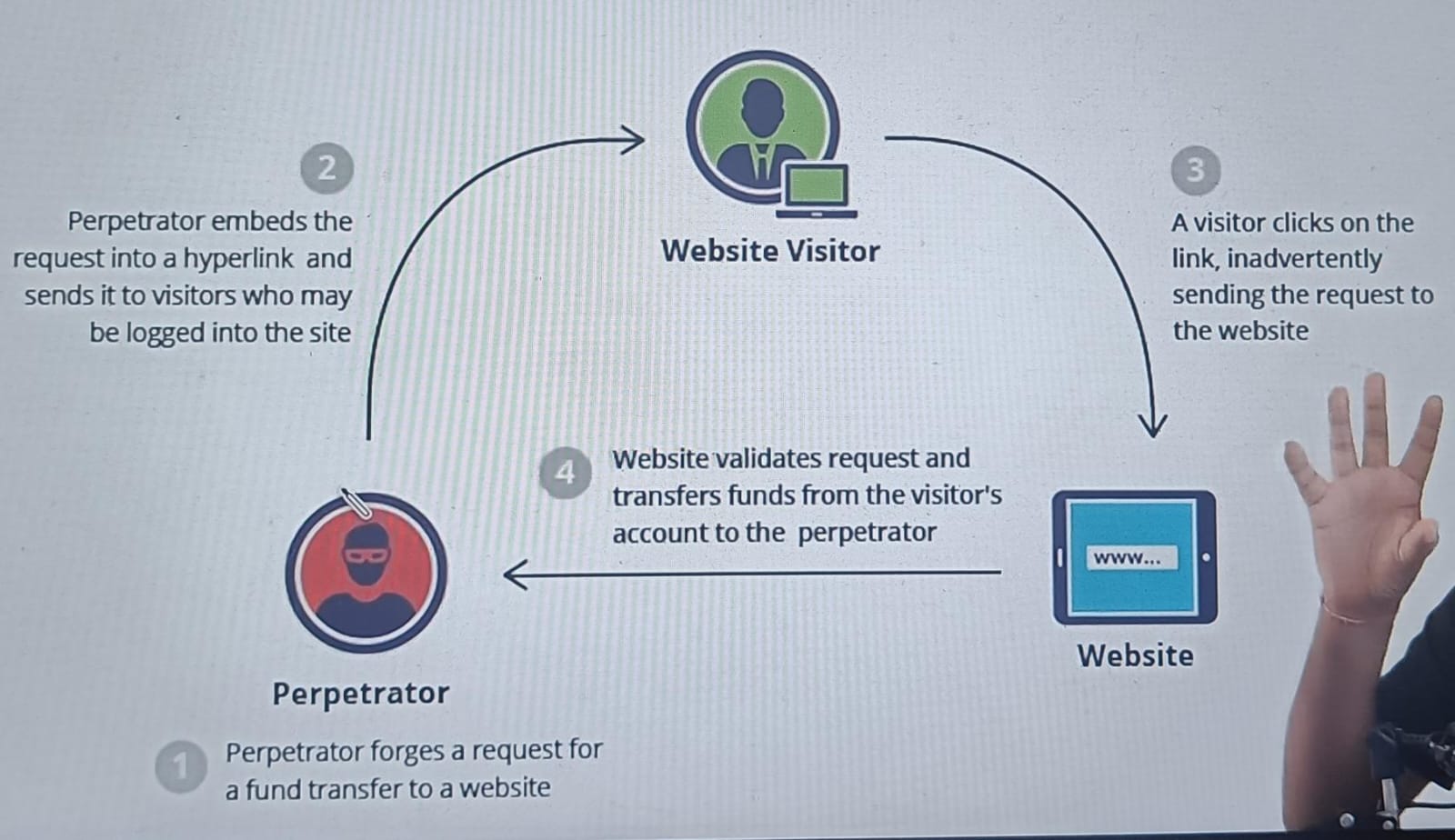
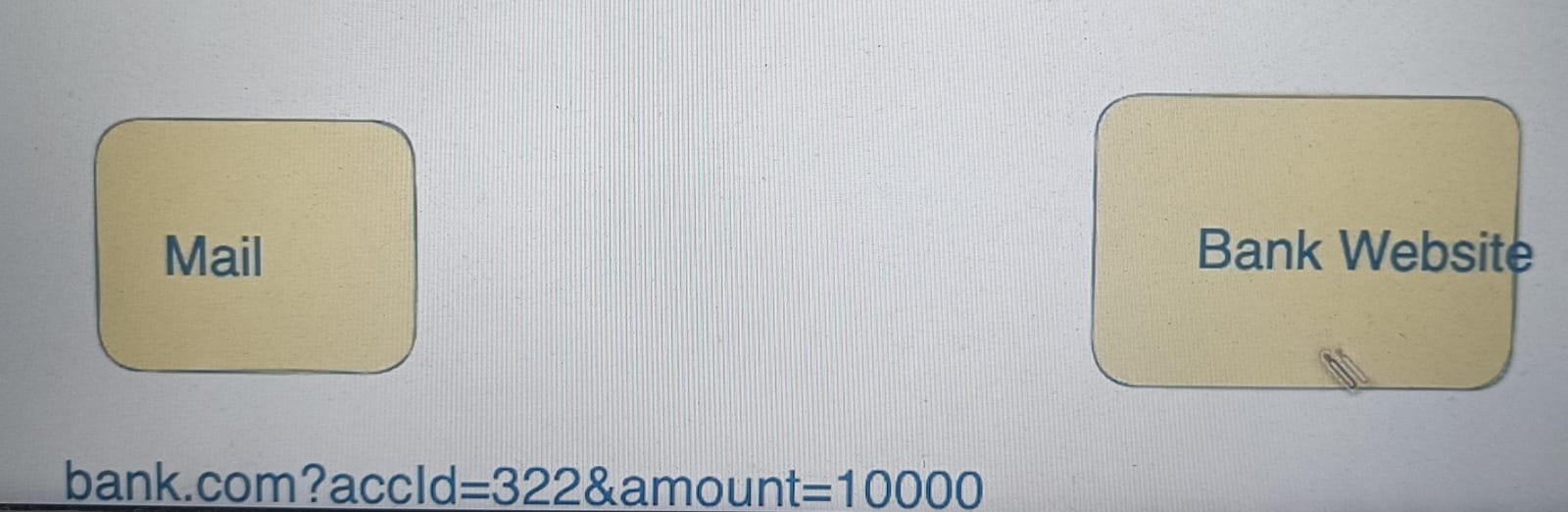
Consider example if you are a banking website and in banking website you have a mechanism that in url you give a account no and the amount that need to be transferred it will be transferred to that account, now say suppose this request is made from via some other website , it can be via any spam mail or any website which have offer kind of thing and when you clicked on that you went to the bank website with that particular url of bank website which has account where the amount need to be transferred and suppose it has amount which need to be transferred and that particular banking website has your session which you already logged in, it means you have already logged in into particular bank website and someone asked you to transferred the amount in a particular account so what you will do? That particular banking website make a request and basically do a transaction.

