# SecureED.

*User manual*

# Abstract

SecureED. is a webapp that simulates a university registration system. It allows students to try to break the app, using various vulnerabilities and exploits. The software also resets itself to its initial settings following a session, allowing users to try various things without worrying about destroying their installation.

This document contains the manual of how to use SecureED. The rest of the document is ordered as follows: Chapter 1 describes the system requirements. Chapter 2 contains an explanation of how to start the program. Chapter 3 includes instructions for Professors to enter grades. Chapter 4 includes instructions for Students to search for and register for courses. Chapter 5 includes instructions for admins to add, search and edit non-admin user accounts. Chapter 6 describes vulnerabilities and provides explicit examples of their implementations.

This document describes functionality of SecureED. v. 1.0, dated 31 August 2022.

# Table of Contents

# 1 REQUIREMENTS

## 1.1 SYSTEM REQUIREMENTS

SecureED. is written in HTML, CSS, Javascript, and PHP, and was built and tested on Google Chrome v89 – thus, Chrome is the recommended browser. Otherwise, all libraries are included in the installation folder. Screen resolutions above 1920x1080, and aspect ratios greater than 16:9, are untested, and thus unsupported.

**Works best with Mozilla Firefox web browser (not run in private mode)**
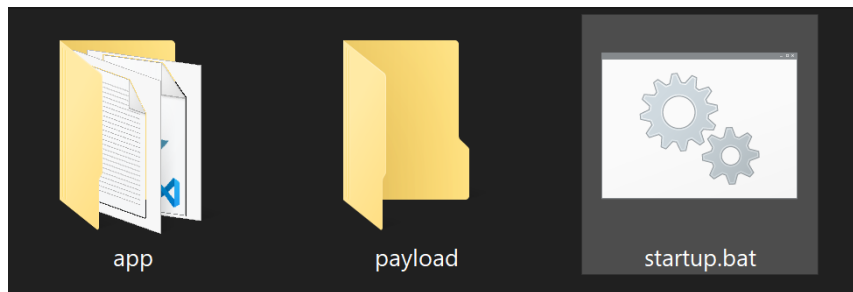
## 2   GETTING STARTED

### 2.1  INSTALLATION

Unzip the file to the folder of your choice. All of the necessary libraries are included – no setup is required.

### 2.2  START-UP

SecureED. is portable, thus self-contained. To start the webserver, please do the following:

1.   Navigate to the root folder of your installation.
2.   Execute startup.bat

Your default browser will be opened, and you will be presented with a login screen. Note the command prompt opened in the background – **DO NOT CLOSE THIS** until you want to end your session! Upon closing the command prompt in the background, your session will be terminated, and all data will be reset to their default values.

## 2.3 LOG IN

Enter a username and password into the spaces provided, and then press "Submit".



*Note: If you have forgotten your credentials, please select "Forgot password?". See the next section for information regarding how to recover your password.*

## 2.4 FORGOT PASSWORD

In order to recover a lost password:
1. Click "Forgot Password?" on the Log In page.

2. On the next page, please enter your email in the provided field, and select "Submit".

**Secure ED.**

**Forgot Password**

Please enter your email:

Email: [ student@email.com ]

[ Submit ]

3. You will then be asked the security question associated with that account.

**Secure ED.**

**Forgot Password**

Where were you born?

Answer: [ Los Angeles, CA ]

[ Submit ]

4. Enter the correct answer, and you will be taken to a page that will allow you to change your password.

**Secure ED.**

**Forgot Password**

Please enter your new password below.

New Password: [ •••••••••• ]

Confirm password: [ •••••••••• ]

[ Submit ]

*Note: Be sure that the contents of the New Password and the Confirm Password fields match!*

## 2.5 LOG OUT

After you have successfully logged in, you may log out at any time. To do so, click on the "Log Out" button on the top pane of any page.



## 2.6 CLOSING THE PROGRAM

To close the program, close your browser and exit the command prompt window. The program will reset itself to its default values the next time you run startup.

