



```
article.html.js
5   .push(tag.name.matches('js') ? 'tag-blue' : '')
6   })()
7   <a href="${tag.link}" class="${tag.classes.join(' ')}${tag.name}></a>
8

JS article.html.js x
1 module.exports = (scope) => '<article>
2   <header>
3     <h1><a href="${scope.link}">${scope.title}</a></h1>
4   </header>
5   ${require('../tags.html.js')(scope)}
6   <div>
7     ${scope.body}
8   </div>
9   </article>';
10

JS video.html.js x
```

HTML (Cyber Guy)

▼ Tags

```
<!DOCTYPE html> #I use HTML5
<h1> hi <h1> #closing tags
<img src ='x'> #tags doesn't need closing tags
<svg></svg> #distance
```

▼ html standard structure

```
<!DOCTYPE html>
<html>
<head>
<title> M8SJT8 </title> <!--title for the badge-->
</head>

<body>

<!-- this is how to comment in html -->
<b>M8SJT8<b> <!--Bold words-->
<i>M8SJT8<i> <!--italic-->
<b><i>M8SJT8</i></b><i>
<p>M8SJT8 Learning <br> html </p>

</body>

</html>
```

▼ Meta Tags

```
<!-- كود تعريفى -->
<meta name="description" content="Pentrayion Testing">
<meta charset="UTF-8"> <!--character encoding-->
```

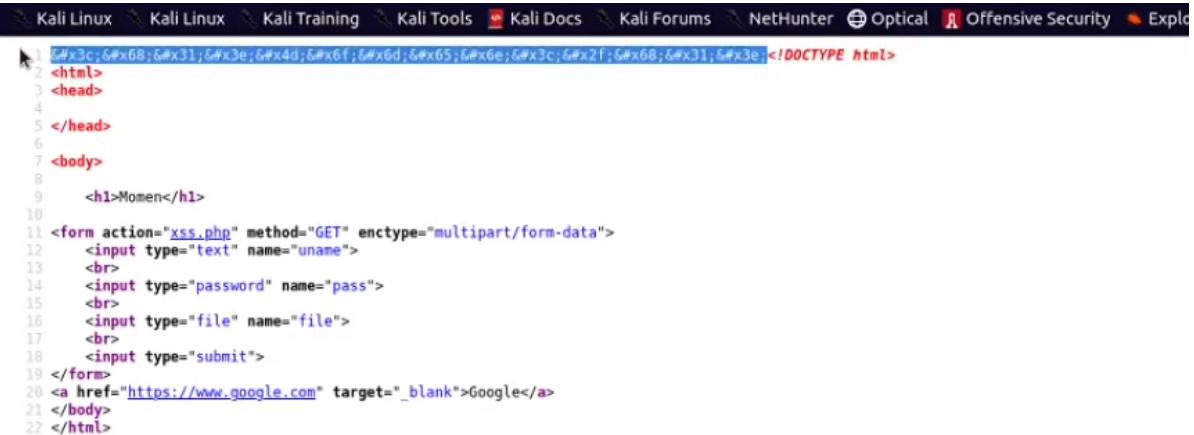
▼ form structure

```
<form action="" method="get">
    <input type="text" name="uname" placeholder="inter your user name ">
    <input type="password" name="upass" placeholder="inter your password ">
    <input type="submit" value="inter">
    <input type="image" src="0.png">
    <input type="file" name="add file">
    <input type="hidden" name="hacked" value="Yourpassword">
</form>
```

▼ form encoding

```
<form enctype="application/x-www-form-urlencoded"> <!--useless-->
<form action="" method="post" enctype="multipart/form-data"> <!--more common-->
```

▼ html encoding



The screenshot shows a browser window with the following address bar content:

```
1 <#x3c;#x68;#x31;#x3e;#x4d;#x6f;#x6d;#x65;#x3c;#x2f;#x68;#x31;#x3e;<!DOCTYPE html>
2 <html>
3 <head>
4
5 </head>
6
7 <body>
8
9     <h1>Momen</h1>
10
11 <form action="xss.php" method="GET" enctype="multipart/form-data">
12     <input type="text" name="uname">
13     <br>
14     <input type="password" name="pass">
15     <br>
16     <input type="file" name="file">
17     <br>
18     <input type="submit">
19 </form>
20 <a href="https://www.google.com" target="_blank">Google</a>
21 </body>
22 </html>
```

