Date: 02/11/2023

Assessment 2

In this assessment we will be learning about data breaching. We'll be understanding about what is data breaching, what are the ways data breaching can be done and we'll be performing it on a target as well to demonstrate the steps involved

Objectives

This assessment encourages us to learn what is data breaching and understand it's criticality. Data breaching is a very common attack and severe hence we must be aware about the consequences and defence regarding the same

Data Breach

Breaching, means making a gap in a wall and breaking through which makes it quite clear that the attacker is finding a gap or purposely making a gap in order to break through the system and get the unintended data. The attacker may gain access to the sensitive data that the company or organisation never wants it to get exposed. By this, the attacker can get access to the fragile data like usernames and passwords, credit card and debit card details, personal identification number, etc. If the hacker gets the PIN number for example, then he can steal all the money and get unreachable. If he gets the username and password if administrator, then he may login as administrator into the company and make unwanted changes or can change the password and threaten to pay money in order to get back the password.

Data breaching can be done using various techniques. It can be done using SQLi (SQL injection), XSS (Cross Site Scripting) or any other. It can also be done using Command Injection to return the file contents having sensitive data. Also, it can be implemented using XSS to get the cookie value which can be used to login without using credentials and get control over the account.

SQL

SQL stands for Structured Query Language which is associated to the RDBMS (Relational Database Management System) used to store data in the backend. SQL as the name suggests, stores the data in a structured way i.e., in a table fashion. The data can be anytime fetched and can be stored. SQL supports connectivity with different languages like PHP, Python, Java and others which gives the programmer a lot of flexibility and increases the capabilities of an infrastructure. The SQL can be implemented by many SQL engines that are available. Some of them are MySQL, MSSQL, Oracle, PostgreSQL, etc.

SQL are used in different platforms like web applications, mobile applications and others. The data can be systematically stored and whenever the user tries to get a data, the programming language connected to SQL can run a query in the backend and fetch the particular data accordingly. The most commonly used engine is MySQL and this service runs on port 3306. It can be altered when the installation process is running and make it run on a different port. The different platforms differ very slightly in terms of syntax. For example, the comments in MySQL can be written with #, /**/ but in MSSQL it is done by a -- (There's a space after two hyphens) and similarly in Oracle. Overall, SQL increases the data maintainability as a result, it is easier to handle data.

SQLi

SQLi stands for SQL injection which basically means trying to send a malicious code along with the traditional code in order to get contents that are not supposed to be displayed. If the web application at the backend doesn't validate the inputs and doesn't handle errors properly then the hacker can "inject" his code into the regular request and fetch data other than requested. The SQLi can be done on a GET URL with parameters and POST request body. Let's take an example:

Let the GET URL look like this:

http://website.com/products.php?name=toys

and if this request, at the backend execute the following command:

select * from products where name='toys';

By manipulating the URL, we can inject our code to get more details. For example, if we corrupt the URL to:

http://website.com/products.php?name=toys' union select null--+

then the query would look like:

select * from products where name='toys' union select null-- ';

Here, anything after (--) will be commented out and nothing will be considered after that. Hence the query executing will return the products under toys and return one null value also. If that can be seen then the web application is vulnerable to SQL injection. Now, the attacker can use information_schema to get the details of different databases, tables in them, no of columns in each and get the data by doing multiple manipulations

SQLi in My Target

I was going through a website which was vulnerable to SQLi. The website had a dedicated page which showed the details of employees working under that organisation. Whenever I change the starting letter of the employee's name or the department or any other factor, there was a POST request sent to the server. On investigating, I understood that the server does not check for illegal characters or malicious words in the request and directly picks the POST body parameters and puts it into the query. I found it by the following request:

POST /school_member/showFacultyDataAjax HTTP/2

Host: XXX.XXXX.XXX.XX

Cookie: ci_session=d2be65mgfs3uku7r2qq2hckvdk0mom5m

User-Agent: Mozilla/5.0 (Windows NT 10.0; rv:109.0) Gecko/20100101 Firefox/115.0

Accept: */*

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Content-Type: application/x-www-form-urlencoded; charset=UTF-8

