Vulnerability Assessment and Systems Assurance Report

TuneStore

Shashank Mondrati

ITIS: 4221

February 2022

VULNERABILITY ASSESSMENT AND SYSTEM ASSURANCE

TABLE OF CONTENTS

1.0 GENERAL INFORMATION

1.1 Purpose

The objective of this TuneStore application assessment is to identify and analyze the vulnerabilities that are present in this TuneStore application. More specifically the assessment and penetration testing is to determine the security levels that are within the scope of the application. The vulnerabilities being discussed in this report will include use cases, and XSS vulnerabilities.

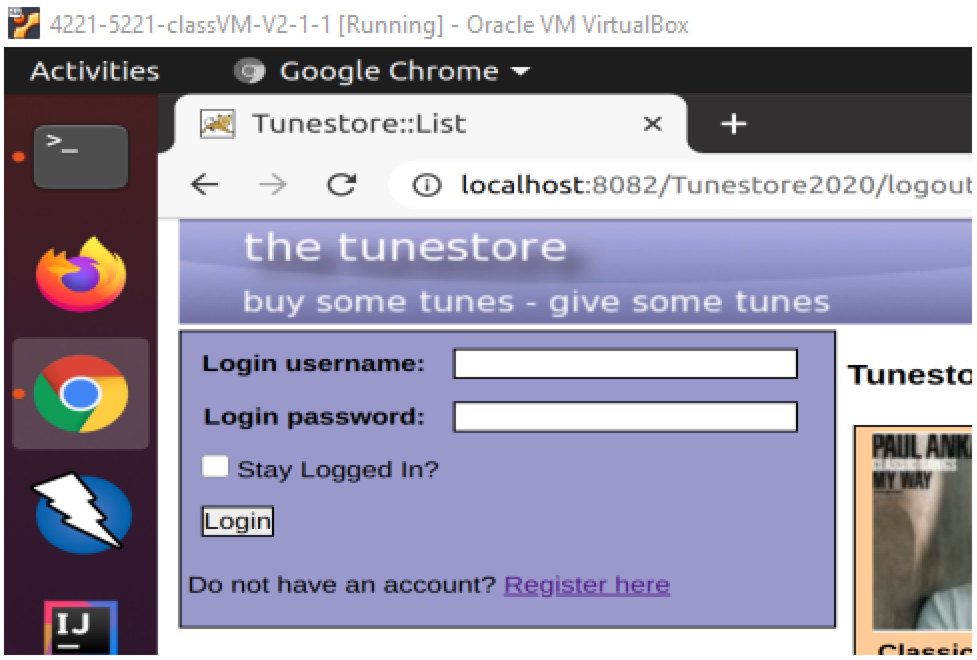
2.0 VULNERABILITIES DISCOVERED

2.1 SQL Injection

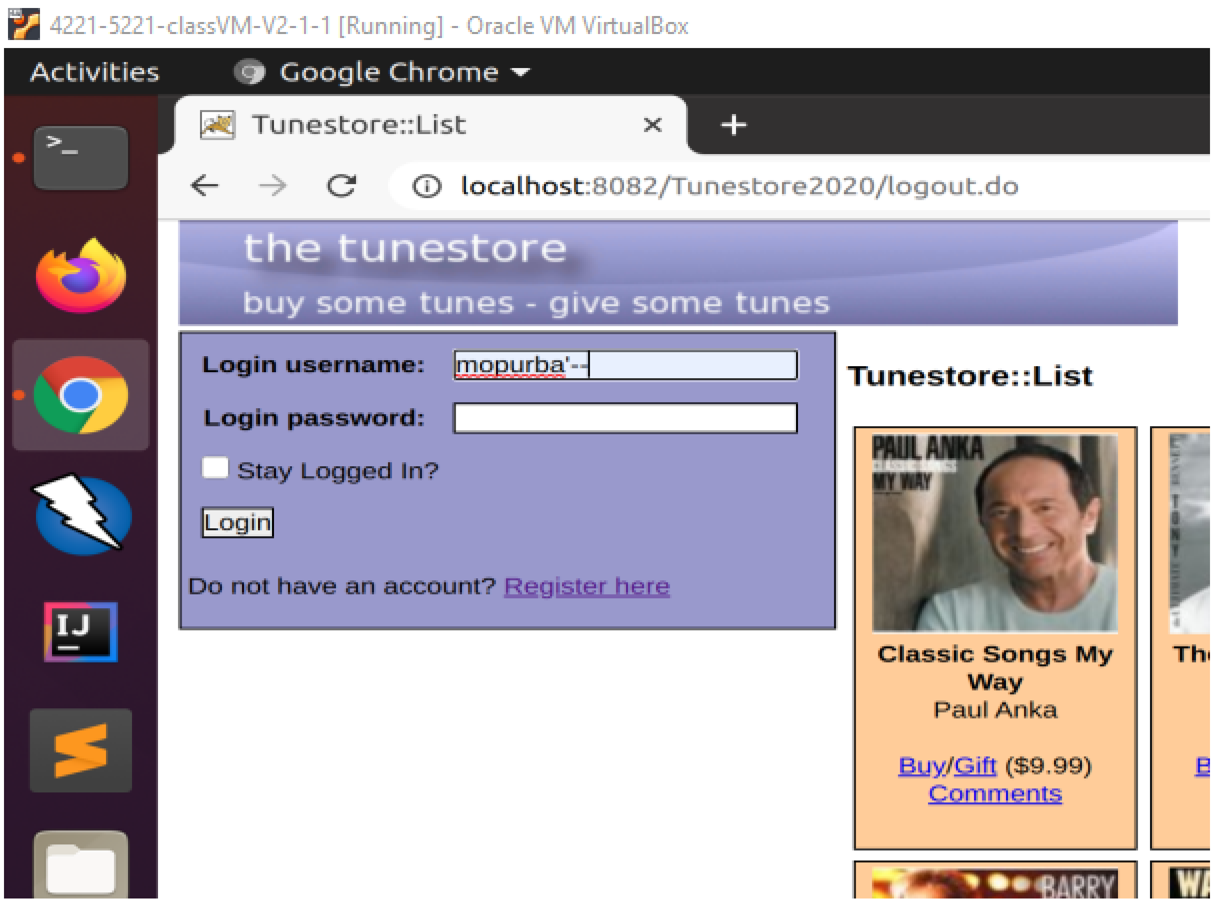
The **TuneStore** application is vulnerable to several SQL injection attacks. An SQL injection attack can occur when a user is asked to enter a normal username or sample credit card number, but instead gives anSQL statement that will unknowingly be run on the database server that is behind the web application. This gives an attacker the opportunity to execute malicious SQL statements on the database. For instance, using SQL injection, an attacker can add the amount they want to their account with the help of a credit card. Additionally, they could access, modify, or delete the data that is contained in the database. In the case of the **TuneStore** application, SQL injection could be used by an attacker to login as a random user without needing to know their credentials.

2.1 SQL Injection - Login as a random User

One example would be perfect in an instance where they can log into the application. By logging into the TuneStore application as a random user, this vulnerability exists in the logging in functionality on the login page of the application.The logging in functionality uses an SQL statement to check if the entered username and password inputs exist in the database. The attacker can easily login as a random user or as a specific user just by adding (**'--**) to the end of the user name, and accessing their information easily. Below is a screenshot of the login page, and where they can login as a random user in the TuneStore application.


4221-5221-classVM-V2-1-1 [Running] - Oracle VM VirtualBox
Activities
Google Chrome
Tunestore::List
localhost:8082/Tunestore2020/logout
the tunestore
buy some tunes - give some tunes
Login username:
Login password:
Stay Logged In?
Login
Do not have an account? Register here
Tunesto
PAUL ANK
MY WAY
Classic

An attacker is able to login with the method that is mentioned above, below is the screenshot of an attacker login using the method.


