Web Security

EXAM PROJECT REPORT

RentOutThings

Group 5
Mads Shawn Hansen
Thao Phuong Vu Do
Petyo Petrov Zhechev

21.05.17

Table of Content

Analysis

Project Description

We are creating a renting app under the name of ROT[1]. Visitors would be able to explore rental ads while users will be also able to create and reply to ads. The posted rental ads can have tags which makes them easier to find. After booking a product, visitors may rate their renting experience. ROT will offer also an informative page with Frequently asked questions.

Problem Statement

Cyber attacks are a serious problem facing web applications for the last decade. A research from Symantec reveals over 70% of websites are vulnerable to attacks[2]. Breaches in applications may cause severe damage not only to the brand/company but also to the users of the product.

The proper reaction to the threat is creating secure code for the application and add additional defense mechanisms in order to protect itself and its users. Keeping the software up to date is a big part of ensuring the security of our site.

Applications are constantly exposed to attacks with malicious intent and action is needed if an application is to remain a safe place for its users.

Requirements

Requirements will be split in 3 parts. The 3 major user groups of the product. Each subsequent group will be able to do/have access to what the previous one is able to.

Note: any mentioned actions that a group can take will mean the previous group is unable to.

[1] ROT - Rent Out Things (The name of the application)

[2] (Symantec) Internet Security Threat Report (ISTR) Vol. 22 p.33 [goo.gl/q7Rukc]

Visitor requirements:

Can see the helpful FAQ page. Can browse through rental ads. Can create a user account.

User requirements:

Can message other users. Can rent or rent out items. Can manage own account information. Can leave reviews. Can report inappropriate content. Can deposit money in their account to use in transactions or withdraw any money on the account.

Administrator requirements:

Can take down reported content.

Risk analysis

The risk analysis sheet will be initially filled with known risks if we were to build the application without securing any of the layers. The sheet will be updated after a security layer has been placed/updated. Once we stop with the conscious improvement of the app ghost chasing would be the next step that will help.

Design principles

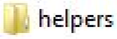
Two complementing principles were chosen in order to have a low success chance of attacks against our product. Defense in depth would be used to minimise damage done (if any) when there is a breach. With multiple layers of security a full breach would be hard to achieve. While the principle of failing securely would ensure that no information on how things are done behind the scenes would be revealed in any scenario nor would unauthorised access to resources can be obtained.

Design

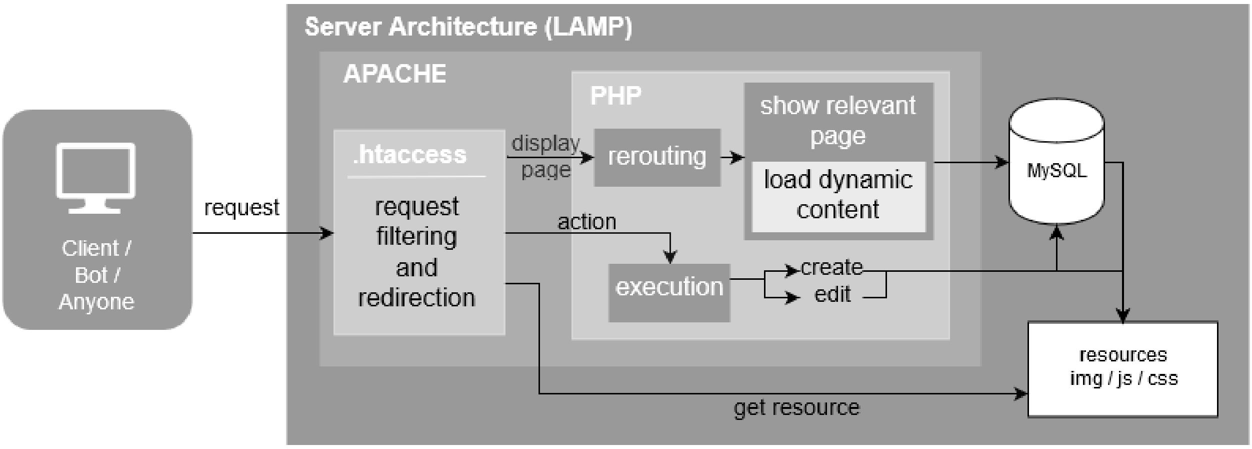
Architectural considerations

Considering the predefined rule of not using frameworks we will be creating the communication structure of our application from scratch. And considering the time span available for development and our application needs, a custom architectural pattern (with significant parts of spaghetti code) would be created.