X-Requested-With: XMLHttpRequest

Content-Length: 67

Origin: https://www.reva.edu.in

Dnt: 1

Sec-Fetch-Dest: empty

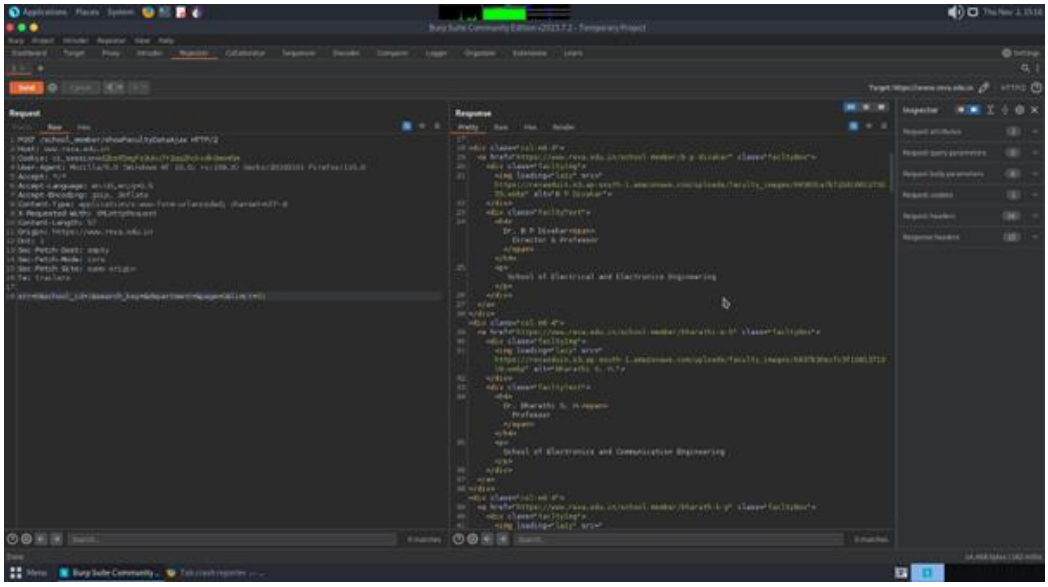
Sec-Fetch-Mode: cors

Sec-Fetch-Site: same-origin

Te: trailers

str=B&school_id=1&search_key=&department=&page=0&limit=51

And the response was:



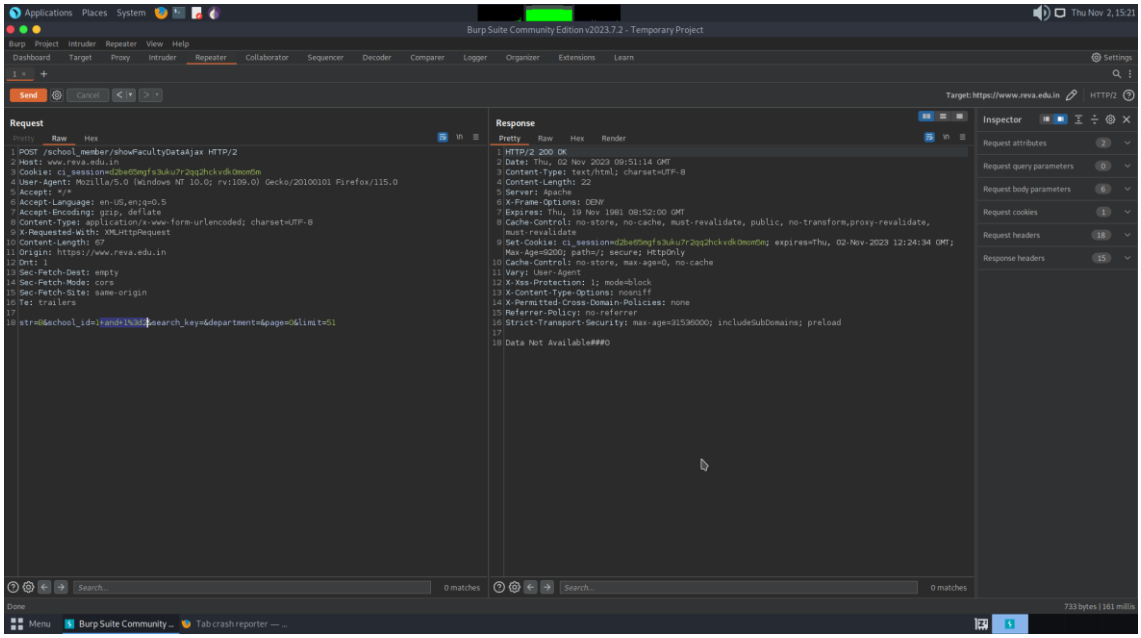
The POST body had many parameters which on checking, I found that changing the school_id=1 parameter to:

school_id=1+and+1%3d1

gave the same response and then changing the parameter to:

school_id=1+and+1%3d2

And the response was:



Here, we got the response saying “Data not available” which means that Boolean based SQLi was possible and hence it was vulnerable to SQLi and data breaching is possible

Conclusion

In this assessment we understood about data leakage and how it is done. Data leakage can have dangerous consequences if not handled wisely. It can cause dangerous situations like losing of accounts in organisations, losing money in banks, misusing of personal data, etc.

Having really strong combination of passwords, unguessable PINs (not having birth date or year as PIN), not keeping any sensitive data in open files, not browsing through important sessions on public Wi-Fi, all these can be done as individual. But as a cyber security analyst, keeping an eye on log files, using Web Application Firewall (WAF), using tools like Wireshark, and other steps can be taken to avoid data breaching