4221-5221-classVM-V2-1-1 [Running] - Oracle VM VirtualBox

localhost:8082/Tunestore2020/logout.do

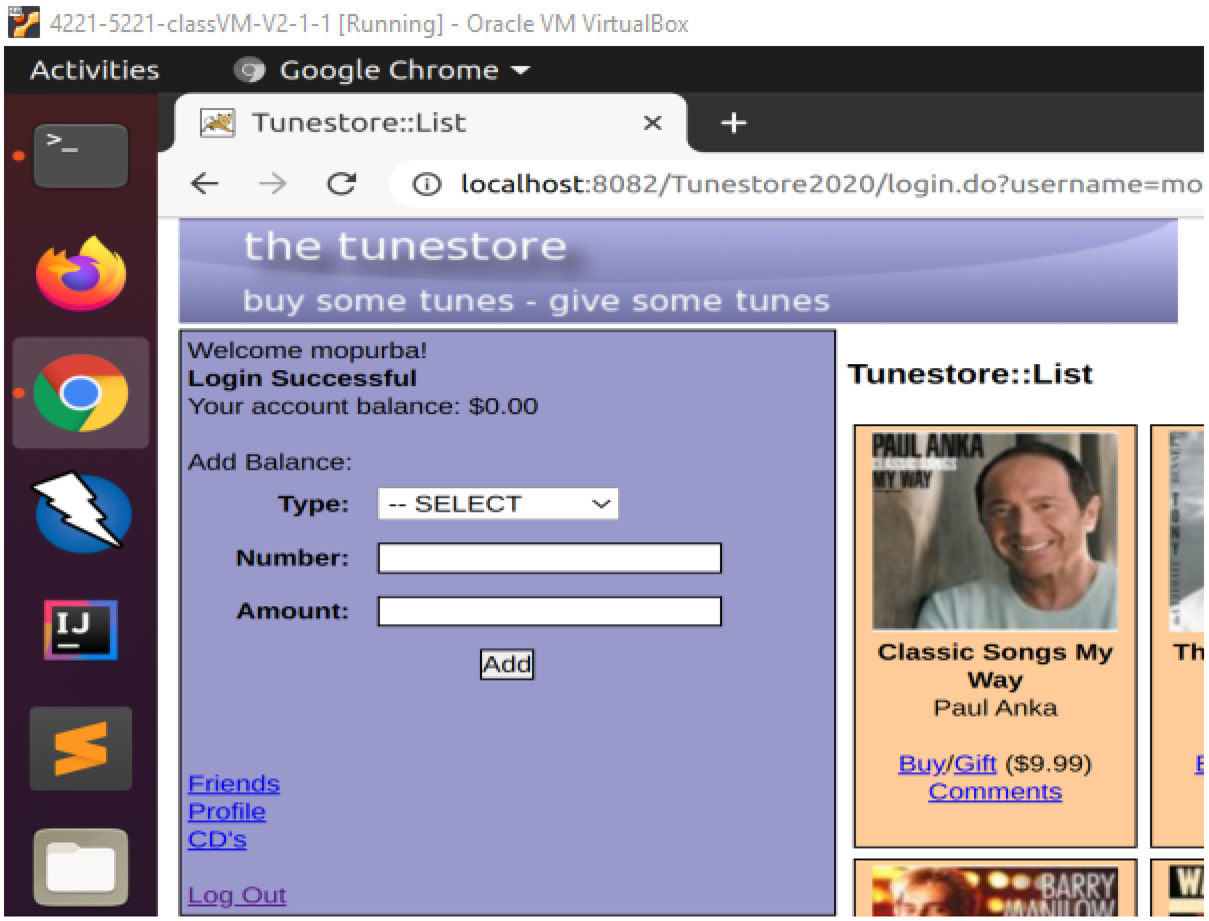
Login username: mopurba'--

Login password:

Stay Logged In?

Using these methods, the attacker can login into almost any user, access their credentials easily making it vulnerable to hack their information.

Below screenshot shows that they can login as a random user of the attacker's choice.


4221-5221-classVM-V2-1-1 [Running] - Oracle VM VirtualBox

Activities Google Chrome

Tunestore::List

localhost:8082/Tunestore2020/login.do?username=mo

the tunestore

buy some tunes - give some tunes

Welcome mopurba!
Login Successful
Your account balance: $0.00

Add Balance:

Type:	-- SELECT
Number:	
Amount:	

Add

Friends
Profile
CD's

Log Out

Tunestore::List

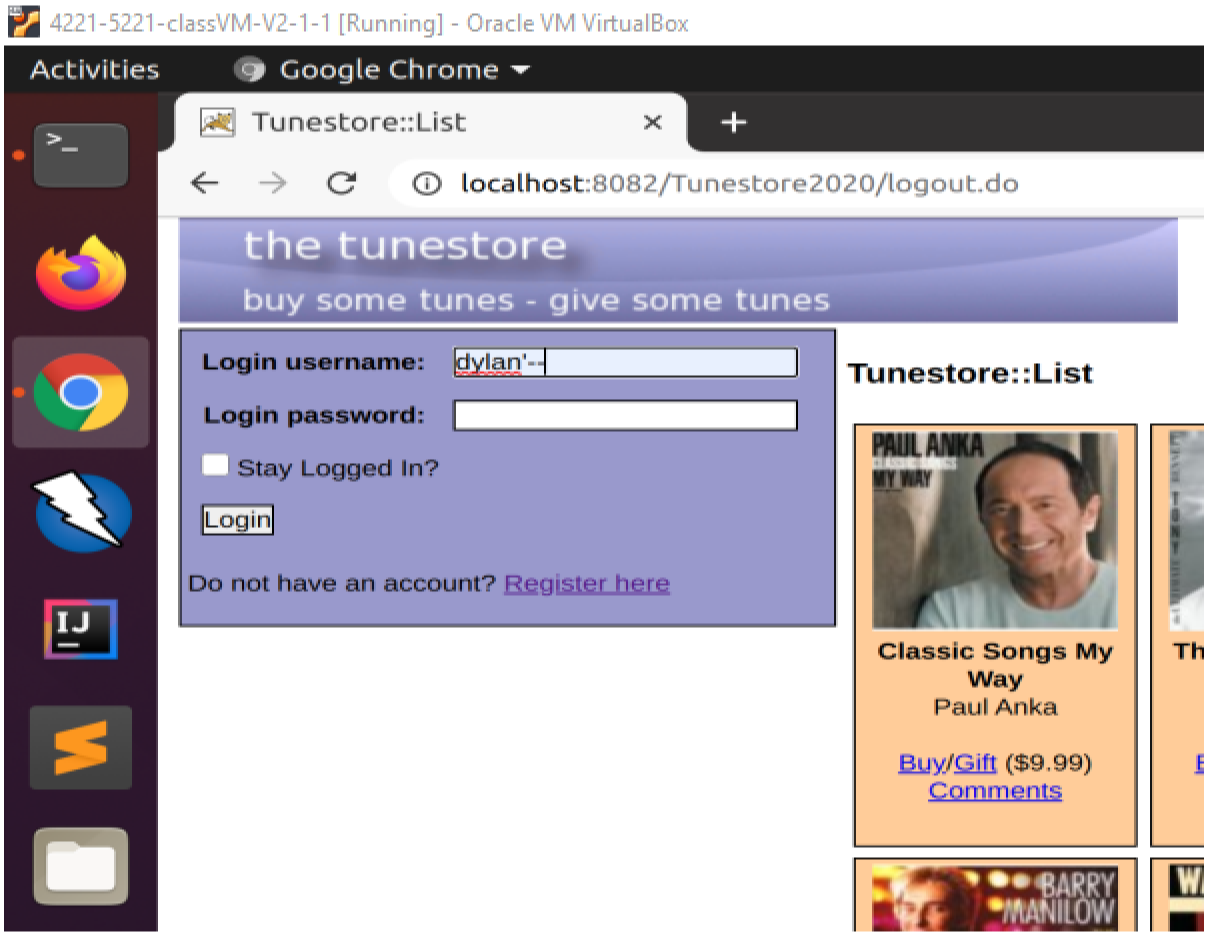
Classic Songs My Way
Paul Anka

Buy/Gift ($9.99)
Comments

As you can see, the attacker is able to login with the method, and they are able to add amount.

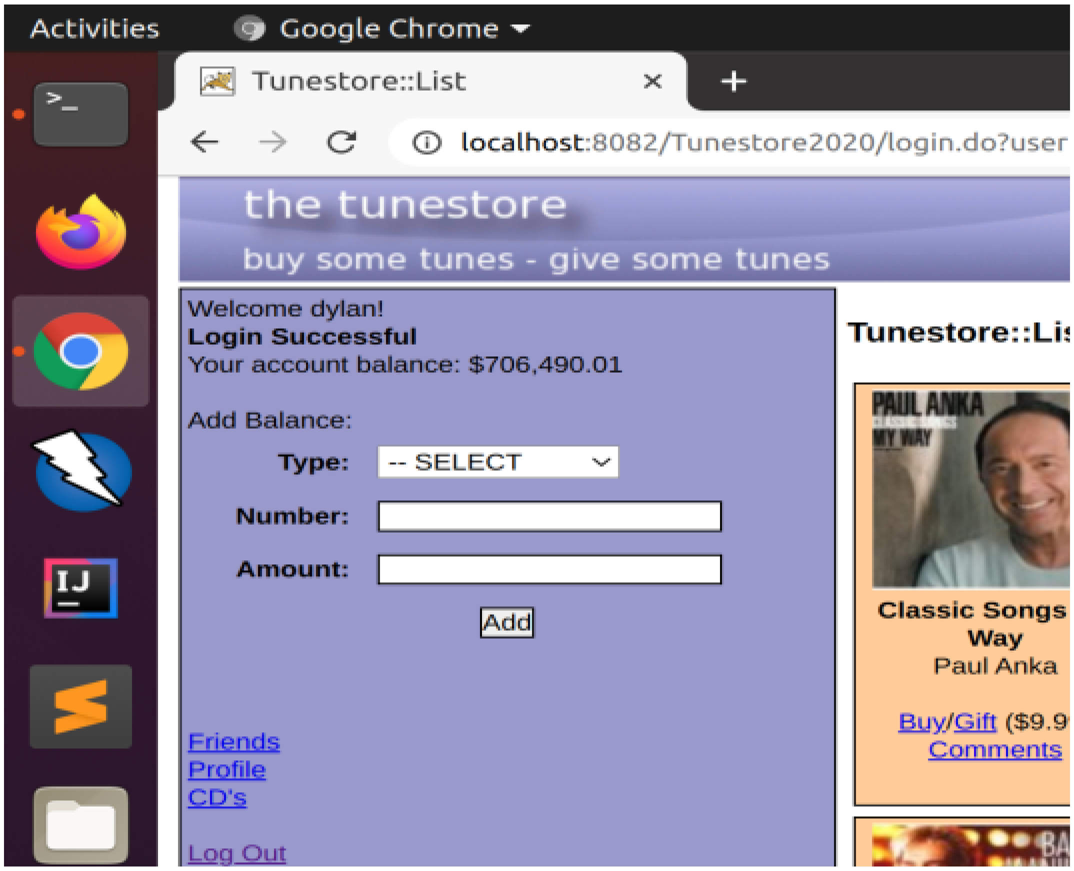
2.2 SQL Injection- Login as a specific user

Using the same methods, we can login to almost any application. For this instance, I asked my friend to create account in TuneStore, since I do not know the password, I was able to login to his account with ease, and access stored money in his account.
 Below is a screenshot of logging to his account and checking his funds in his account.


