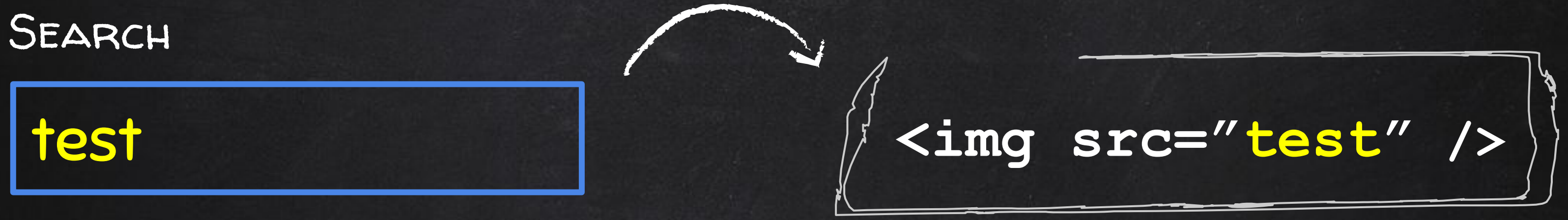
test

```
<img src=″test″ />
```

Search

“> <script>alert(2)</script>

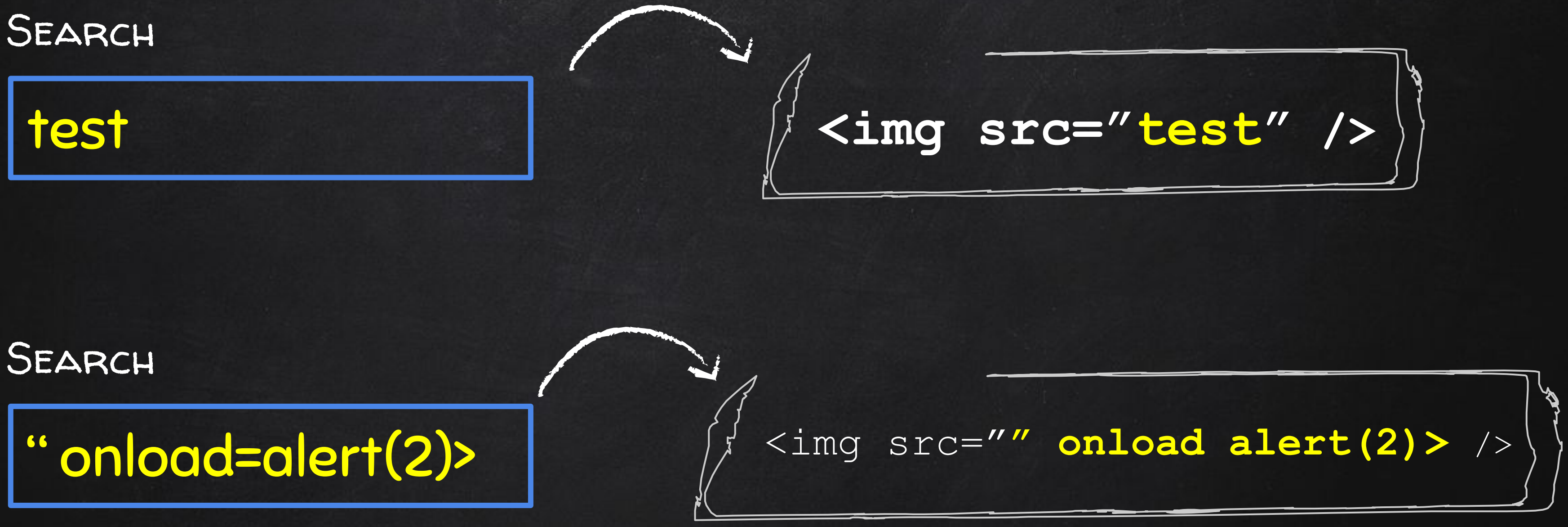
```
<img src=″″><script>alert(2)</script> />
```



Search

test

```
<img src=”test” />
```



Search

test

```
<img src="test" />
```

Search

“ onload=alert(2)>

```
<img src="" onload alert(2)> />
```

CSP & XSS

Content Security Policy CSP

- Browser feature that prevents XSS and other attacks.
- To enable it, response headers would include

```
Content-Security-Policy
```

XSS

Cross Site Scripting

Bypassing Security