# 3 FACULTY

## 3.1 DASHBOARD

| 🔒 | **Secure ED.** |
|---|---|

**Faculty Dashboard**
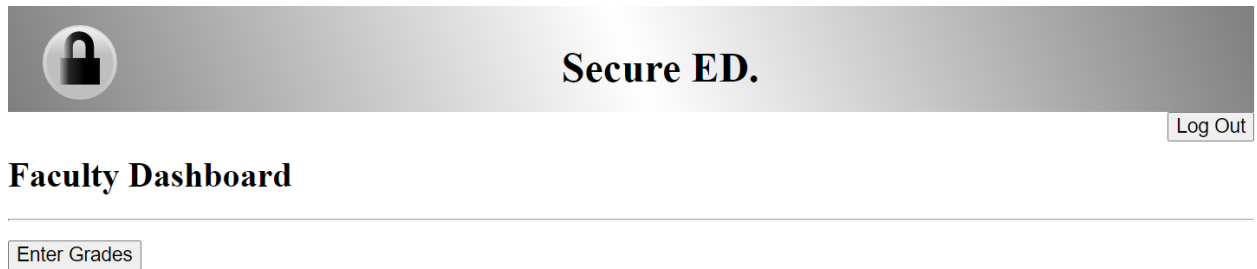
Enter Grades

## 3.2 ENTER GRADES

### 3.2.1 DESCRIPTION

The Enter Grades functionality allows a faculty member to add final grades to a course, using a .csv (comma-separated values) file.

### 3.2.2 CSV FORMAT

Uploaded files must be comma separated text values, in the following format:
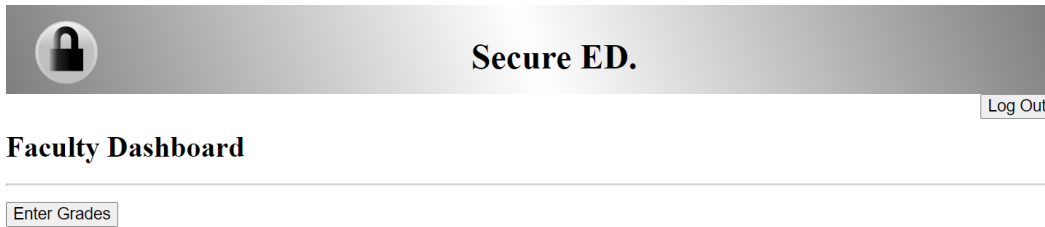
USER_ID,GRADE
USER_ID,GRADE
USER_ID,GRADE
…

e.g.:

927066652,A
927190481,F
927201552,C
927237023,B
927369978,D
927410321,F
927512079,A
927535222,A
927565074,B
927600214,C
927620146,A
927671464,F
927694539,C
927790514,B

### 3.2.3 INSTRUCTIONS
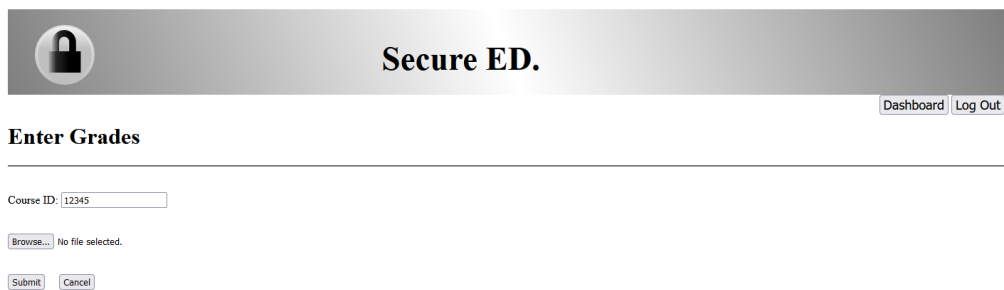
1. From the Dashboard, select "Enter Grades".



2. On the following page, enter the Course ID (CRN).



3. Select "Browse".

4. Select a .csv file from your computer (see 3.2.2 for correct formatting).

5. Select submit to upload the file.

# 4 STUDENTS

## 4.1 DASHBOARD



## 4.2 COURSE SEARCH

### 4.2.1 DESCRIPTION

Course search allows students to find current, past and future courses.

### 4.2.2 INSTRUCTIONS

1. From the Dashboard, select "Course Search".
2. On the following page, enter information relevant to the course you may be interested in.