4221-5221-classVM-V2-1-1 [Running] - Oracle VM VirtualBox
Activities
Google Chrome
Tunestore::List
localhost:8082/Tunestore2020/logout.do
the tunestore
buy some tunes - give some tunes
Tunestore::List
Classic Songs My Way
Paul Anka
Buy/Gift ($9.99)
Comments

Login username: dylan'--
Login password:
Stay Logged In?
Login
Do not have an account? Register here

Above screenshot shows, adding ('--), we can login to the specific user account, and you can check the below screenshot to check his funds.

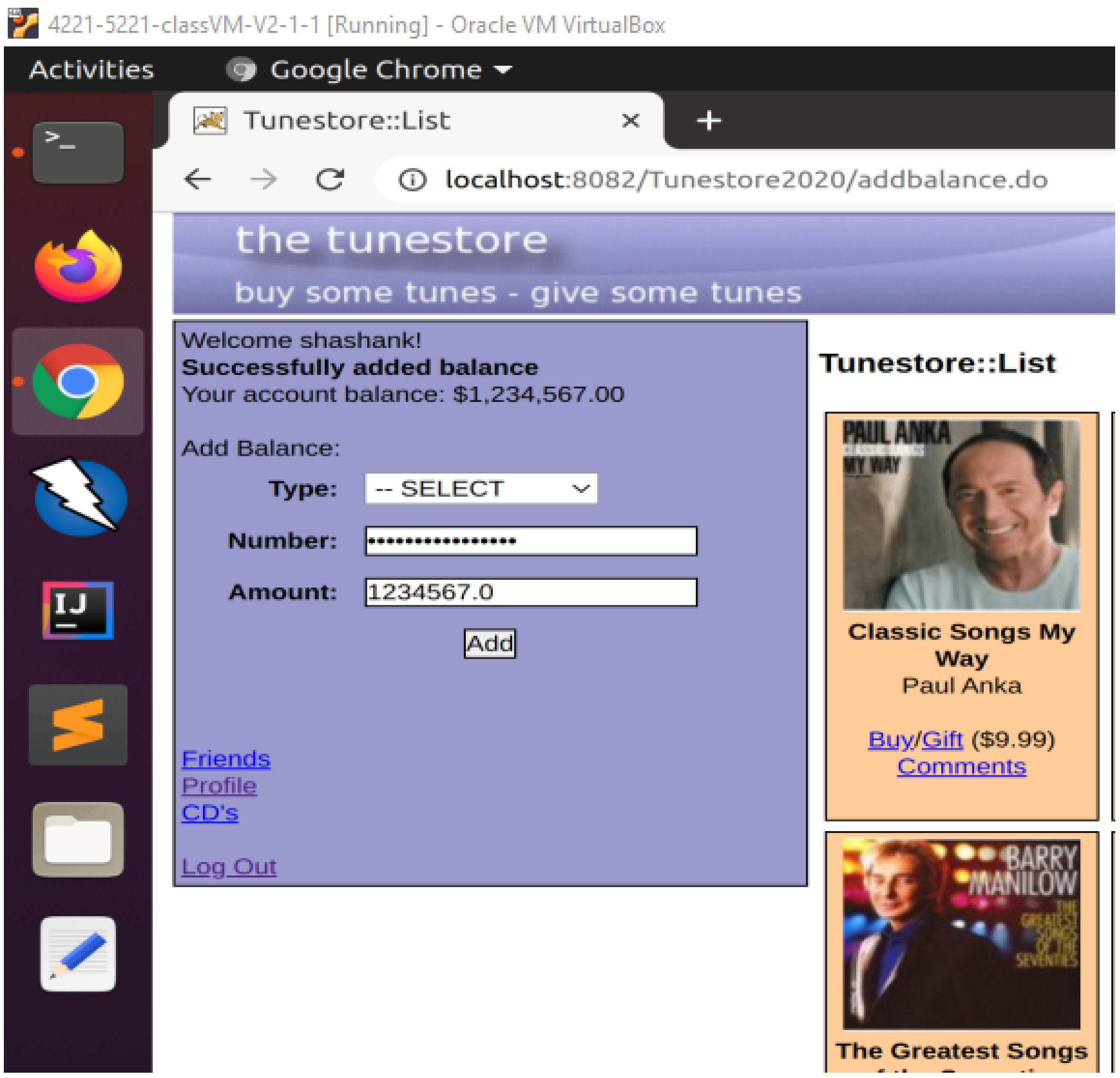

Activities
Google Chrome
Tunestore::List
localhost:8082/Tunestore2020/login.do?user
the tunestore
buy some tunes - give some tunes

Welcome dylan!
Login Successful
Your account balance: $706,490.01

Add Balance:

Type: -- SELECT
Number:
Amount:

As you can see above, there are over $700,000 in his account, the vulnerability is anyone can hack into anyone's account, and access their funds to buy anything in the TuneStore application, not to forget there's also another vulnerability where the attacker can add money without the help of adding their credit card numbers, the alternate way of doing this is the attacker can add money by putting the type as TYPE when adding balance.
Below is a screenshot where they can add funds without specifying that type of balance to add.


4221-5221-classVM-V2-1-1 [Running] - Oracle VM VirtualBox

localhost:8082/Tunestore2020/addbalance.do

the tunestore

buy some tunes - give some tunes

Welcome shashank!
Successfully added balance
Your account balance: $1,234,567.00

Add Balance:

Type:	-- SELECT
Number:	•••••••••••••••••
Amount:	1234567.0

Add

Friends
Profile
CD's

Log Out

Tunestore::List

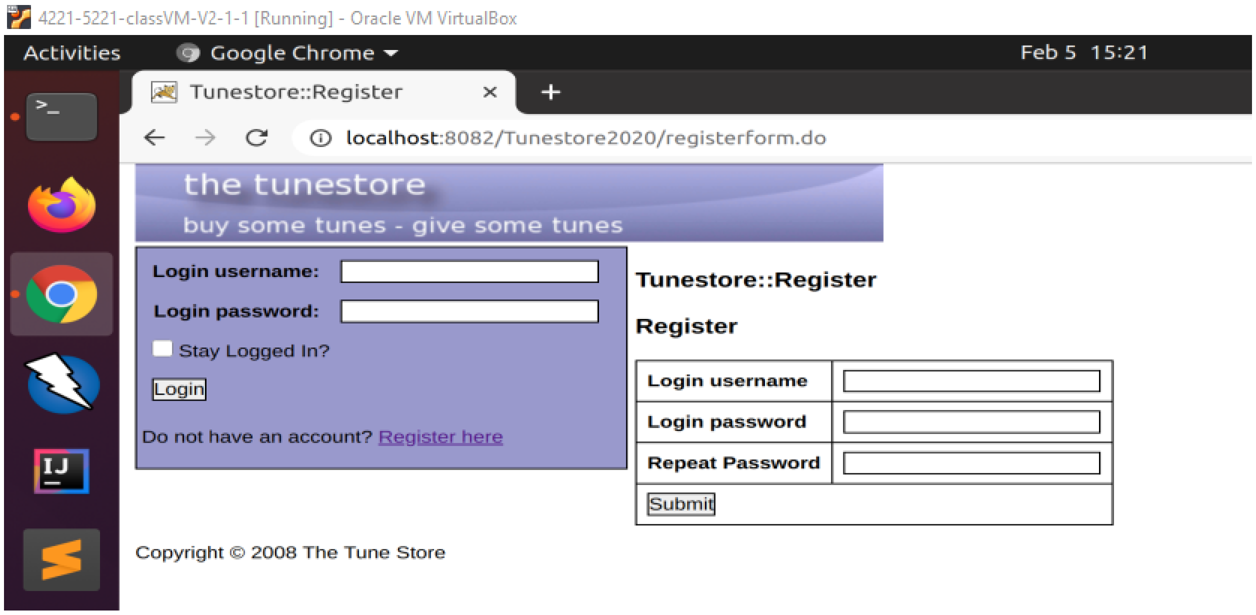
Classic Songs My Way
Paul Anka
Buy/Gift ($9.99)
Comments

The Greatest Songs

This screenshot shows they can add their money with 16 random numbers, and by putting the type of option to add money as TYPE to add money, once they are done adding money, the attacker is able to add money with ease. You can also see that when they click on ADD, on the top it says "Your new account balance : $1,234,567.00".

2.3. SQL Injection- Register a new user with lots of money in account w/out paying for it.

Users also have the ability to register an account, although there are many use cases in this TuneStore application, where as the users can login, logout, create an application, add their friends, buy something from the application. When trying to register for an account, the TuneStore application registry does not ask for very specific information, asks for basic information. Below is the screen shot, when asked to register for an account.



4221-5221-classVM-V2-1-1 [Running] - Oracle VM VirtualBox

Activities   Google Chrome   Feb 5 15:21

Tunestore::Register

localhost:8082/Tunestore2020/registerform.do

the tunestore

buy some tunes - give some tunes

Login username:

Login password:

Stay Logged In?