Folder structure

.htaccess urlRoute.php	- - - - - - - - - - - - - - -
uploads	Contains files uploaded by users
css images js	Web app scripts and imagery
 helpers	PHP scripts that aim simplifying the page development. Will have sanitisation, validation and policy helpers
actions	PHP scripts meant for users to interact with the system/other users/data. (uploading files, creating and modifying accounts, rental ads etc.)
pages	Contains header, footer and different display pages. Mix of html and php to connect with the database and display dynamic data

Request/data flow



Security requirements

All requests would be checked through the .htaccess in our root site directory.

If a resource is required, it will be given.

If an action is to be executed (f.ex. “Create-account” or “add-product”), before its execution, an authorisation and identity check will be made.

If a page is requested, the user would be rerouted and an authorisation and identity check will be made if she/he is allowed to see it. Dynamic content will then be loaded from our database based on the user’s identity. A page will be picked based on the request data sent and comparing it with a whitelist of values. If no match is found, a default page would be showed. All pages will be secured with https ensuring that any data being passed between the user and the application will be safe from eavesdropping.

Any user input that will be displayed to other users will be output using the htmlentities php function in order to avoid XSS attacks. All data being edited or created on the database would go through php and every sql call will be with bound parameters to ensure we are protected against SQL injections. All of the forms and buttons related with an action execution will be having a unique token in order to prevent CSRF attacks. All data will be sanitised and validated between the different layers of our application. Doing so we expect to prevent cross site scripting and maybe PHP injection. To prevent bots from using the register form, reCAPTCHA from Google will be implemented. If the service would be to expand and have a high user base, additional defense mechanisms would have to be placed in order to prevent the more sophisticated robot attacks[3]. Bruteforcing a user's account should be made impossible. An user account will be banned from that IP address for 10 minutes after 5 wrong attempts in a 5 minute period. Also trying to log in with 10 different usernames within 20 minutes would result in another 5 minute ban. Whenever wrong/invalid input is entered, the user will be notified with a custom error message. Uploaded images would be limited to known image types so a script can not be uploaded on the server.

As we want to be sure to minimise damage if someone “breaks in” our web app we would have to set up some extra defense mechanisms. Our database will have multiple accounts set up. Each with different privileges. All of them would have absolutely no rights (except the provided stored procedures we have initially set up for them). In case a user finds a way to execute SQL scripts he wants, he would be limited to whatever procedures he was allowed to use. In case someone does manage to get someones account information out of the database, he will find out that the password is hashed and he can not use it in its current state. Since we will be having transactions done on the website and it’s our most sensitive/important part, we would have to use HSTS. HTTPS in our case can easily be bypassed with a man-in-the-middle attack and we would need to think about “http strict transport security”. Implementing it in the header would give another layer of security to users that visit more than once.

All uploaded images would undergo multiple checks to verify that they would not contain malicious code, path traversal or unwanted data formats.

[3] https://goo.gl/M75e4W

“Data types”

When saving a username in the database, a maximum of symbols may be mandated, but there are multiple other rules which a username might have to adhere. For example a username may consist of letters from a-z, big or small, numbers, but can not start with a number, underscores and dashes and can not end with them, nor have multiple one after another.

In the code, checks for the multitudes of requirements would have to be made. Certain “data types” will be grouped together based on the “entity” they belong to. A user can be defined for example by username, password and email so a helper class will be designed for the user creation that checks if all requirements are met and if the data is valid.