3. Select search.

**Course Search**

---

**Search Filters:**

Semester: [Fall]      Department: [        ]
Course Name: [        ]      Course ID: [        ]

[Search]

→

**Results:**

| Course Name | Course ID | Professor | Semester | Location | |
| --- | --- | --- | --- | --- | --- |
| Intro to CyberSecurity | 123 | gsinclair@email.com | Fall 2030 | Building A | [Enroll] |
| Intro to CyberSecurity | 1343 | gsinclair@email.com | Fall 2030 | Building X | [Enroll] |
| Intermediate CyberSecurity II | 5000 | scienceguy@email.com | Fall 2030 | Building X | [Enroll] |

4. For a list of sections, select enroll.

| **ıester** | **Location** | |
| --- | --- | --- |
| 2030 | Building A | [Enroll] |
| 2030 | Building X → | [Enroll] |
| 2030 | Building X | [Enroll] |

5. You will be presented with a list of sections to browse.

| Course Code | Section | Professor | Time | Location | |
|---|---|---|---|---|---|
| CYBR 2200 | A | gsinclair@email.com | 8:30:AM - 9:45:AM | Building A | Enroll |
| CYBR 2200 | A | gsinclair@email.com | 8:31:AM - 9:46:AM | Building X | Enroll |

*Note: If you don't wish to enroll at this time, you can always select "Dashboard" to return to your landing page without enrolling in a class!*

## 4.3  COURSE ENROLL

### 4.3.1 DESCRIPTION

After finding a course using Course Search, a student can enroll in a section of the course.

### 4.3.2 INSTRUCTIONS

1. Find a course per the instructions in 4.2.2.

2. Select "Enroll" to bring up a list of sections.

3. Select "Enroll" next to the section you wish to enroll in.



| Course Code | Section | Professor | Time | Location | |
|---|---|---|---|---|---|
| CYBR 2200 | A | gsinclair@email.com | 8:30:AM - 9:45:AM | Building A | Enroll |
| CYBR 2200 | A | gsinclair@email.com | 8:31:AM - 9:46:AM | Building X | Enroll |

You will automatically be returned to your dashboard. ⟶

# 5 ADMINISTRATORS

## 5.1 DASHBOARD
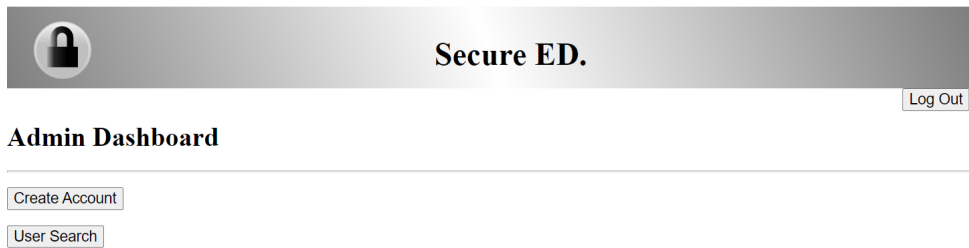


## 5.2 CREATE ACCOUNT

### 5.2.1    DESCRIPTION

Allows the administrator account to create new accounts for new users. This is the only way to create a new account!

### 5.2.2    INSTRUCTIONS

1. From the Dashboard, select "Create Account".
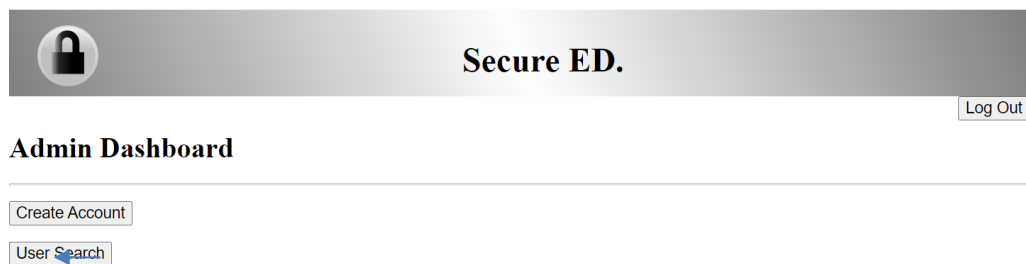
2. On the following page, select the account type.
*Note: Admin accounts cannot be created!*

---



🔒 **Secure ED.**

**Create Account**

---

Account type: [Faculty ⌄]

| | | |
|---|---|---|
| First Name: | [　　　　　⊞] | Last Name: [　　　　　] |
| Date of Birth: | mm/dd/yyyy 📅 | |
| Rank: | [Instructor ⌄] | |
| Email: | [　　　　　] | |
| Confirm Email: | [　　　　　] | |
| Password: | [　　　　　👁] | |
| Confirm Password: | [　　　　　👁] | |
| Security Question: | [　　　　　] | |
| Answer: | [　　　　　] | |

[Submit] [Cancel]

---

3. Enter the user's:
   - First and Last Name
   - Date of Birth
   - Rank (faculty) or Year (students)
   - Email
   - Password
   - Security Question
   - Answer (to Security Question)

4. Confirm the user's e-mail and password, and re-enter them in their respective confirmation fields.

5. Select "Submit".

# 5.3 USER SEARCH

## 5.3.1  DESCRIPTION

Allows administrators to search for users using filters.