Login

Do not have an account? Register here

Tunestore::Register

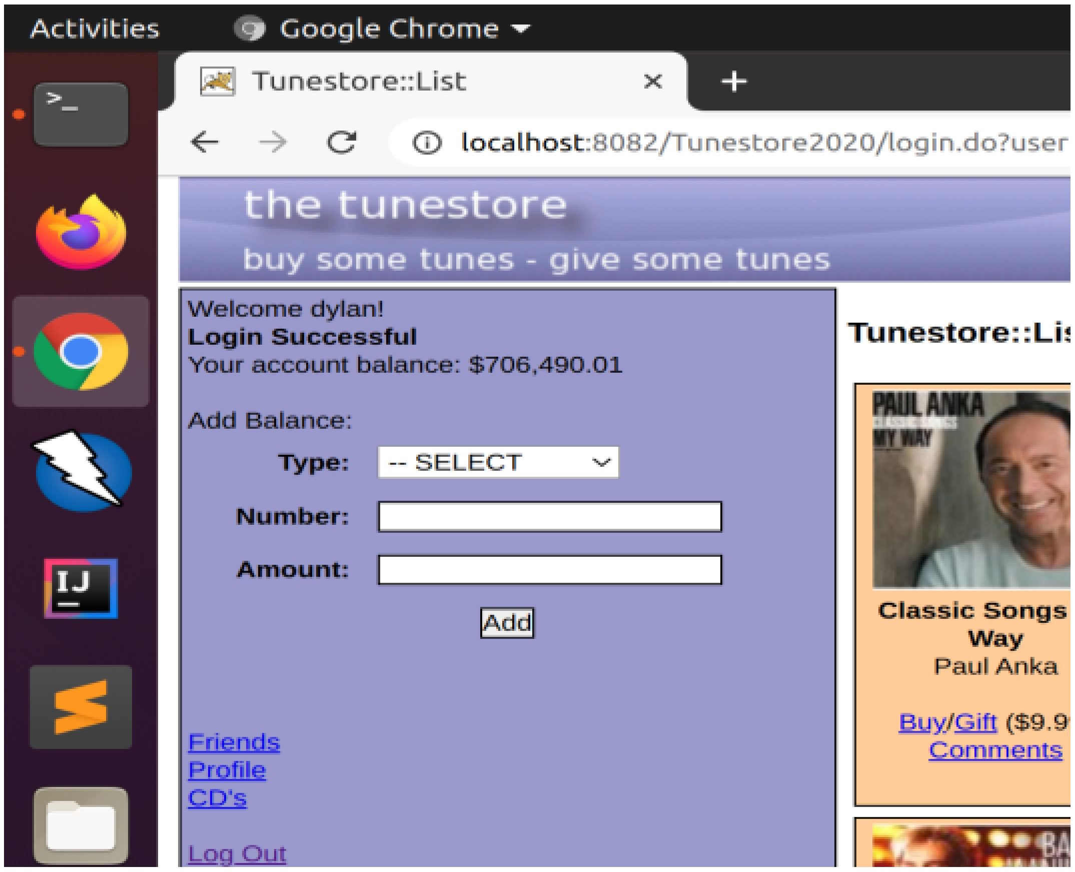
Register

Login username	
Login password	
Repeat Password	
Submit	

Copyright © 2008 The Tune Store

Above is the screenshot, where you can register for an account. I suggest if you wish to not get hacked use some extra symbols , because that prevents attackers from accessing the user's information.

I did some SQL injection another account, where I found there were a lot of money in their account. Below is a screenshot where it shows how much money they have.



Welcome dylan!
Login Successful
Your account balance: $706,490.01

Add Balance:

Type:	-- SELECT
Number:	
Amount:	

Add

Friends
Profile
CD's

Log Out

As you can see, the above screenshot shows the user how much money they have in their account.

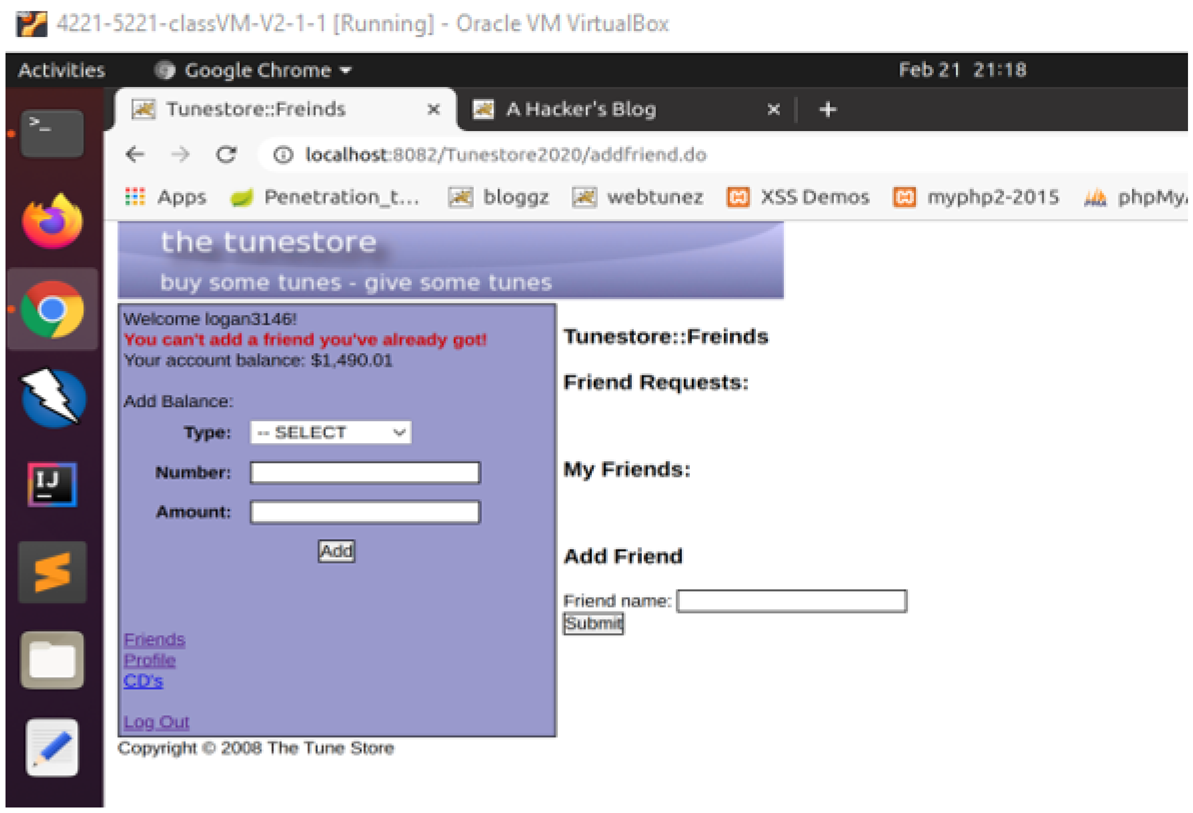
2.1 CSRF Injection

The **TuneStore** application is vulnerable to several CSRF injection attacks. A CSRF injection attack can occur when a user is asked to enter a normal username or sample credit card number, but instead gives anSQL statement that will unknowingly be run on the database server that is behind the web application. This gives an attacker the opportunity to execute malicious SQL statements on the database. For instance, using SQL injection, an attacker can add the amount they want to their account with the help of a credit card. Additionally, they could access, modify, or delete the data that is contained in the database. In the case of the **TuneStore** application, CSRF injection could be used by an attacker via a website that acts as a fake to login as a random user without needing to know their credentials, or give gifts, or changing passwords. Below is the application where I am conducting my CSRF attacks.

2.4 CSRF Attack: Adding a Friend

An easiest way to define CSRF attacks are, when a user tries to click on a website disguised as another website, and when the user decides to take an action on the disguised website, they are doing something related to the user. In the case, the user loses something personal to the user, like money from the bank account. In this instance, in TuneStore, when a user tries to add a friend, the user already opened a malware website disguised as another account. Additionally, they could alter changes, modify, or delete the data that is contained in the database. In the case of the TuneStore application, CSRF injection could be used by an attacker to add the attacker as a random user without needing who's on the other side.

Below is a screenshot of where I tried to add a friend named "Dylan" where the HTML pages contained the attack embedded in it added someone else named "Ohara2099", which is name of the user I created.



Welcome logan3146!
You can't add a friend you've already got!
Your account balance: $1,490.01

Add Balance:
Type: -- SELECT
Number:
Amount:
Add

Friends
Profile
CD's
Log Out

Copyright © 2008 The Tune Store

Tunestore::Freinds

Friend Requests:

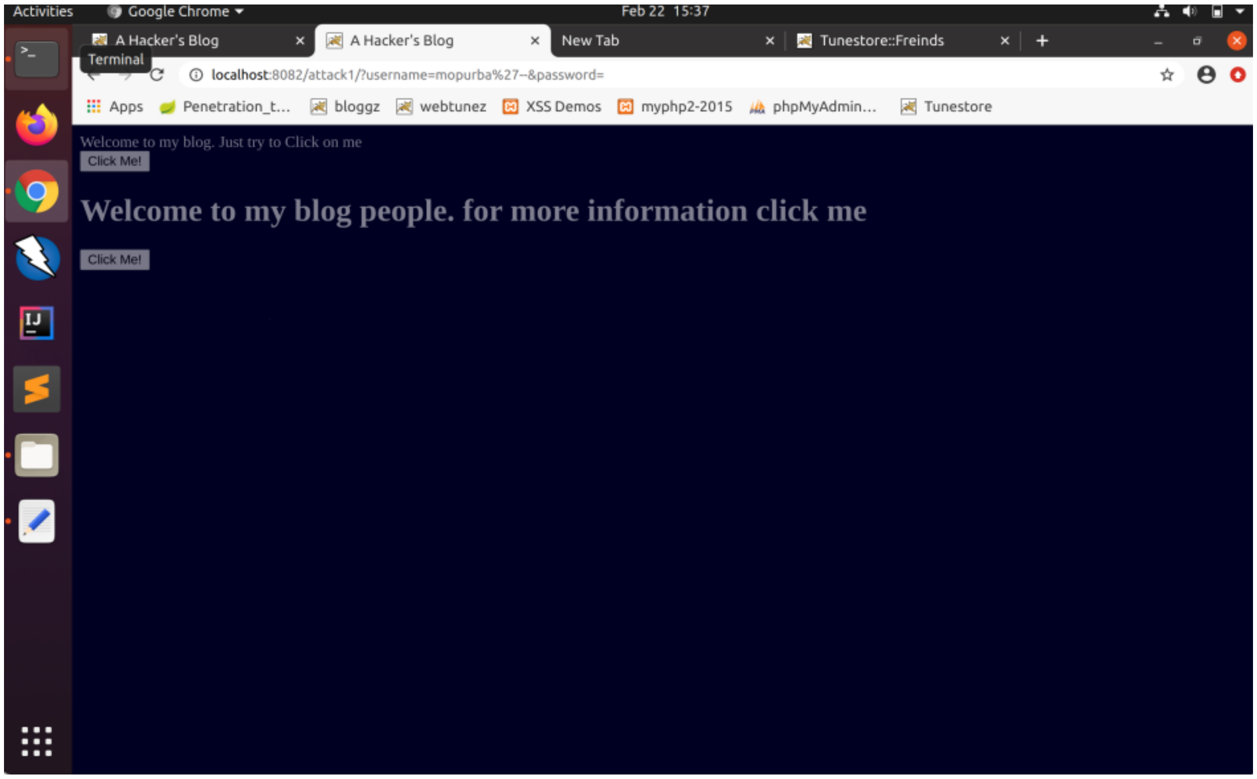
My Friends:

Add Friend

Friend name:
Submit

The person who created the webpage can add themselves to the friend list or someone the user doesn't know to the victim's friend list.

Below is the attack page where I will be doing attacks on the use cases in my table of contents.

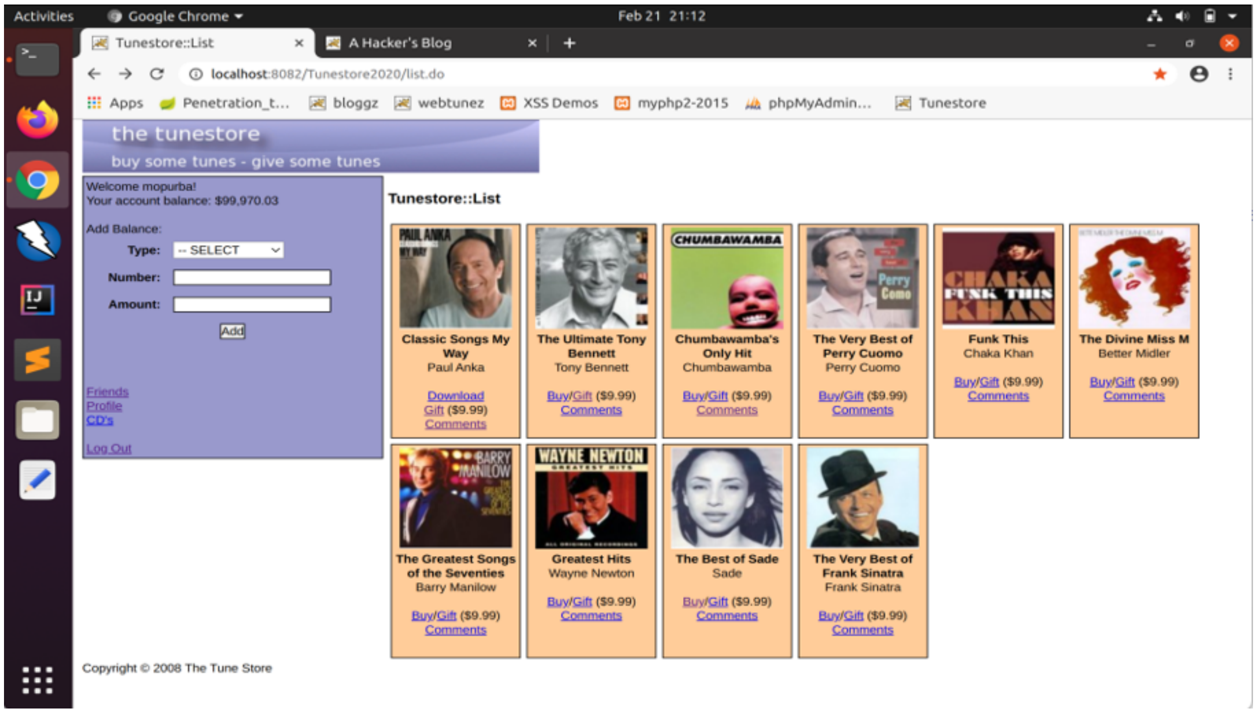


When they reload the page, and check the friends list, they should see the person added to the list.

2.5 CSRF Attack: Give Gift

The TuneStore application has a feature where the user can send gifts. The user can choose from a variety of gifts from the song list from the authors. They can buy the gift with a certified amount starting from $9.99. The user can click on Buy gift, and when they do they can keep the gift or give the gift to a person they know. In this instance, when the user decides to buy a gift, it can be any gift, but the difference is when the user already had a disguised website opened up in another tab. When the user click on send gift to a person, it sends to another person where the user doesn't know who the person is. End result is the user lost the money he bought the gift from, and the user doesn't know who the person received the gift from. This is the Cross Site where the gift is sent to another person.

Below is a screenshot of the application which shows what kind of gifts the users can download by buying, and gift it to other users.



the tunestore
buy some tunes - give some tunes

Welcome mopurba!
Your account balance: $99,970.03

Add Balance:
Type: -- SELECT
Number:
Amount:
Add

Friends
Profile
CD's

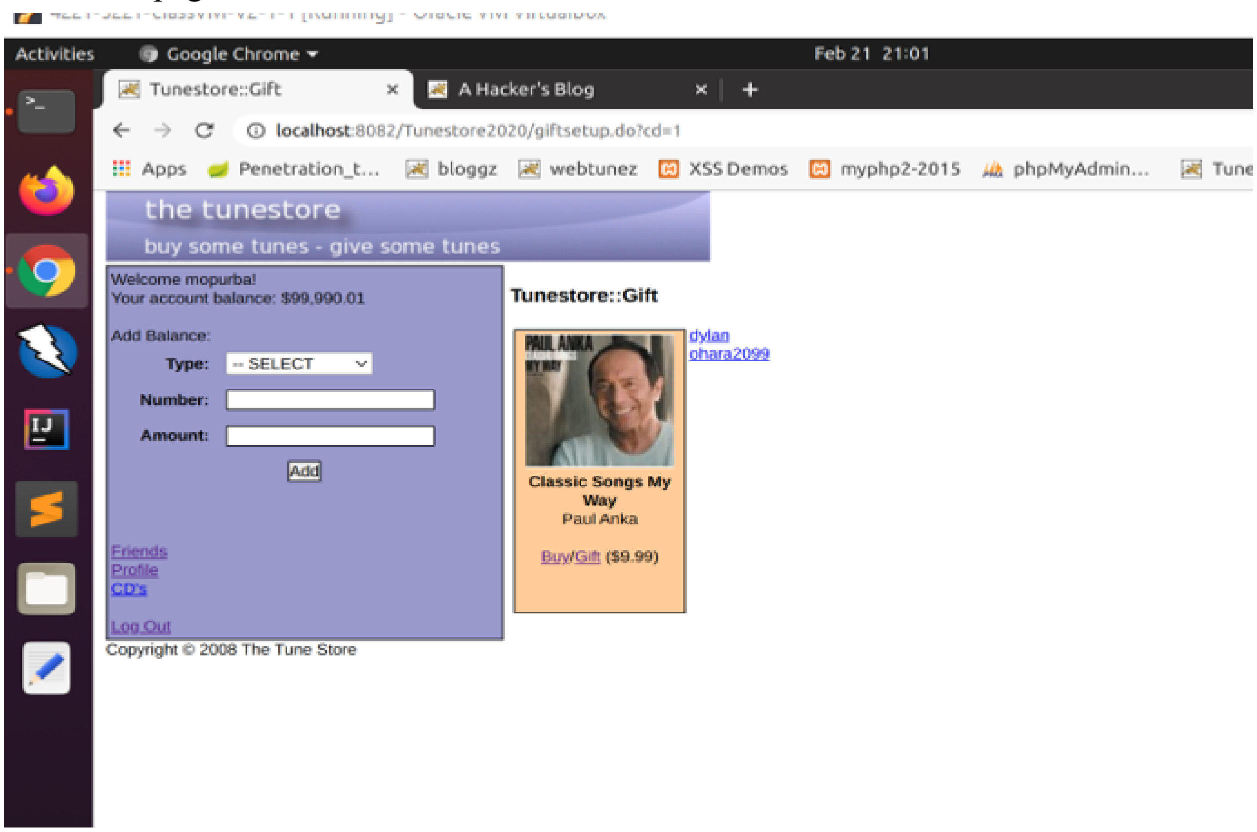
Log Out

Tunestore::List

Classic Songs My Way Paul Anka Download Gift ($9.99) Comments	The Ultimate Tony Bennett Tony Bennett Buy/Gift ($9.99) Comments	Chumbawamba's Only Hit Chumbawamba Buy/Gift ($9.99) Comments	The Very Best of Perry Cuomo Perry Cuomo Buy/Gift ($9.99) Comments	Funk This Chaka Khan Buy/Gift ($9.99) Comments	The Divine Miss M Better Midler Buy/Gift ($9.99) Comments
The Greatest Songs of the Seventies Barry Manilow Buy/Gift ($9.99) Comments	Greatest Hits Wayne Newton Buy/Gift ($9.99) Comments	The Best of Sade Sade Buy/Gift ($9.99) Comments	The Very Best of Frank Sinatra Frank Sinatra Buy/Gift ($9.99) Comments		