In real world scenario we have some mechanism to save guard those thing like it ask for otp, multi factor authentication and what not, but in general if you don’t have those securities then it will be make such transactions, so why such thing can happen in general we have to understand any malicious website is able to make a request to any other cross site and without the user intent something is getting executed and some actions are performed, consider example I clicked on a mail which I got and somehow I see on the messenger all my friends got some chat message or I clicked on mail and I redirected to my facebook account and I see ohh there is some post which went without my consent, I did not send a post but somehow a post is being send without my consent that is how this cross site request forgery will happen.



So example if you are a hacker here basically or someone who wanted to do some malicious over here, so in this case we are not going to inject some script where your script will steel some information like we are doing in xss attack, no I am going to forge the request so I know this is a api this is a endpoint which is used to transfer in your application so I am going to create those request and from a different domain I am going to ask you to click on that so once you click on that your request goes to that respective website with the embedded hyperlink where user is already logged in then what will happen visitor will click on particular link and finally this abc.com will be able to do fund transfer.



Inside your mail you are going to inject something called bank.com with account id and amount and when user will come and click on the mail some button which have internal url injected it will redirected to the website which is bank.com and inside this you will see one interested thing will happen, if you are already logged in the bank website then this url will trigger on the backend and backend will say ohh I have the cookie I have all the information why don’t I validate and transfer to this particular amount.