Query Monitoring

```php
<?php
	session_start();

	include('../helpers/DBconnect.php');

	//the policy helper also sanitizes and validates data
	include('../helpers/policy_userAccount.php');

	$sUser = $_POST['user'];
	$sEmail = $_POST['email'];
	$sPass = $_POST['pass'];

	//first we save the received data as is (for it may
be malicious and we want to exam in it later)
	$query = $sUser.' '.$sEmail.' '.$sPass;
	$queryEntered($query);

	//error1 - custom policy/validation fail
	//error2 - database write fail

	switch ($policy_userAcount($sUser,$sEmail,$sPass)) {
	case 'passed':
[...]
```

All data sent from users, regarding of its validity will be sent through a query to the databases so it can be stored and examined by developers. Additional information will be collected in order to find out the alarming cases. Did the query data require sanitisation? Did it pass the validation tests? Which user sent the query? User agent and IP? Page sent through?

Implementation

.htaccess

```
1:  Options -Indexes
     Options -MultiViews

2:  RewriteEngine on

3:  RewriteCond %{REQUEST_FILENAME} !-d
     RewriteCond %{REQUEST_FILENAME} !-f
     RewriteRule ^(.)$ urlRoute.php [QSA,P]

4:  <Files *.php>
       Deny from all
       Allow from 127.0.0.1
     </Files>

5:  <FilesMatch
"action_[A-Za-z]+?(-[a-zA-Z]+)?.php$">
       Order Allow,Deny
       Allow from all
      </FilesMatch>

6:  <Files urlRoute.php>
       Allow from all
     </Files>
```

1: Disallows directory access. Shows error page 403

2: The RewriteEngineOn line allows usage of Rewrite rules and conditions

3: The conditions before the rule state that it will not apply if there is an existing file/directory that is being requested. And the rule would redirect ALL requests to urlRoute.php

4: All access to php files will be denied (except from localhost)

5: All access to php files starting with action_[word]-[word].php is allowed

6: All access to urlRoute.php is allowed

Cookie/Identity Protection

The default session cookie from PHP is used to identify users. To protect them from possible XSS attacks, the php.ini file was edited so session cookies can be http only:

```
session.cookie_httponly = True
```

Additionally a check for XST attacks was made by seeing if TRACE method is allowed with the following command our server's terminal:

```
curl -v -X TRACE http://188.226.140.143
```

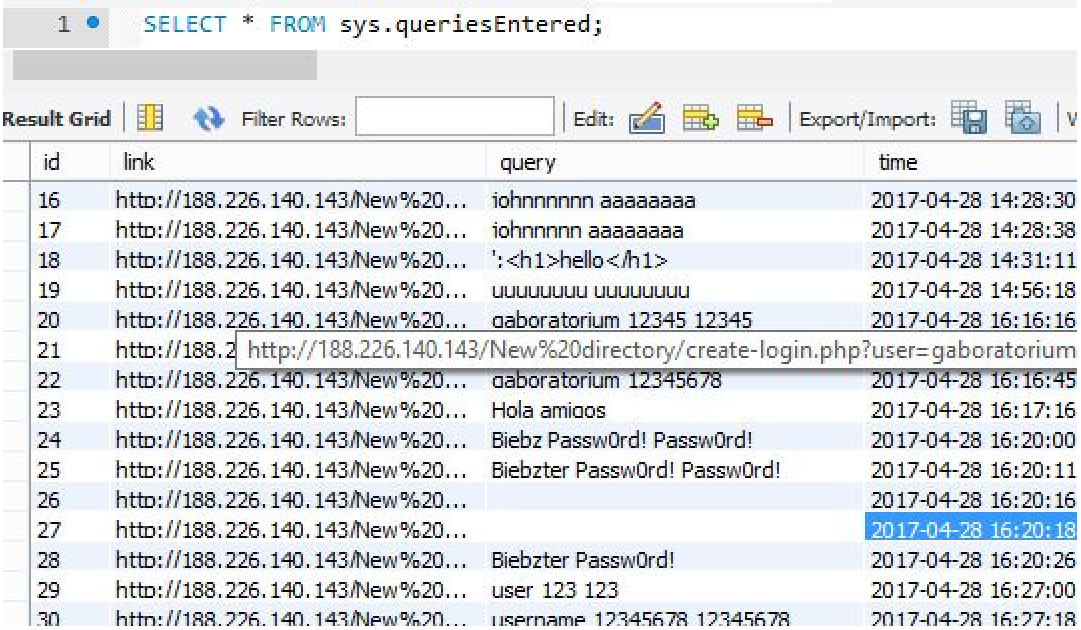
The response was with the error code 405: Method Not Allowed