Copyright © 2008 The Tune Store

Below is a screenshot of the test where I sent a gift to my account "Ohara2099", where the HTML pages is contained with the attack code embedded in it.



the tunestore
buy some tunes - give some tunes

Welcome mopurba!
Your account balance: $99,990.01

Add Balance:
Type: -- SELECT
Number:
Amount:
Add

Friends
Profile
CD's

Log Out
Copyright © 2008 The Tune Store

Tunestore::Gift

Classic Songs My Way
Paul Anka
Buy/Gift ($9.99)

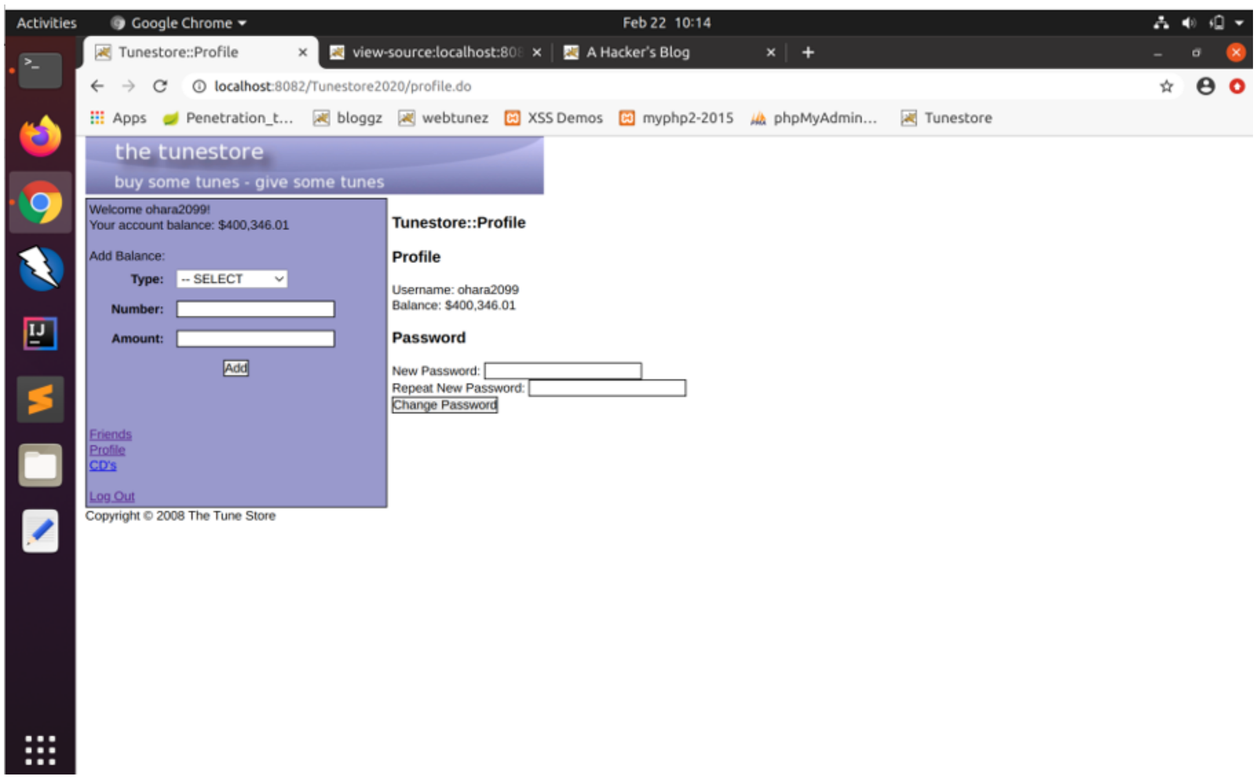
dylan
ohara2099

who also created the website can send the gift to the person they intended to, so that the user can decrease the money in their account, and send gifts to unknown users with CSRF attacks.

2.6 CSRF Attack Change Password

Changing passwords works in instances I have mentioned above, when the user already had a CSRF disguised website opened up, the website fills out the malware information on the user's webpages. For example, the user clicked on a malware website asking the user to enter their credentials, but in reality the malware website is changing something on the user's real page, like we can say changing passwords, because on the disguised website most of the form filling elements are set to hidden, or display to none. I'm pretty sure that the attacker already has the user' username, with the password change attack they can access the victim's password easily.

Below is a screenshot where the user can change the password, it does not ask for any specific information like easy questions to get back into. Just the user re-entering the new password and entering it again. But the attacking website fills out this information just by opening the website, because the attack code is embedded in the HTML in it.



Welcome ohara2099!
Your account balance: $400,346.01

Add Balance:
Type: -- SELECT
Number:
Amount:

Tunestore::Profile

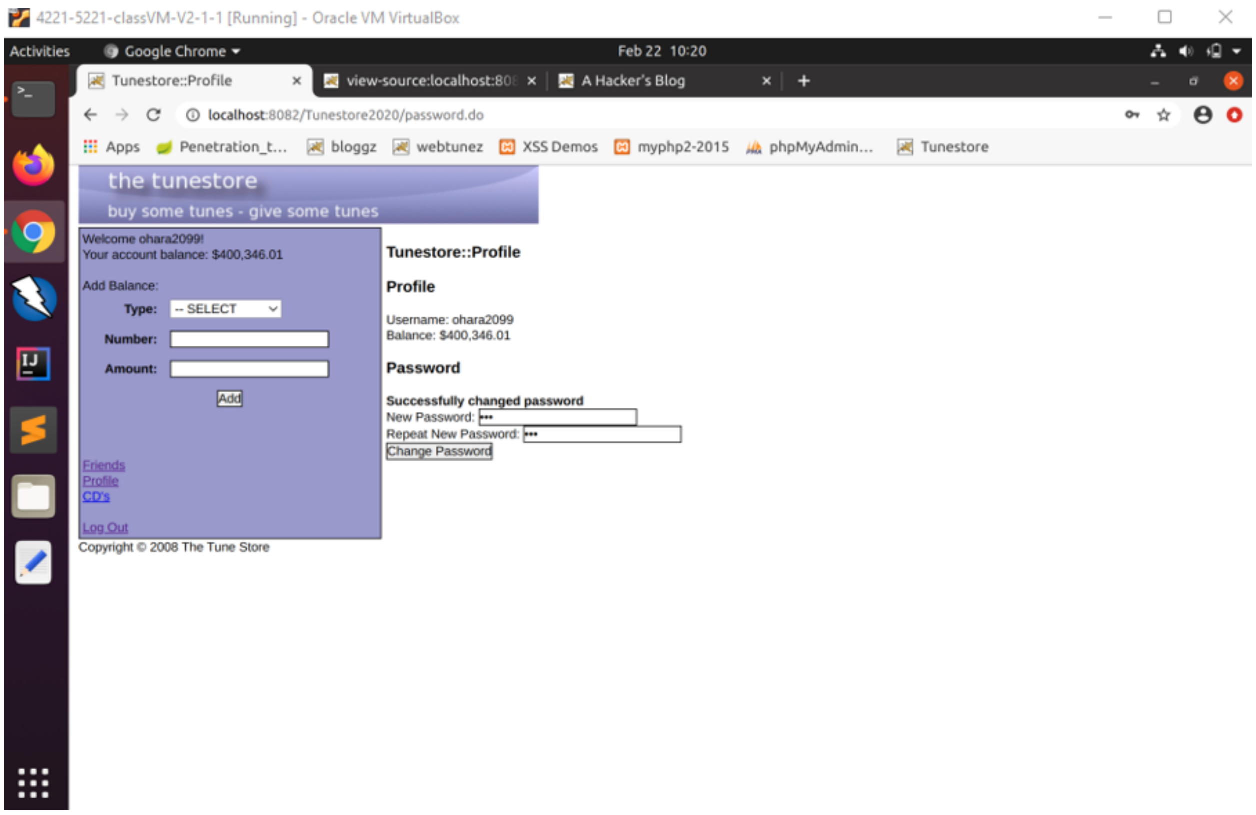
Profile

Username: ohara2099
Balance: $400,346.01

Password

New Password:
Repeat New Password:

Copyright © 2008 The Tune Store



Welcome ohara2099!
Your account balance: $400,346.01

Add Balance:

Type: -- SELECT
Number:
Amount:
Add

Friends
Profile
CD's

Log Out

Copyright © 2008 The Tune Store

Tunestore::Profile

Profile

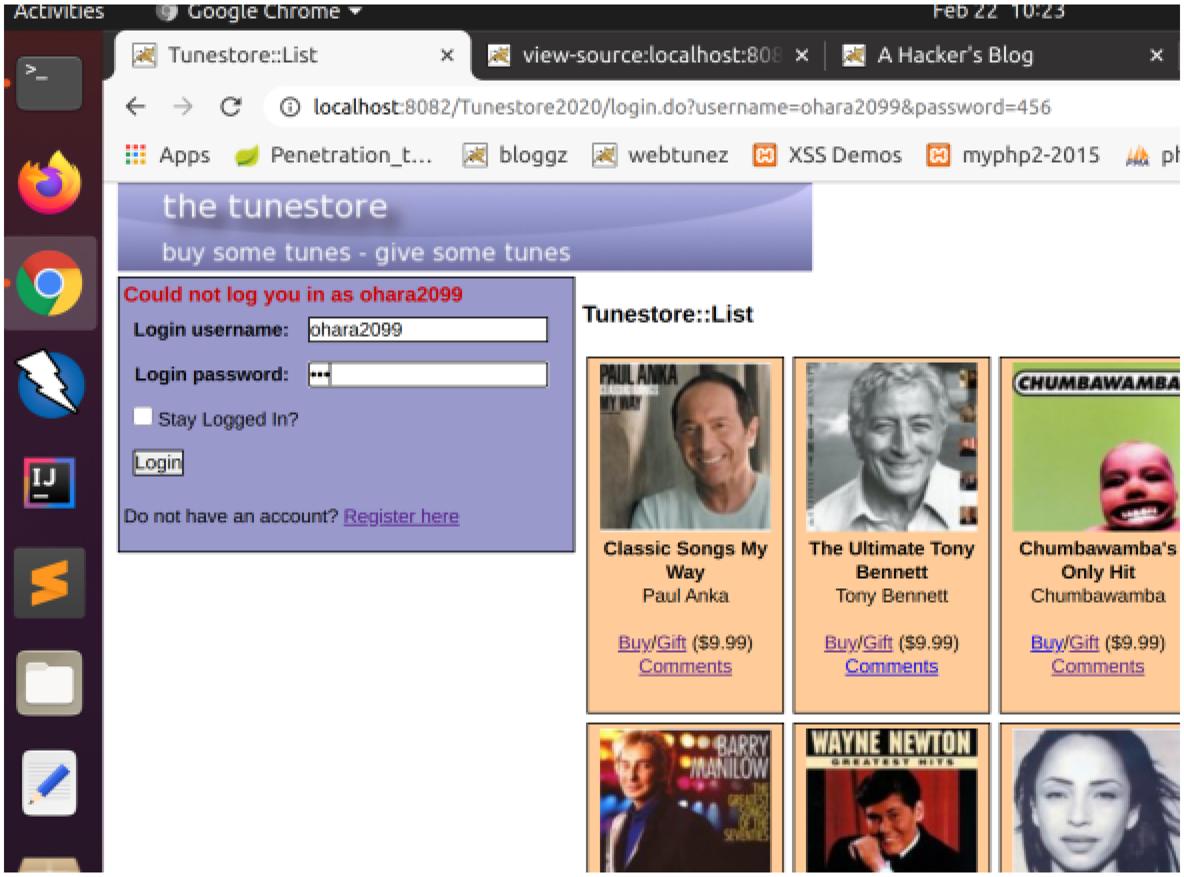
Username: ohara2099
Balance: $400,346.01

Password

Successfully changed password
New Password: •••
Repeat New Password: •••
Change Password

Above is the screenshot where the password is changed successfully with the attacking website right next to the TuneStore tab. The password in the HTML is "123", but the password I changed to is "456". When I tried to login as the same user, it says the password entered is invalid.

Below is the screenshot of the application of the changed password with CSRF changed password attack.



localhost:8082/Tunestore2020/login.do?username=ohara2099&password=456

Could not log you in as ohara2099

Login username: ohara2099

Login password: •••

Stay Logged In?

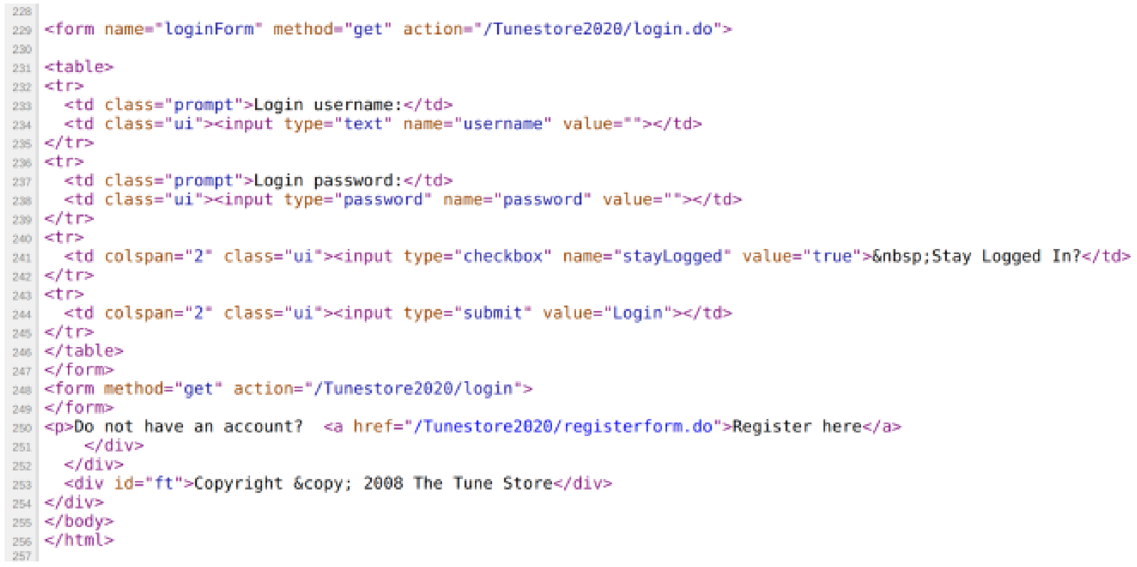
Login

Do not have an account? Register here

2.7 Broken Access Control Vulnerability.

Broken Access Control vulnerability, an easiest way to describe is where a vulnerability is present within the application, and also very hard to find the glitch in the application, also the user has access to it. In Layman's words the user has access to a place in which he's not supposed to have access to. An example in this instance would be, acting as an user without being able to log into the application or acting as an admin to the application. With this, they can manipulate the metadata such as replaying, or tampering with the application's JWT.

Below is the screenshot of the application where I found a broken access control vulnerability.



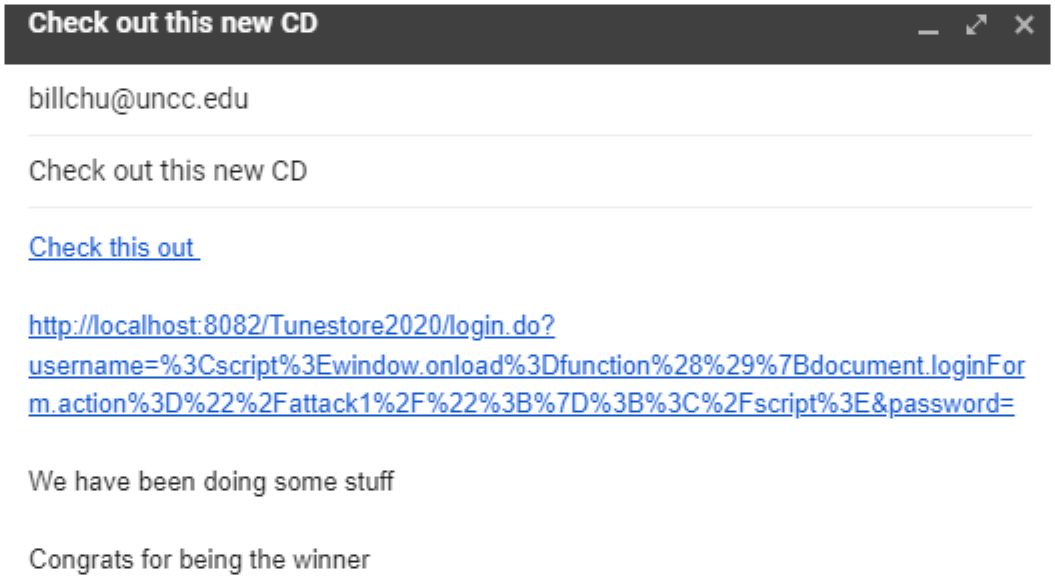
```
<form name="loginForm" method="get" action="/Tunestore2020/login.do">

<table>
<tr>
  <td class="prompt">Login username:</td>
  <td class="ui"><input type="text" name="username" value=""></td>
</tr>
<tr>
  <td class="prompt">Login password:</td>
  <td class="ui"><input type="password" name="password" value=""></td>
</tr>
<tr>
  <td colspan="2" class="ui"><input type="checkbox" name="stayLogged" value="true">&nbsp;Stay Logged In?</td>
</tr>
<tr>
  <td colspan="2" class="ui"><input type="submit" value="Login"></td>
</tr>
</table>
</form>
<form method="get" action="/Tunestore2020/login">
</form>
<p>Do not have an account?  <a href="/Tunestore2020/registerform.do">Register here</a>
    </div>
  </div>
  <div id="ft">Copyright &copy; 2008 The Tune Store</div>
</div>
</body>
</html>
```

You can find this by inspecting the source page of the TuneStore application. For this instance I found mine on the login page, when you scroll down to form where user can submit their credentials, they can see the names of the fields and values. If the attacker gets ahold of this information they can easily store the victim's user credentials. I'm thinkin that that sort of information shouldn't be public to users.

2.8 XSS Attack Exploit

To harvest another user's credentials, we just need an attack code run. In this instance I have run my attack code in my VM and embedded inside a link. So when the user decides to enter their credentials, sothat the attacker can know what they entered.

Below is a screenshot of the attack email where the attacker sent an email to the victim embedded in it.



