Assignment 5	**Tyler Wilding**
COSC 4426	Due Date: 23/10/16
Prof. Biocchi	Source Code
TA: –	Live Demo

Suspicion Critera

Login with admin@notturing.com:Test1234, and go to the admin dashboard and you can see all of my test cases displayed.

When looking for suspicious activity, I tried to identify anomalies that a malicious user would do when logging in. The first of these would be what I call a **Targetted Attack**. This would be if someone wanted to get into a specific account and know some sort of information as to what the password might be and would try several times but not an impractical amount, over 10 invalid logins in a day.

The second suspicious activity would be **spam**. This would be when a bot or malicious user makes a new account that is not verified and attempts to login, over 5 attempts to login to an un-verified account.

The third suspicious activity was a **Bruteforce or Dictionary attack**. Either of these attacks would be obvious by the insane amount of login attempts in a short timespan, I look for over 10 failed logins per minute.

The fourth suspicious activity is a **suspicious location**. A user will typically log in from the same IP address routinely, a suspicious location is when suddenly someone logs in from a completely different IP than usual.

The fifth suspicious activity is an **XSS attack**. If the malicious user enters text that resembles HTML or another languages tags into the email field, it will be detected.

The second last suspicious activity is a simple **SQL injection** attempt. Similar to the XSS, if the user attempts to enter something that resembles an injection attempt, it will be flagged.

And lastly, detecting a **proxy connection**. One of the easiest to detect, if the user is using a proxy it can easily be detected with PHP because the user will most likely have a forwarding HTTP address.

How the Tool Works

All login attempts are stored to the database whenever the user tries to log in, whether it is a successful login or not. As a result, we can then query the database and scan the table. The tool runs once, and updates another table in the database which logs all suspicious findings, a button in the dashboard can be used to run this tool and update the logs.

I debated on using an external program, but it didnt make sense with all the existing code I have that already connects to the database, so all of the work is done using php.

The main process is querying the database for all unsuccessful login attempts, and then using this information to scan through and find anything that meets the criteria described above.

Queries Used

Spam Accounts

```
SELECT email, count(result) FROM tyler.login_attempts WHERE result = 0 GROUP BY email;
```

Suspicious Location

```
SELECT email, ipaddr, count(*) FROM tyler.login_attempts WHERE result = 1 GROUP BY
    email, ipaddr ORDER BY email, count(*) DESC;
```

Proxy Connection

```
SELECT * from tyler.login_attempts WHERE ipaddr_fwd != ;
```

10 Incorrect Logins in a Day

```
SELECT email, ipaddr, count(*) AS ‘count‘, DATE(timestamp) as ‘date‘ FROM
    tyler.login_attempts WHERE result = -1 GROUP BY DATE(login_attempts.timestamp),
    email, ipaddr HAVING count >= 10;
```

10 Incorrect Logins in a Minute

```
SELECT email, ipaddr, count(*) AS ‘count‘, date_format(timestamp, ‘%Y-%m-%d
    %H:%i:00‘) as ‘date‘ FROM tyler.login_attempts WHERE result = -1 GROUP BY ‘date‘,
    email, ipaddr HAVING count >= 10;
```

How it Could be Better

One problem with detecting suspicious logins is that it could be a log easier to determine the nature of the attacker, if the password could be stored in plaintext. This could be used to see if it was a bruteforce attack, however this obviously opens up the security hole of having plaintext passwords, as well as near-miss passwords in the database.

One issue with some of the detection strategies, for example for XSS or SQLi attacks, is the very high potential for false positives. A user may just have a semicolon as part of their password, but this would be picked up as an attack. Likewise, a user may just have a bad memory and forgot their password, yet they will be flagged as targetting that account and trying to access it instead of using the password reset page.