### 5.3.2  INSTRUCTIONS

1. From the Dashboard, select "User Search".

2. Select an account type, and a Rank (faculty) or Year (students).
   *Note: These two fields are mandatory!*



3. Enter any more relevant information into the fields provided.

4. Select "Search".

A list of matching accounts will appear below.

## 5.4 EDIT ACCOUNT

### 5.4.1 DESCRIPTION

Allows an administrator to edit a non-administrator account.

### 5.4.2 INSTRUCTIONS

1. Find a user per the instructions in 5.3.2.

2. Select "Edit" to the right of the user's information.

3. Make any desired changes to the prepopulated data in the fields provided.



*Note: Don't forget to confirm any changes to the e-mail or password!*

4. Select "Submit".

# 6 VULNERABILITIES

This section contains the vulnerabilities in the application. Simply follow the steps to see examples of each vulnerability. Some of these vulnerabilities will alter the behavior of the application. Restarting the application (running startup.bat) will reset the application to its original configuration. Use Mozilla Firefox browser (not in private mode) for best results.

## 6.1 IMPROPER INPUT VALIDATION

| | | |
|---|---|---|
| *Improper Input Validation* occurs when input or data is not properly checked to ensure that it can be processed correctly. | | |
| ## | Instructions | Result |
| Ex.1 | 1. Go to the Login page: http://localhost:8000/public/index.php 2. Enter the following for username: *' OR 1=1;* 3. Leave the *Password* field blank. 4. Click "Submit" 5. When done, click "Log Out" | Both the username and password fields are not validated<br><br>This bypasses the login using a Boolean attack, allowing you to access the admin account.<br><br>*Also SQL Injection* *Also Improper Access Control* |

## 6.2 PATH TRAVERSAL

| | | |
|---|---|---|
| *Path Traversal* occurs when users are able to access parts of the web app that they are not supposed to be able to access, due to a lack of proper pathname restrictions. Potentially, methods can be employed which will allow an attacker to access directories that are not normally public, such as the root folder. | | |
| ## | Instructions | Result |
| Ex.1 | 1. Navigate to<br><br>http://localhost:8000/public/../<br><br>2. Click on the folder db 3. Click to download the database<br><br>persistentconndb.sqlite<br><br>4. Go to the login page http://localhost:8000/public/index.php | This will download the database, which should not be directly accessible.<br><br>*Also Improper Access Control* *Also Exposure of Sensitive Information* |

## 6.3   CROSS-SITE SCRIPTING

| | | |
|---|---|---|
| *Cross-Site Scripting* is a type of attack whereby malicious script is injected into a site to be run on the client. | | |
| ## | Instructions | Result |
| Ex.1 | 1. Go to the Login page:<br><br>    *http://localhost:8000/public/index.php*<br><br>2. Enter the following for username:<br><br>    *admin@email.com*<br><br>3. Enter the following for password:<br><br>    *Password1*<br><br>4. Click "Submit"<br>5. Click "User Search"<br>6. For "Account Type" select "Student"<br>7. For "Year" select "Junior"<br>8. Click "Search"<br>9. Click "Edit" to change a student's information.<br>10.     For "First Name" enter the following:<br><br>*<iframe height="166" src="http://www.youtube.com/embed/oHg5SJYRHA0?autoplay=1" frameborder="0"></iframe>*<br><br>11.     Click "Submit"<br>12.     Repeat steps 6 to 8<br>13.     When done, click "Log Out" | This puts a video on the search results.<br><br>*Also Improper Input Validation* |

