INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)

www.ijprems.com
editor@ijprems.com

Vol. 03, Issue 05, May 2023, pp : 1121-1124

e-ISSN : 2583-1062

Impact Factor : 5.725

PASSWORD SPRAYING DETECTION USING API

Rahul S[1], Manoj Kumar D[2], Tris Ram V[3], Shimona S[4]

[1,2,3,4]Student, Computer Science and Engineering, Agni College of Technology, Chennai – 600 130, Tamil Nadu, India

DOI: https://www.doi.org/10.58257/IJPREMS31434

ABSTRACT

At various stages of a breach, adversaries can use password spraying. It can be utilized to gain initial access to a system, but it can also be utilized to increase privileges after access has been gained. In many cases, this strategy ironically makes use of a password rotation security measure that is frequently used by businesses. Some business users may choose predictable passwords when theyupdate them after they expire.

Specifically, in two cases when an attacker has gained access to the target network, detecting potential password spraying attacks against Active Directory installations.

Hackers can rely on companies using usernames that are identical to those found in public domains when they are able to get data on employees frompublic sources. These usernames will be concatenated by the hacker.

1. INTRODUCTION

An attack method known as "password spraying" allows criminals to gain access to legitimate account credentials by using a single password or a short list of widely used passwords on a large number of usernames. Password spraying uses the opposite strategy from brute force attacks, which aim to obtain valid credentials while avoiding account lockouts by targeting a single user or small group of users with a huge number of passwords. If the target organization doesn't have the right monitoring and detection controls in place, this permits adversaries to go undetected. The use of this potent tactic has beendocumented by penetration testers, cybercriminals, and nation-state actors.

Unlike brute force assaults, password spraying targets large volumes of passwords rather than single ones, which makes it different from brute forceattacks.

2. LITERATURE SURVEY

Mitnick Security et. al., Cybersecurity measures are in place at many organizations to stop threat actors from getting past security and launching

attacks. Although, there may be a gaping hole in your organization’s security: untrained employees.

Threat actors can take advantage of the poor security habits of your network and system users through a technique known as password spraying in order to gain access and wreak havoc on your organization. Below, we’ll discuss password spraying attacks demonstration video included and what you can doto mitigate the risks.

Citrix et. al., proposed that Both a drive connected to a web-based application used in Citrix's consulting business and a shared network drive included business documents that were stolen by the attackers. As a result of Citrix's failure to build a strong password strategy, the software firm has come under fire for allowing the hackers to access its IT infrastructure via a password spraying assault, which takes advantage of weak passwords.

Citrix is by no means the only business that struggles with password security. 44 million Microsoft users were using the same usernames and passwords that had already been made public online following security breaches at other online services, it was discovered by a threat research team scanning all Microsoft user accounts in the early months of 2019.

3. EXISTING SYSTEM

An attack using password spraying takes place in two steps. A hacker obtains a list of usernames and then uses the same password to try to log in with each account. To get access to accounts and systems, the attacker keeps going through the process with fresh passwords until the target authentication systemis compromised**.**

PROPOSED SYSTEM

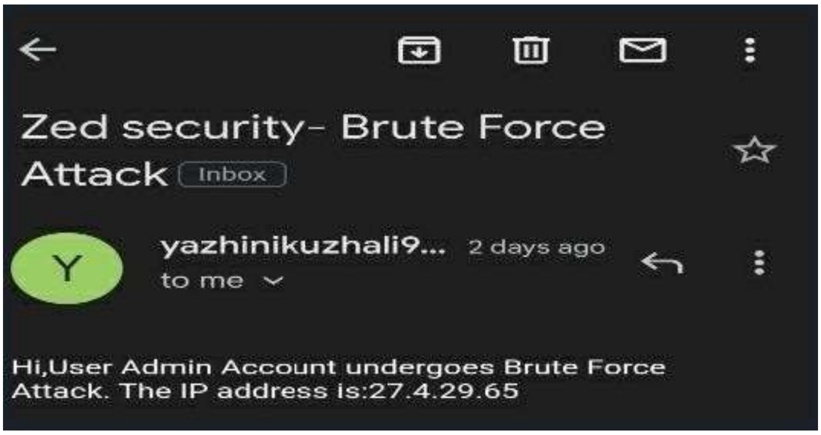
In contrast to conventional brute force attacks, which attempt to guess a password for a single account, password spraying employs a variety of strategies. It yet continues to use the widespread trial-and-error methodology that characterizes a brute force attack. Because a password spray assault tries password guesses on a variety of accounts until it finds a match, it is regarded as a brute force attack.. In this attack we proposed a set of only six attempt if user login the page ,when it cross the limit more than six ,it send the mail tothe admin.(while attempt (>6).