Check out this new CD

billchu@uncc.edu

Check out this new CD

Check this out

http://localhost:8082/Tunestore2020/login.do?username=%3Cscript%3Ewindow.onload%3Dfunction%28%29%7Bdocument.loginForm.action%3D%22%2Fattack1%2F%22%3B%7D%3B%3C%2Fscript%3E&password=

We have been doing some stuff

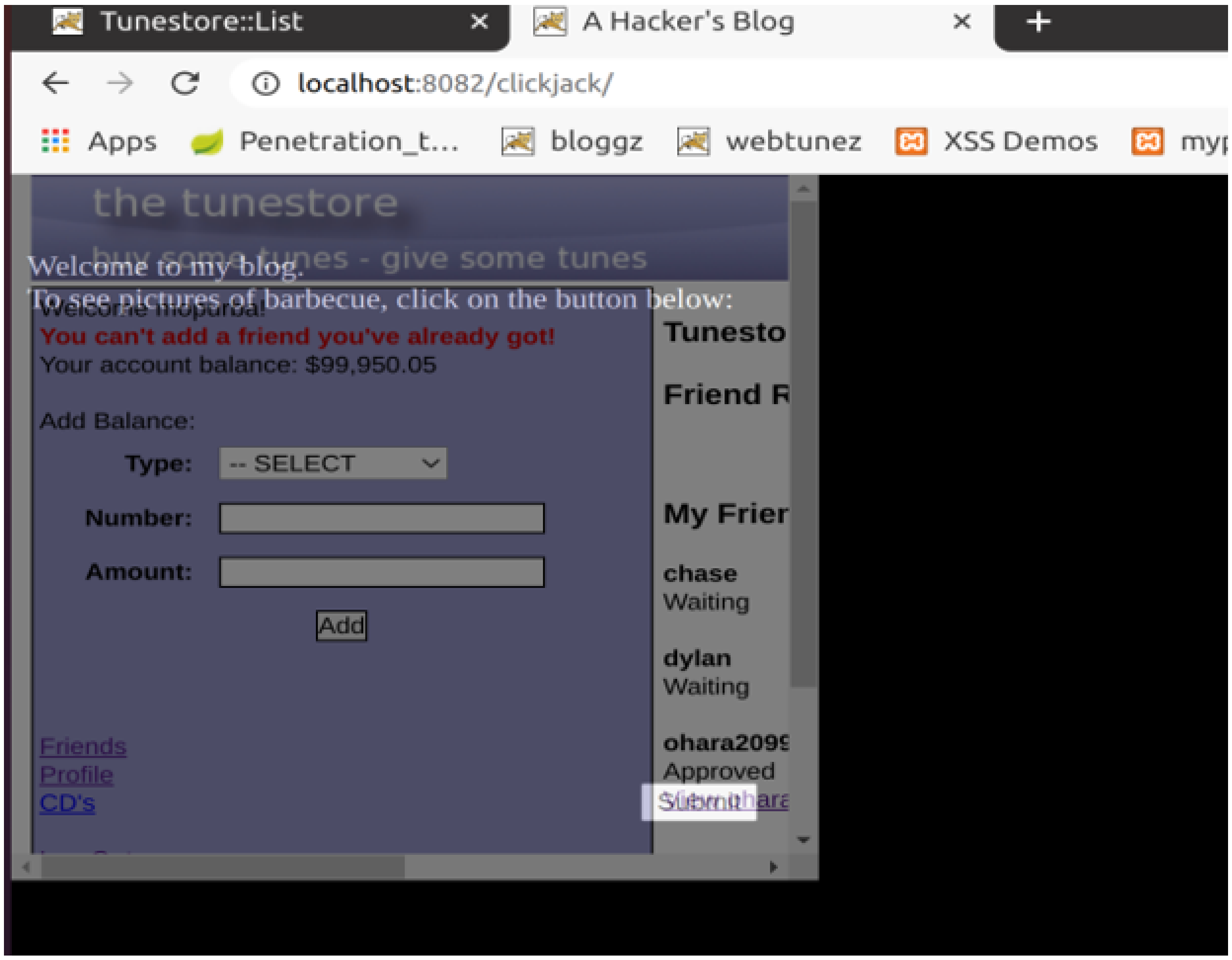
Congrats for being the winner

Above is the screenshot. When the recipient gets the email and when they are prompted to enter their credentials, they are leaking the credentials to the person who sent the email.

2.9 CSRF: ClickJacking: Friend

An easy way to to explain this would be when a user tries to login to a page they are forced to enter another page.

Below is the screenshot where the user was click jacked into buying something when they logged into the application.



Welcome to my blog.

To see pictures of barbecue, click on the button below:

Welcome mopurba!

You can't add a friend you've already got!

Your account balance: $99,950.05

Add Balance:

Type: -- SELECT

Number:

Amount:

Tunesto

Friend R

My Frier

chase
Waiting

dylan
Waiting

ohara209
Approved

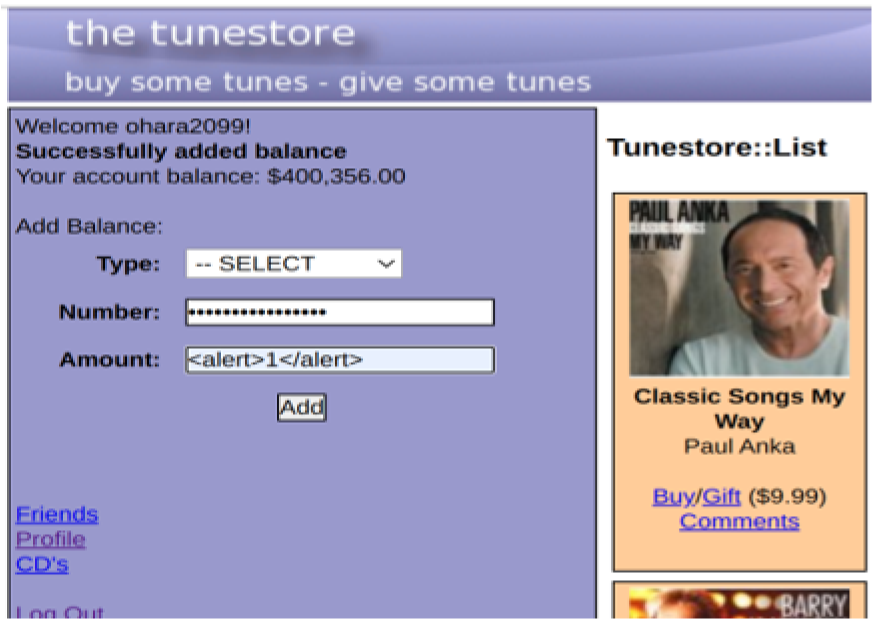
As you can see above the user is already logged in and you can see their balance, and to the right their friends. The way click jack works is when the user clicks on a link, a malware link they are guided to another link. This user decided to add a friend he does not know, with the code in the attack code. The user was was forced to add a friend he did not know.

3.0. XSS Vulnerabilities

What is XSS Vulnerability? We can define it as a common attack vector that hackers use by injecting malicious code where they can find a vulnerability in the web application. XSS differs from other web attack vectors (e.g., SQL injections), in that it does not directly target the application itself. Instead, the users of the web application are the ones at risk.

3.1. Stored XSS Vulnerability

We can define a Sstored XSS vulnerability as when a perpetrator located a vulnerability in the application, and they inject malicious code into the field to steal valuable information. They can insert malicious code in the comment field or user input field. Below is a screenshot showing Stored XSS vulnerability. That can be used to steal valuable information such as session cookies, and can be activated when ever they visit the site.



Welcome ohara2099!
Successfully added balance
Your account balance: $400,356.00

Add Balance:

Type: -- SELECT

Number: ••••••••••••••••

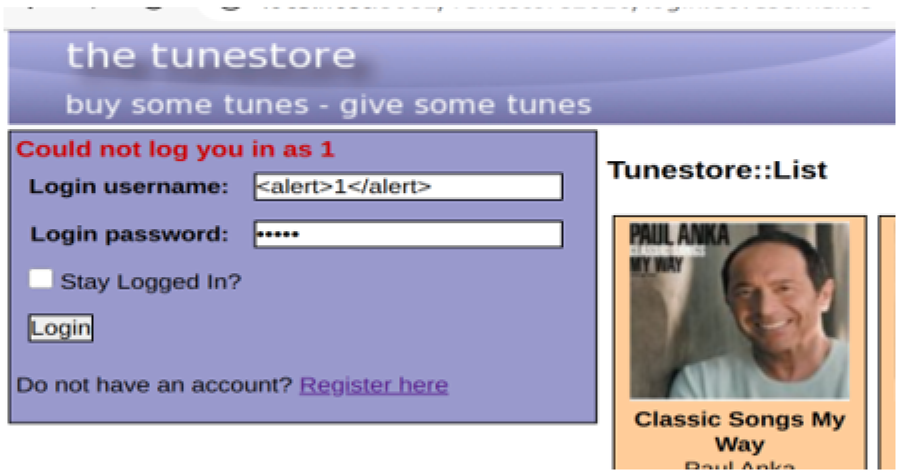
Amount: <alert>1</alert>

Tunestore::List

Classic Songs My Way
Paul Anka

Buy/Gift ($9.99)
Comments

3.2 Reflective XSS Vulnerability
Reflected XSS attacks, also known as non-persistent attacks, occur when a malicious script is reflected off of a web application to the victim's browser. As you can see, when executed we can see the parameter javascript which can be seen as a vulnerability in the application.



Could not log you in as 1

Login username: <alert>1</alert>

Login password: •••••

Stay Logged In?

Login

Do not have an account? Register here

When hackers type the malicious code in the user name field, you can see that sample javascript field has been executed, when it shouldn't be executed, this can be seen as a weakness to hack their way through.