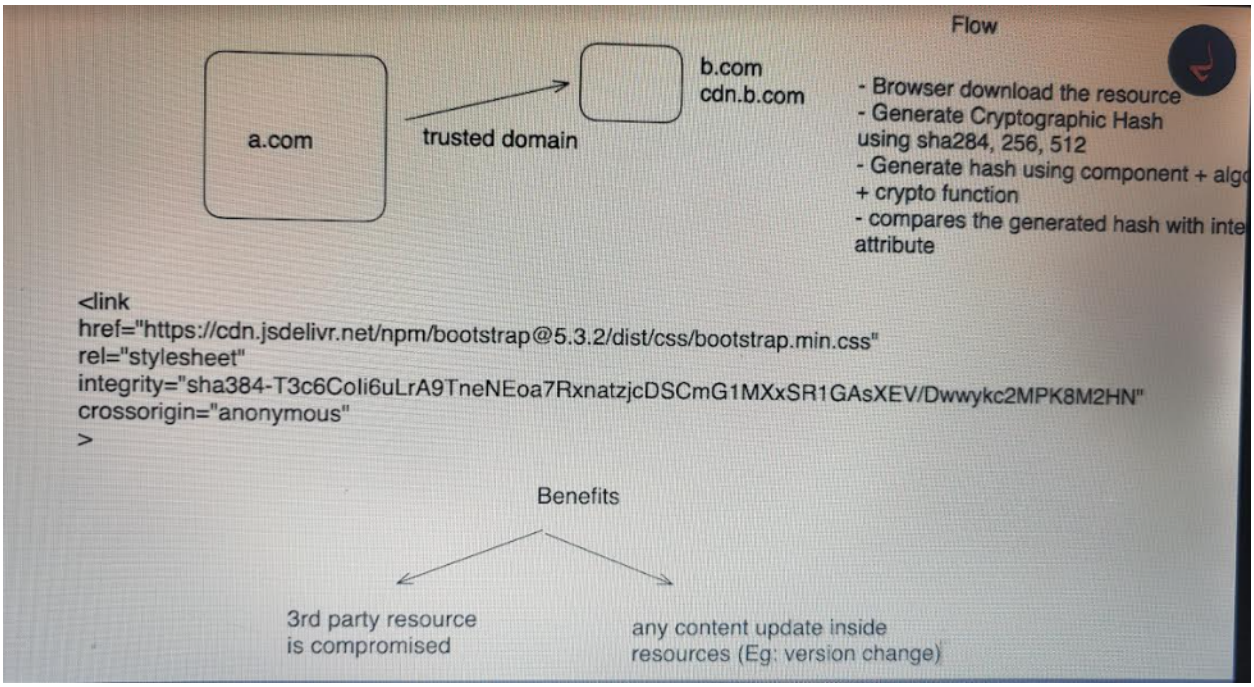
Subresource Integrity(SRI)

Suppose if you give your friend money. Something bad happens with him and he is not able to return the money. You trust your friend and your friend is good as well but still he is not able to return the money.

Subresource Integrity (SRI) is a security feature that enables browsers to verify that resources they fetch (for example, from a CDN) are delivered without unexpected manipulation. It works by allowing you to provide a cryptographic hash that a fetched resource must match.

https://developer.mozilla.org/en-US/docs/Web/Security/Subresource_Integrity



<link href=””.....>

What does it do ?

1. Downloads the resource from specified url in href attribute
2. Generates cryptographic hash using sha384, sha256 etc. It generates hash using content + algorithm + crypto function.
3. Compares the hash with value of integrity attribute specified
4. If value matches, it loads the resource
5. If CDN is crossorigin, we need to mention crossorigin=”anonymous” attribute as well

Benefits

1. If value doesn’t match, we know our 3rd party resource is compromised and it cannot be trusted.
2. If any content update happens in resources like version change or something, it ca break your application. We will get to know this.

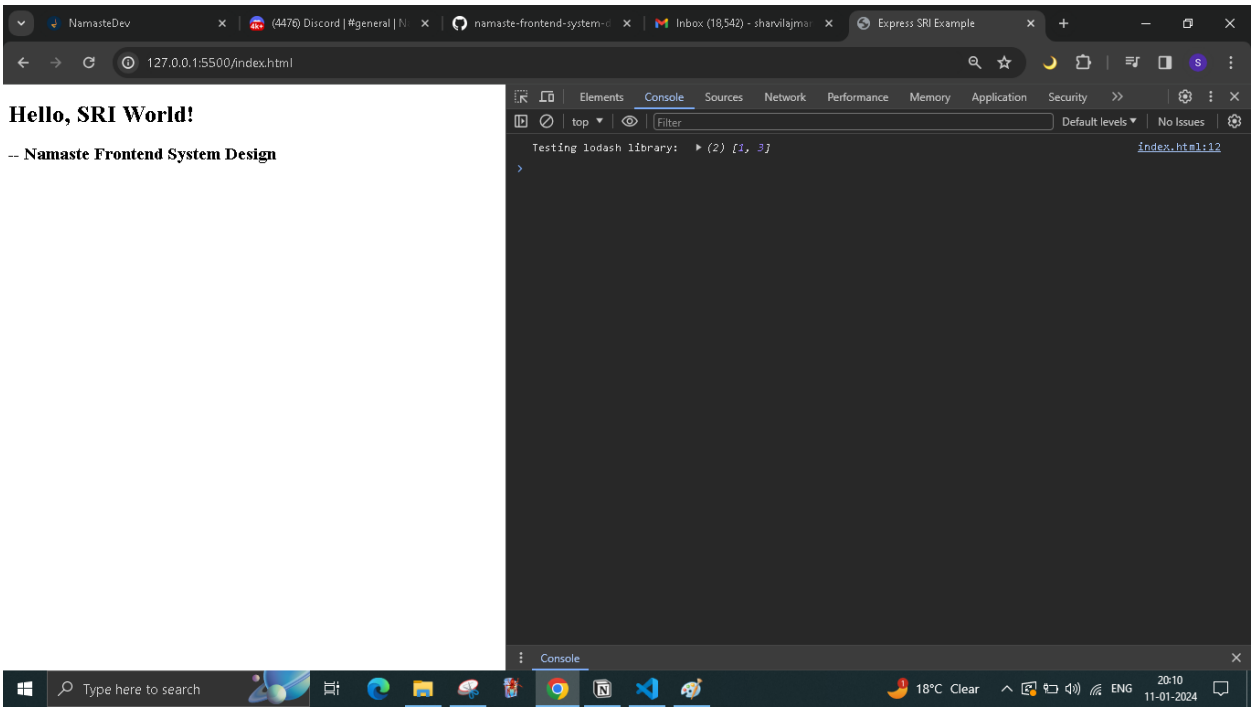
SRI hash generator - https://www.srihash.org/

Example-

We are loading lodash library using CDN and using it.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initia
l-scale=1.0">
    <title>Express SRI Example</title>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/lodas
```

```
h.js/4.17.21/lodash.min.js"
        integrity="sha384-H6KKS1H1WwuERMSm+54dYLzjg0fKqRK5ZRy
ASdbrI/lwrCc6bXEmtGYr5SwvP1pZ"
        crossorigin="anonymous"></script>
    <script>
        var sample = _.filter([1,2,3], (data) => data % 2 );
        console.log('Testing lodash library: ', sample);
    </script>
</head>
<body>
    <h1>Hello, SRI World!</h1>
    <h2>-- Namaste Frontend System Design</h2>
</body>
</html>
```



Now make some random changes in integrity attribute and reload web page again.

It will give error.