SOFTWARE USED: framework- Springboot, language - Html, Mysql, angular, java.

4. DIRECTORIES OF MODULES

4.1 MODULE 1: USER INTERFACE

The user login a web page by log in number attempt and after the password cracks by intruder send mail to an alert message to the admin.



Zed security- Brute Force Attack Inbox

yazhinikuzhali9... 2 days ago

to me

Hi,User Admin Account undergoes Brute Force Attack. The IP address is:27.4.29.65

4.2 MODULE 2: UNAUTHORIZED ACCESS

When the thirty party try to crack the user password, the attempts tried by the intruder get noticed and detected.

4.3 MODULE 3: ENHANCED METHOD

The method which the wronguser access will be detected only by the attempt logged in ,after the attempt goes more than six,get noticed and get alert message.

5. MODELLING AND ANALYSIS

5.1 SYSTEM ARCHITECTURE

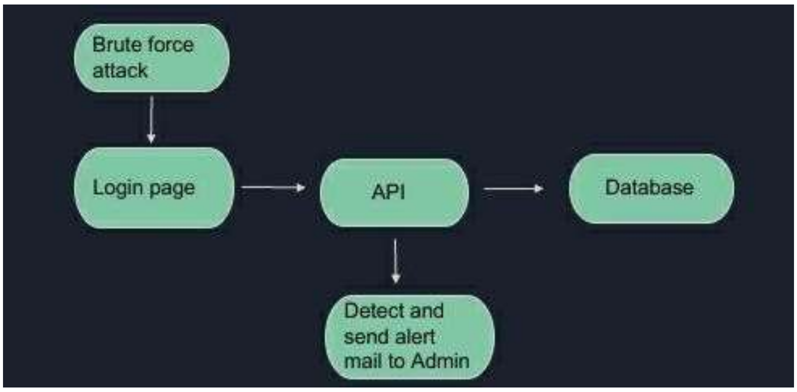


Figure 5.1 System Architecture

5.2 WORKFLOW DIAGRAM

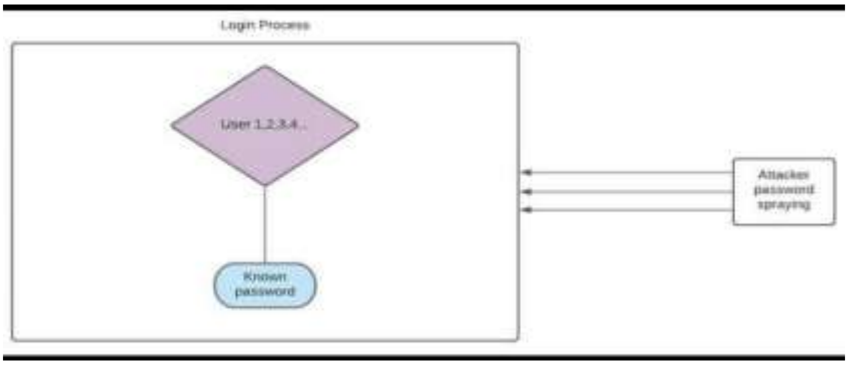


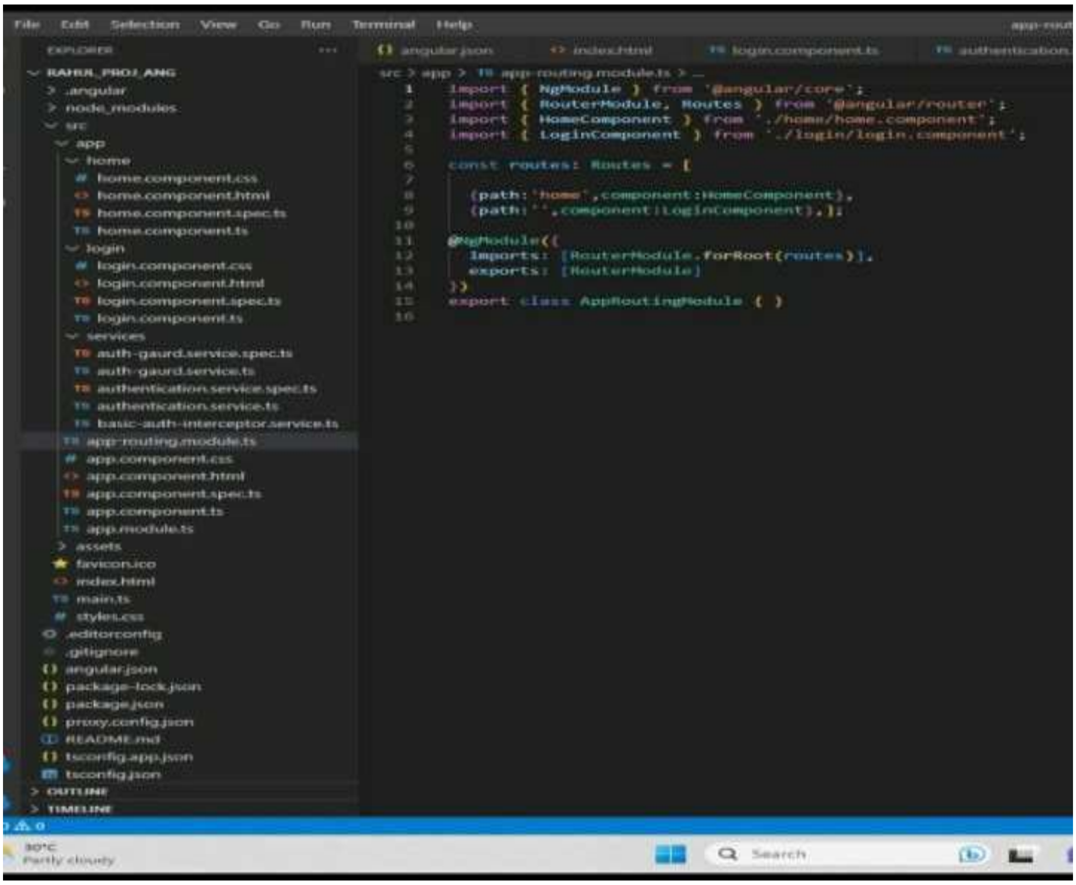
Figure 5.2 Workflow

6. RESULTS AND DISCUSSION

In this article, I won't discuss how to avoid this technique, but how to detect that it may be occurring in your environment. This will enable you take the best protection measures

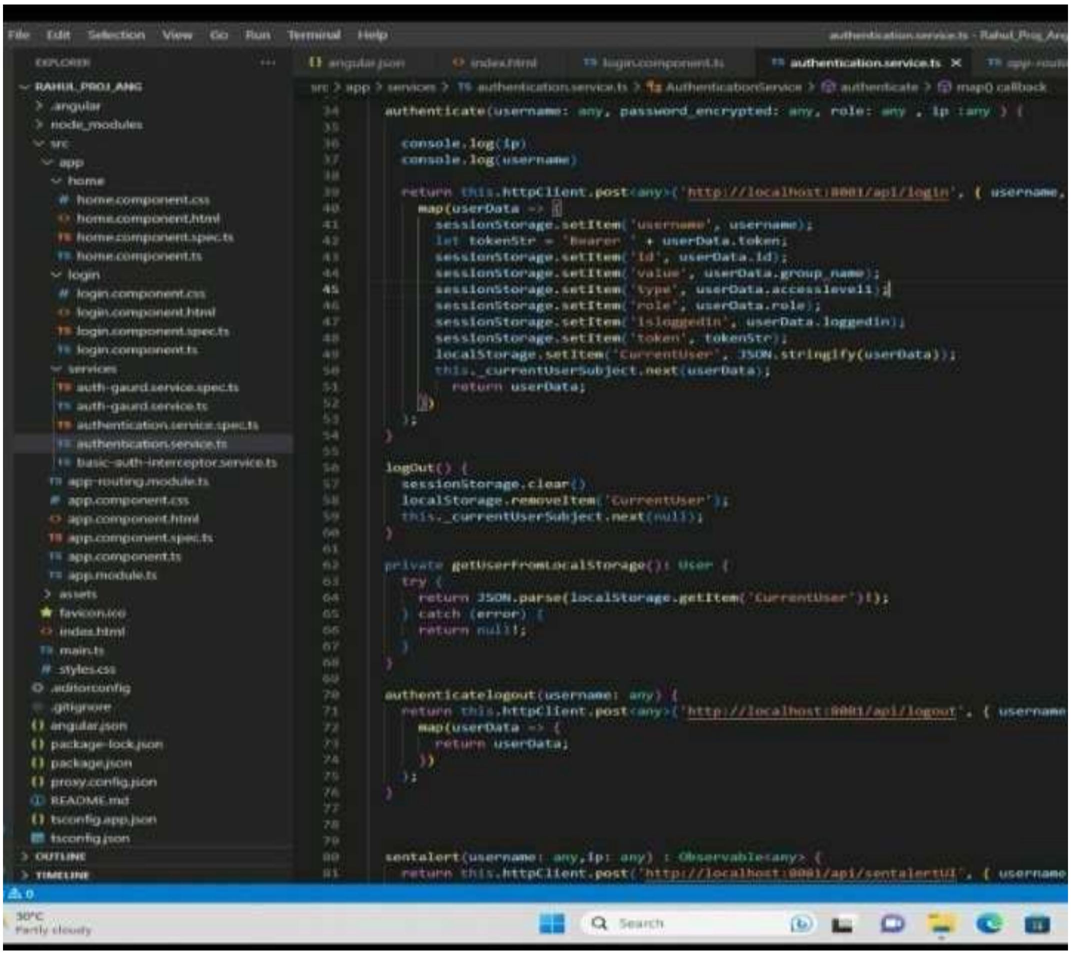
against a threat.this is to how much secure to the user can aware of these kind of various attacks that tools which are used by an intruder to access tha all data.