Actions

```php
<?php
       // error_reporting(E_ALL);
       // ini_set('display_errors', 1);

       //SET A SESSION AND GET THE FORM PARAMETERS
       session_start();
       $sUser = $_GET['email'];
       $sPass = $_GET['pass'];
       $nameSession = $sUser;


       //GET THE DB CONNECTION DETAILS
       require_once '../pages/db_connect.php';

       //ERROR ARRAY AND BAN VARIABLES
       $errors = array();
       $ban = 0;
       $siteBan = 0;

       $queryInserted = $sUser." ".$sPass;

[...]

       if(!empty($lockIp) && $triedUsername == $sUser
&& $siteBan == 0){
       $errors['attempts'] = 'User currently locked
from this ip';
       $ban = 1;
       }

       //SUCCESSFUL LOGIN
       $sql = $con->prepare("SELECT * FROM
websecusers WHERE user=:sUser");
       $sql->bindParam(':sUser', $sUser);
       $sql->execute();
       $result = $sql->fetchAll();
       $resultC = $sql->rowCount();

[...]
```

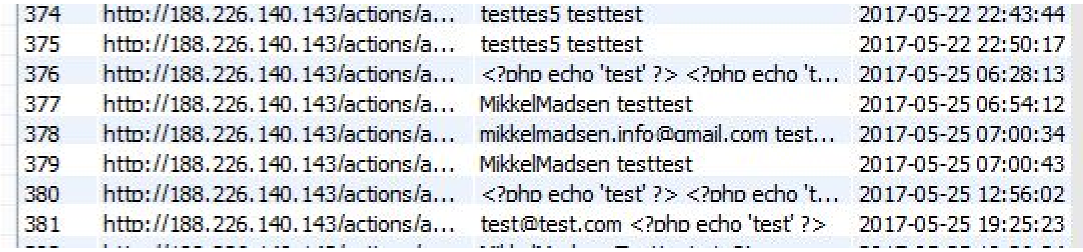
Actions typically operate with user input. They are the entry point of data.
The typical frame of an action is as follows:

1. A session is started
2. User input is saved in variables
3. Database configuration file is called
4. A string containing all entered data is created and sent to the database for future examination
5. Validation and sanitisation checks are made on the input
6. If tests passed, input is being sent to the database through a PDO and prepared statements



```
1 ● SELECT * FROM sys.queriesEntered;
```

id	link	query	time
16	http://188.226.140.143/New%20...	johnnnnnn aaaaaaaa	2017-04-28 14:28:30
17	http://188.226.140.143/New%20...	johnnnnnn aaaaaaaa	2017-04-28 14:28:38
18	http://188.226.140.143/New%20...	';<h1>hello</h1>	2017-04-28 14:31:11
19	http://188.226.140.143/New%20...	uuuuuuuu uuuuuuuu	2017-04-28 14:56:18
20	http://188.226.140.143/New%20...	gaboratorium 12345 12345	2017-04-28 16:16:16
21	http://188.2 http://188.226.140.143/New%20directory/create-login.php?user=gaboratorium		
22	http://188.226.140.143/New%20...	gaboratorium 12345678	2017-04-28 16:16:45
23	http://188.226.140.143/New%20...	Hola amigos	2017-04-28 16:17:16
24	http://188.226.140.143/New%20...	Biebz Passw0rd! Passw0rd!	2017-04-28 16:20:00
25	http://188.226.140.143/New%20...	Biebzter Passw0rd! Passw0rd!	2017-04-28 16:20:11
26	http://188.226.140.143/New%20...		2017-04-28 16:20:16
27	http://188.226.140.143/New%20...		2017-04-28 16:20:18
28	http://188.226.140.143/New%20...	Biebzter Passw0rd!	2017-04-28 16:20:26
29	http://188.226.140.143/New%20...	user 123 123	2017-04-28 16:27:00
30	http://188.226.140.143/New%20...	username 12345678 12345678	2017-04-28 16:27:18