## 6.4   SQL INJECTION

| | | |
|---|---|---|
| *SQL Injection* is an insertion of an SQL query in a field or file where other inputs are expected, potentially revealing sensitive data or modifying the database. | | |
| ## | Instructions | Result |
| Ex.1 | Note: This will delete portions of the database and the application will need to be restarted.<br><br>1.  Navigate to Forgot Password page:<br><br>    *http://localhost:8000/public/ForgotPassword.php*<br><br>2.  Enter the following for "Email":<br><br>    *scienceguy@email.com*<br><br>3.  Click "Submit"<br>4.  Enter the following for "Answer":<br><br>    *Charity Nye*<br><br>5.  Click "Submit"<br>6.  Enter the following for "New Password" and "Confirm Password":<br><br>    *'; DROP TABLE User;*<br><br>7.  Click "Submit"<br>8.  Enter the following for username:<br><br>    *admin@email.com*<br><br>9.  Enter the following for password:<br><br>    *Password1*<br><br>10.    Click "Submit" (Note: you won't be able to login)<br><br>11.    Since the User Table has been deleted, restart the application (run the batch file startup) | This will drop the User table. The application will no longer recognize any users.<br><br>*Also Improper Input Validation* |
| Ex.2 | 1.  Navigate to Forgot Password page:<br><br>    *http://localhost:8000/public/ForgotPassword.php* | This will add a new admin account for Hacker Man. |

| | | |
|---|---|---|
| | 2. Enter the following for "Email": | *Also Improper Input Validation* |
| | *scienceguy@email.com* | |
| | 3. Click "Submit" | |
| | 4. Enter the following for "Answer": | |
| | *Charity Nye* | |
| | 5. Click "Submit" | |
| | 6. Enter the following for "New Password" and "Confirm Password": | |
| | *'='a'; INSERT INTO User (UserID, Email, AccType, Password, FName, LName, DOB, Year, Rank, SQuestion, SAnswer) VALUES ('10000000', 'hackerman@getrekt.com','1', '111', 'Hacker', 'Man', '1111-11-11', NULL, NULL, 'get', 'rekt');'-- = ' '='a'; INSERT INTO User (UserID, Email, AccType, Password, FName, LName, DOB, Year, Rank, SQuestion, SAnswer) VALUES ('10000000', 'hackerman@getrekt.com','1', '111', 'Hacker', 'Man', '1111-11-11', NULL, NULL, 'get', 'rekt');'--';* | |
| | 7. Login as hacker by entering the following for username: | |
| | *hackerman@getrekt.com* | |
| | 8. Enter the following for password: | |
| | *111* | |
| | 9. Click "Submit" | |
| | 10.    When done, click "Log Out" | |

## 6.5  EXPOSURE OF SENSITIVE INFORMATION

| |
|---|
| *Exposure of Sensitive Information*, as the name suggests, is any information in a web app's code, pages, or file structure, which is sensitive in nature. This can include things like private user information, business information, or any other data that should otherwise be considered confidential. |

| ## | Instructions | Result |
|---|---|---|

| Ex.1 | 1.  Try to login without entering a username or password | This will dump debugging info to the screen. |
|---|---|---|
| Ex.2 | 1. Go to the Login page:<br><br>    *http://localhost:8000/public/index.php*<br><br>2. Enter the following for username:<br><br>    *admin@email.com*<br><br>3. Enter the following for password:<br><br>    *Password1*<br><br>4. Click "Submit"<br>5. Click "User Search"<br>6. Right-click on the page and select "view source"<br>7. Scroll to the bottom of the page<br>8. Right-click to copy the link at:<br><br>    *<script async*<br>    *src="../resources/usersearchdisplay.js"></script>*<br><br>9. Or copy the link below:<br><br>    *http://localhost:8000/resources/usersearchdisplay.js*<br><br>10.    Paste the link into browser (navigate to page) | View detailed implementation information with developer's comments. |

## 6.6 UNRESTRICTED FILE UPLOAD

| | | |
|---|---|---|
| *Unrestricted File Upload* is an exploit whereby files of an unrestricted type can be uploaded, especially types which were not intended to be uploaded by the web designers. | | |
| ## | Instructions | Result |
| Ex.1 | 1. Go to the Login page:<br><br>    *http://localhost:8000/public/index.php*<br><br>2. Enter the following for username:<br><br>    *scienceguy@email.com*<br><br>3. Enter the following for password: | The system uploads an incorrect (and potentially malicious) file that is accessible.<br><br>*Also Improper Input Validation*<br>*Also Improper Access Control* |

| | |
|---|---|
| *Password2* <br><br> 4. Click "Submit" <br> 5. Click "Enter Grades" <br> 6. Enter anything for the Course ID <br> 7. Click "Browse" and select the file named "HelloWorld.php". It should be in the folder named "payloads" in the application folder. <br> 8. Click "Submit" <br> 9. Navigate to the following link: <br><br>     *http://localhost:8000/uploads/HelloWorld.php* <br><br> 10.    When done, go back to the previous page and click "Log Out" | |