<h1>Momen</h1>

Momen

[Google](#)

▼ Event handlers

Cross-Site Scripting (XSS) Cheat Sheet - 2022 Edition | Web Security Academy



This cross-site scripting (XSS) cheat sheet contains many vectors that can help you bypass WAFs and filters. You can select vectors by the event, tag or browser and a proof of concept is included for every vector. You can download a PDF version of the XSS cheat

<https://portswigger.net/web-security/cross-site-scripting/cheat-sheet>

```
<!--
[1]- onclick
[2]- onload
[3]- onmouseover
[4]- onmouseout
[5]- onfocus
[6]- onerror
[7]- onkeyup
[8]- onkeydown
[9]- onkeypress
example:
<button onclick="alert('hello world')"> click me </button>
-->
```

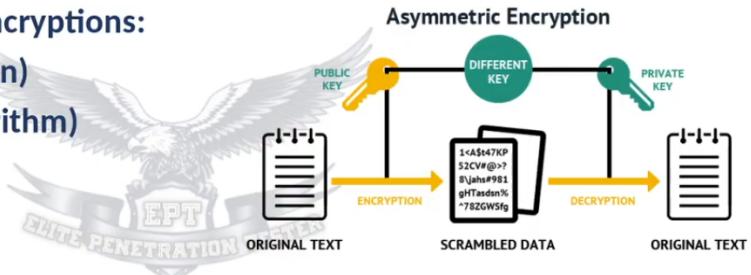
▼ Cryptography

• What Is Asymmetric

- Examples Of Asymmetric Encryptions:
- RSA (Rivest Shamir Adleman)
- DSA (Digital Signature Algorithm)

Want To Know More About Asymmetric ?

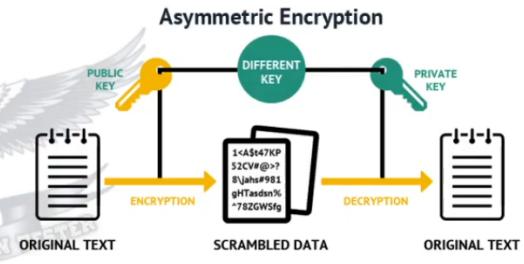
- → <https://bit.ly/3lO0VTS>



10:27

• What Is Asymmetric

- AKA Public-Key Cryptography
- In This Type Of Encryption We Use Two Keys, One For Encryption And Other For Decryption.
- When We Encrypt Some Text Or Data With Asymmetric Encryption There Is Two Keys Generated: A Public Key For Encryption And Private Key For Decryption.
- They Are Called ' Public and Private Key Pair.'



• **What Is Symmetric Encryption**

- **Examples Of Symmetric Encryption:**

- AES (Advanced Encryption Standard)
 - DES (Data Encryption Standard)
 - Blowfish (Drop-in replacement for DES or IDEA)
 - IDEA (International Data Encryption Algorithm)
- Want To Know More About Symmetric ?
• → <https://bit.ly/3tIjtHK>

Algorithm	Plaintext	After Encryption	After Decryption
DES	240 KB	328KB	240KB
TDES	240KB	614KB	240KB
AES	240 KB	847KB	240KB
Blowfish	240 KB	955KB	240KB
Twofish	240 KB	955KB	240KB

• **Why I Need To Understand Some Cryptography**

- You Need To Understand Cryptography To Know How The HTTPS Works.
- If You Understand How The HTTPS Encryption Works So That You Will Have A Deeper Knowledge Of Web Protocols.
- When You Understand The Concept Of Cryptography You Will Get Knowledge In Other Topics Like: HTTPS Certificates, Some Of SSH Mechanisms .. etc

• What Is Symmetric Encryption

- It's A Type Of Encryption That Uses Only One Key For Both Encrypt And Decrypt A Text.
- So If I Wanna Encrypt Some Plain Text To Make It Safe And Unreadable For Third Parties I Encrypt & Decrypt It With One Key Then I Send It – With The Key – To The Receiver.
- And The Problem Was Here, Because If Any One Was Able To Sniff The Network He Can Now Decrypt Any Message Encrypted With That Key