6.1 SCREENSHOTS



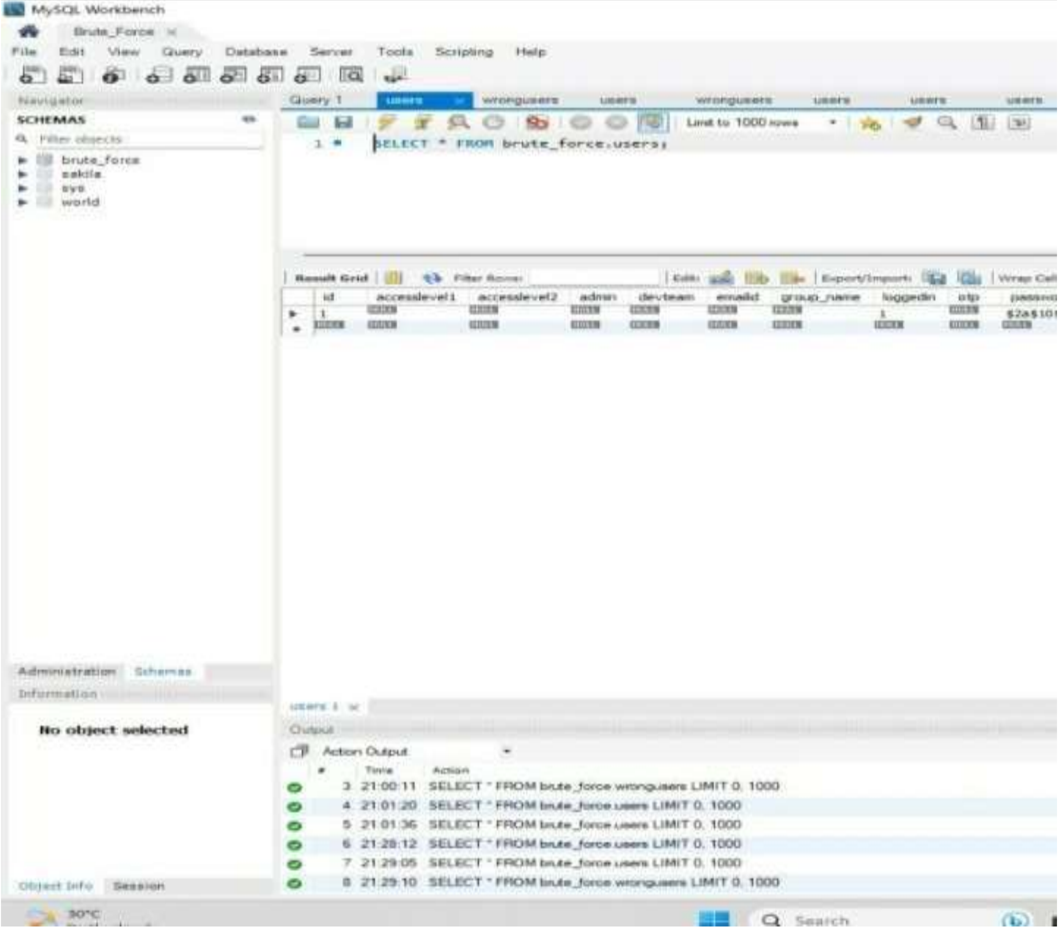
```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { HomeComponent } from './home/home.component';
import { LoginComponent } from './login/login.component';

const routes: Routes = [

  {path:'home',component:HomeComponent},
  {path:'',component:LoginComponent},];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule { }
```