## 6.7 IMPROPER ACCESS CONTROL

| *Improper Access Control* is when the application does not provide access to a resource appropriate to the access level of the user. | |
|---|---|
| ## | Instructions | Result |

| ## | Instructions | Result |
|---|---|---|
| Ex.1 | 1. Go to the Login page: <br><br>     *http://localhost:8000/public/index.php* <br><br> 2. Enter the following for username: <br><br>     *student@email.com* <br><br> 3. Enter the following for password: <br><br>     *Password5* <br><br> 4. Click "Submit" <br> 5. Navigate to the following link: <br><br>     *http://localhost:8000/public/enter_grades.php* <br><br> 6. When done, go back to the previous page and click "Log Out" | A functionality that is intended for faculty user is accessible to student user |

## 6.8  IMPROPER AUTHENTICATION

| | | |
|---|---|---|
| *Improper Authentication* is when an app does not properly check the claimed identity of an actor. | | |
| ## | Instructions | Result |
| Ex.1 | 1. Go to the Login page:<br><br>     *http://localhost:8000/public/index.php*<br><br>2. Enter the following for username:<br><br>     *scienceguy@email.com*<br><br>3. Enter the following for password:<br><br>     *1e031774109ee2e6ac244e778ca579d5199e9 4fa3753848a3180e9d2e27e8ff7*<br><br>4. Click "Submit"<br>5. When done, go back to the previous page and click "Log Out" | This logs in with a hashed password, instead of requiring the actual password itself. |

## 6.9  CROSS-SITE REQUEST FORGERY

| | | |
|---|---|---|
| *Cross-Site Request Forgery* is an attack whereby a user can be made to do things on their account that they don't want to do, often via phishing or social engineering. | | |
| ## | Instructions | Result |
| Ex.1 | Note: Requires Mozilla Firefox (Do not run private mode)<br><br>1. Go to the Login page:<br><br>     *http://localhost:8000/public/index.php*<br><br>2. Enter the following for username:<br><br>     *admin@email.com*<br><br>3. Enter the following for password:<br><br>     *Password1*<br><br>4. Click "Submit"<br>5. Find the file named "cutecat.html". It should be in the folder named "payloads" in the application folder.<br>6. Double-click or open the file using a web browser.<br>7. Go back to the SecureED application and click "Log Out" | This will create a new admin account for Hacker Woman. |

| 8. Login as hacker by entering the following for username: | |
|---|---|
| *fake@email.com* | |
| 9. Enter the following for password: | |
| *csrf* | |
| 10.     Click "Submit" | |

## 6.10  CODE INJECTION

| | |
|---|---|
| *Code Injection* is any injection of malicious code, whether by Unrestricted File Upload, Cross-Site Scripting, SQL Injection, or any other method. This example demonstrates PHP code injection via unrestricted file upload. | |

| ## | Instructions | Result |
|---|---|---|
| Ex.1 | Note: You may need to turn off your antivirus software for this example.<br><br>1. Using a text editor, create a file with the following content:<br><br>    *<?php eval ("echo ".$_REQUEST["parameter"].";"); ?>*<br><br>2. Save the file as "Code.php"<br>3. Go to the Login page:<br><br>    *http://localhost:8000/public/index.php*<br><br>4. Enter the following for username:<br><br>    *scienceguy@email.com*<br><br>5. Enter the following for password:<br><br>    *Password2*<br><br>6. Click "Submit"<br>7. Click "Enter Grades"<br>8. Enter anything for the Course ID<br>9. Click "Browse" and select the file named "Code.php" (from step 2)<br>10.     Click "Submit"<br>11.     Navigate to the following link:<br><br>    *http://localhost:8000/uploads/Code.php?parameter=value* | Code is injected that allows the attacker to run php commands |

| | 12. Navigate to the following link: *http://localhost:8000/uploads/Code.php?parameter=value; phpinfo();* | |
|---|---|---|

## 7 USER CREDENTIALS

| Account Type | Username | Password | Security Question | Security Answer |
|---|---|---|---|---|
| Faculty | scienceguy@email.com | Password2 | Favorite Relative? | Charity Nye |
| Student | student@email.com | Password5 | Where were you born? | Los Angeles, CA |
| Admin | admin@email.com | Password1 | How many siblings do you have? | 0 |