374	http://188.226.140.143/actions/a...	testtes5 testtest	2017-05-22 22:43:44
375	http://188.226.140.143/actions/a...	testtes5 testtest	2017-05-22 22:50:17
376	http://188.226.140.143/actions/a...	<?php echo 'test' ?> <?php echo 't...	2017-05-25 06:28:13
377	http://188.226.140.143/actions/a...	MikkelMadsen testtest	2017-05-25 06:54:12
378	http://188.226.140.143/actions/a...	mikkelmadsen.info@gmail.com test...	2017-05-25 07:00:34
379	http://188.226.140.143/actions/a...	MikkelMadsen testtest	2017-05-25 07:00:43
380	http://188.226.140.143/actions/a...	<?php echo 'test' ?> <?php echo 't...	2017-05-25 12:56:02
381	http://188.226.140.143/actions/a...	test@test.com <?php echo 'test' ?>	2017-05-25 19:25:23

Saving all links and query parameters helped figure out what is being entered in the fields by the users.

Discussion

Since HTTPS in our case can easily be bypassed with a man-in-the-middle attack we would need to think about “http strict transport security”. Implementing it in the header would give another layer of security to users that visit more than once.

Conclusion

We managed to create a fully functioning website which fulfils the features described in the project description. Regarding the level of protection we have achieved throughout the site, we are not satisfied with the results.

We have managed to implement sql protection by using prepared statements, we thought about using stored procedures to further secure ourselves against sql injection but did not have the time. We thought about making a new user account on the servers mysql database with limited privileges in order to minimize the damage that could be done by attacks. We managed to set up the login and register in a way so that the passwords are hashed in our database for safe keeping. We are disabling server errors from being displayed to a potential hacker by setting up rules in the configuration files. We have also set up the default Ubuntu firewall.

Appendix

Group contract

A collaboration agreement is a tool for producing effective group processes. You should all present your expectations to the group work and to each other. That way you can discuss your different wishes and decide how you will work as a group. The group contract should document all your decisions about you group work.

The group contract is relevant because it will help you insure a good group process. A good group process determines a good individual learning process. Some examples of advantages of group work are:

- You grow and learn along with other students
- You learn to co-operate and get insight into your and others' strengths and weaknesses
- You get continuous sparring and discussions with other students
- You can learn from people from other backgrounds than your own
- You will have a small group of people to relate to

When your group is not working some of the disadvantages are:

- Less learning
- Poorer results

- Frustration

On 2nd semester you must use the table below as a foundation for your collaboration agreement:

Group members' full names	E-mail addresses	Phone numbers
1.Thao Phuong Vu Do	thao0056@stud.kea.dk	+45 53832704
2.Mads Shawn Hansen	navn0187@stud.kea.dk	+45 51895059
3.Petyo Zhechev	petyozh@yahoo.com	+45 52222019

Following matrix should be regarded as inspirational. You might find areas of interest and other norms you need to "formalize" within the group. Feel free to add new items. The matrix is your internal "group contract" where you should be able to find answers on to how to cope with all sorts of obstacles you might meet within your group work. You should see it as a work tool…