So, CSRF majorly happens because of 2 things –

* Statelessness of HTTP
* User authentication

Vulnerabilities –

* Get API call to update data or perform action

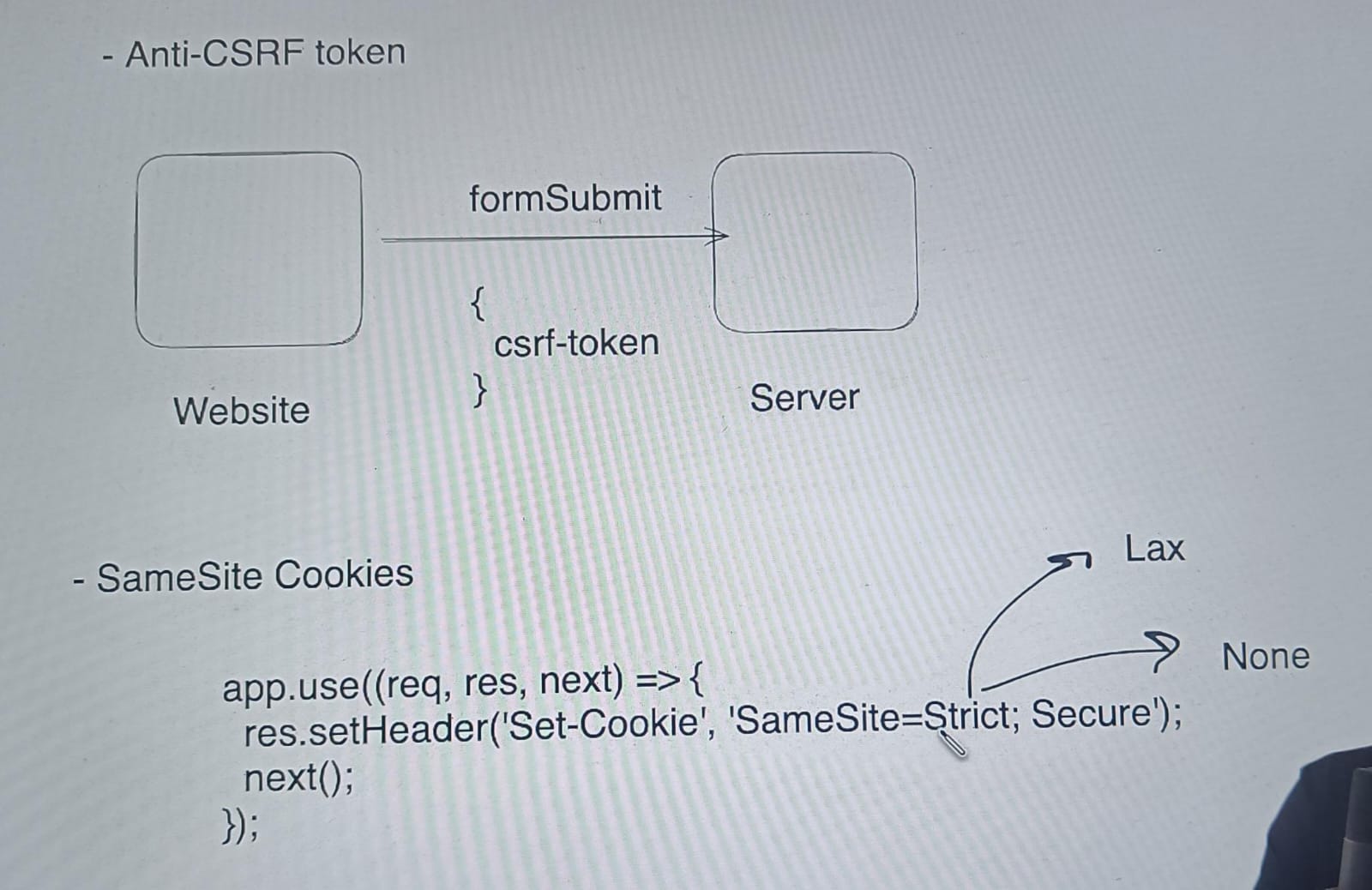
<http://bank.com/fundtransfer?accId=21312&amout=10000>

<a href=”” /> Offer</a>

<img src=””/>

Mitigation –

* Anti-CSRF token – so problem happen whenever someone else is making a request and you logged in and it has cookies and everything along with it and you never get to know that request is right or request is correct you don’t know, if I say that every request that you make I am going to give you a token with that token only you should respond me back, I am not saying I am going to give you a token inside some header which you can read and send me back no, every time when you try to render this form kind of thing I am going to tell you that this is a token that you should make a request in the next time, when you going to send a token with a request I am going to validate authentication, if you are a valid user and what not along with that I am going to see for this request this was a token that you suppose to send, did you send a token with a particular request that is something which I am going to validate. How lets try to see.



* Referrer-based Validate
* Use captcha
* CSP header
* Please do not use get method for update operation