Figure 6.1 Routing model


File Edit Selection View Go Run Terminal Help
authentication.service.ts - Rahul_Proj_Ang
EXPLORER
RAHUL_PROJ_ANG
angular.json
index.html
login.component.ts
authentication.service.ts
app-routin
src > app > services > authentication.service.ts > AuthenticationService > authenticate > map() callback
30°C
Partly cloudy
Search

```
authenticate(username: any, password_encrypted: any, role: any , ip :any ) {

  console.log(ip)
  console.log(username)

  return this.httpClient.post<any>('http://localhost:8081/api/login', { username,
    map(userData => {
      sessionStorage.setItem('username', username);
      let tokenStr = 'Bearer ' + userData.token;
      sessionStorage.setItem('id', userData.id);
      sessionStorage.setItem('value', userData.group_name);
      sessionStorage.setItem('type', userData.accesslevel1);
      sessionStorage.setItem('role', userData.role);
      sessionStorage.setItem('isloggedin', userData.loggedin);
      sessionStorage.setItem('token', tokenStr);
      localStorage.setItem('CurrentUser', JSON.stringify(userData));
      this._currentUserSubject.next(userData);
        return userData;
    })
  );
}

logOut() {
  sessionStorage.clear()
  localStorage.removeItem('CurrentUser');
  this._currentUserSubject.next(null);
}

private getUserFromLocalStorage(): User {
  try {
    return JSON.parse(localStorage.getItem('CurrentUser')!);
  } catch (error) {
    return null!;
  }
}

authenticatelogout(username: any) {
  return this.httpClient.post<any>('http://localhost:8081/api/logout', { username
    map(userData => {
      return userData;
    })
  );
}



sentalert(username: any,Ip: any) : Observable<any> {
  return this.httpClient.post('http://localhost:8081/api/sentalertUI', { username
```

Figure 6.2 Authentication Services



```
1  SELECT * FROM brute_force.users;
```

id	accesslevel1	accesslevel2	admin	devteam	emailid	group_name	loggedin	otp	passwo
1	NULL	NULL	NULL	NULL	NULL	NULL	1	NULL	$2a$10
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
#  Time      Action
3  21:00:11  SELECT * FROM brute_force.wrongusers LIMIT 0, 1000
4  21:01:20  SELECT * FROM brute_force.users LIMIT 0, 1000
5  21:01:36  SELECT * FROM brute_force.users LIMIT 0, 1000
6  21:28:12  SELECT * FROM brute_force.users LIMIT 0, 1000
7  21:29:05  SELECT * FROM brute_force.users LIMIT 0, 1000
8  21:29:10  SELECT * FROM brute_force.wrongusers LIMIT 0, 1000
```

Fig 6.3 Password stored database

7. CONCLUSION

We must advance alongside technology. Regarding identity management, there is no longer any advantage to continuing with the old practises. Going passwordless could be the solution your business needs to safeguard itself from a variety of other equally harmful cyberattacks in addition to password spraying..these kind of attacks will access all data from the user and may get account in trouble and in critical state by preventing and detecting using API by demonstrating password spraying detection.

8. REFERENCES

[1] Credential Spill Report" Shape Security. January 2017. p. 23. The most popular credential stuffing tool, Sentry MBA, uses "config" files for target websites that contain all the login sequence logic needed to automate login attempts.

[2] Montoro, Massimiliano (2005). "Cain & Abel User Manual: Brute-Force Password Cracker". oxid.it (defunct). Archived from the original on August 20,2013. Retrieved August 13, 2013.

[3] What Is Password Spraying? How to Stop Password Spraying Attacks". Bahadursingh, Roman (January19, 2020).

[4] A Distributed Algorithm for Brute Force Password Cracking on n Processors". doi:10.5281/zenodo.3612276.