Topic	Explanation	Group's decision
Group's purpose	What assignment – in brief, is the group going to solve	
Group's goals	What goals – except solving the assignment – is the group going to put focus on: learning goals, process/social goals etc.	We hope to find the time to share our knowledge with each other, as well as holding some social meetings.
Group's ambition	Are you going to be the best in class – or are you going for just reaching the essentials	We aim to make a project that the company would pick as the best solution about security.
Individual group members' goals/ambition	Have you got any individual agendas; ex. part(s) of your individual semester learning goal, that you want to achieve within this group work.	Thao - to get better at coding, web security, and to get better at explaining her programming
		Line - Get better at web security, practice more programming and getting a great new toolbox of programs for web security.
		Petyo – Get to know how some of my classmates work & improve group working skills
		Mads - get in depth knowledge of encryption and user sessions.
Group's strategy	How will the group ensure that each member develops the desired skills; are you going to meet a lot, read a lot, share knowledge a lot, teach each other etc.	We will hold workshops where group members more skilled in certain area can teach others.
How many hours of daily group work? How many hours of daily individual work?		It will vary depending on member's work schedule.
What activities outside the studies will each	Have you got any "outside the school obligations" that you want to	Thao – learning Danish , work Line - Work every Wednesday

member prioritize?	tell your group members about; i.e. children, work, other that is of importance to you and something you want to-/have to prioritize – perhaps even above the group work.	Mads - work Petyo - cleaning the house when it's my turn
When will the group meet for group work?	Make a schedule!	Wednesdays and weekend, depending on the amount of work that needs to be done

How will the group divide the task(s)?	Are you going to do everything sitting together, are you going to work one and one, are you going to solve things in pairs? To give answer to this you should try to breakdown the assignment into different tasks…	We will divide it based on our individual goals and work load. We plan to work individually, in pairs and as a whole group.
How are you going to take decisions in the group?	Are you going to vote and/or use other "democratic tools? Are you aiming for unity and willing discuss each and every issue until everybody agrees or are you going to appoint responsibility and let group members working with i.e. design solely take decisions regarding design…	We will work democratically, talking everything through and eventually voting if we can't reach unity. For each individual matter, the opinion of people with more expertise on the area will hold more value.
What problems are likely to arise in the group process?	Have you any pre knowledge of potential "working together issues". Try to have some explicit considerations regarding who you are as "team player types"	We have never done group work together before, but we are aware of potential problems, and can also deal with them efficiently because we all had experiences with group work before.
How are you going to organize the group?	Are you going to build a group hierarchy with a leader and other appointed "organizational tasks" If so, how are you going to appoint the leader/ decide about the hierarchy? Or are you going to work in a very horizontal and partly "anarchistic" group structure	We have chosen group name "**Fantastic Thrilled Team** of **Programmers (FTTP)**" We will talk with each others and make plans / schedules together.
How will group evaluate the group process the group's ability to reach it's goals?	Will you evaluate every day using ex. SCRUM meetings or will you evaluate otherwise	We will use scrum and burn down charts on Squidhub. We also keep track of our work with Squidhub and Google Drive.
How and when will you punish violation of the group norms?	How often are you going to violate the rules in order to get punished? What types of punishment will you use; baking a cake, split up the group, expelling a team member? If you are going to expel a team member, how will you do it?	The punishment will include bringing "healthy" snacks aso. (Maybe cake.)

RISK ANALYSIS

Risk	Description	Affected System	Probability [1-10]	Probability Description	Severity [1-10]	Severity Description	Mitigation	Control [1-10]	Risk factor
cookies	stealing users cookies and obtaining access to his/her account		5	BEAST attack from 2011 successfully gained full access to cookies from browser	5			3	200
server access	obtaining access to the server using admins credentials		4	multiple conditions must be met to obtain access	10	the intruder would be able to infect the code	separation of privilege	6	200
altering post requests	poking the system with a stick to see what will it do in different cases		9	changing a post request is what most users with malicious intent can do	2	there would be default actions set and failing to use a predefined action would result in nothing.	failing securely	9	36

						If an error is achieved, the returned information would not be sufficient to figure out an exploit or to lead the malicious user into another action			
fake users	creation of a fake user with unreal information		10	the most basic "hack" that could be tried would be registering a fake user and "lending" out to people without actually giving them anything	7	many accounts can be created with ease and filled with fake ads		3	560