# General Questions

* Are this Error pages need to be there or not? Such as: Error 500: Internal Server Error and Stack Trace below it???
* Do I need to use XSS for everything, orthere is some place I can generically refer to them??
* The same with SQL injections, is there any specific place you can generically refer to everything, or you need to refer to them sepsarately ??
* Does the Contact.php work?
* Controller.php-de bir az qalib $f3->get('…. To be CLEANed
* Do you want even outputted ones to not to represent Maruqee shit? In other words, shall I use CLEAN even for Views/Templates on output
* Do you want us to use STRIP\_TAG as well?????/
* ADMIN/COMMENTS SECTION IS is FUCKED UP
* What is the Avatar, on the User/Edit page
* HOMEPAGE IS BLAAAAAAANk nowwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
* Have Fucked Up the Users Page, becomes fucked u, if you play with the Avatar routing
* Can’t log in back to the admin or test  , now I can, chto za random

* Change DEBUG from 3 to 0
* Page/display/somebullshit will work
  + Put a check for it
  + If (empty($page)){
  + Return $f3->error(404);
  + }
* ../../composer.json works, but it shouldn’t
* Htaccess 🡪 protect files
* Sessions are fucked up, that is one of the reasons why you always get an error on first login to the system
* Moderating comment has some problems
* If (POST) check for all forms…. Actually those that make some query to database
* CAPTCHA for email contact form
* Remove the actual Captcha output from all places you have actually outputted it
* Instead of putting captcha straight away count 3 times in 10 minutes and only then allow CAPTCha to come up
* Internal Server Error 505 put if check whther there is usch an entity in database
* Use Function H instead of HTMLSPECIALCHARACTER if necessary
* There is some EVAL being used in BASE.php, doesn’t sound good and lib/db/jig/mapper.php, and view/page/display.htm
* FORM VALIDATION for everything, possible VALIDETTA
* HIDDEN field within contact Us form, transfer it to the function
  + Another one is in the comments section
* PHP PAGE problem still there
* You can comment on others behalf
* <http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/user/view/1>
* DESTROY captscha varaibel ones used it
* Search, still allow for the tags to be used as well… put a hidden field for it

# Target Specific Issues

## SQL Injection

* Database queries, find the queries
* Change your Search and Login to fit Ben’s
  + This->Model->Posts->Find
  + This->controller->Model->Users->fetch
    - Don’t forget to CAST it…
* Encryption done
  + Don’t forget to put SALT entity to the User table
* Still need to do the same with Search box
* Ask more from uncle

## Information Exposure

* Deny for all in Config, View, Helpers and Models folders
* Controller.phps, contact.php.bak, delete them
* Block foo.php, functions.php, install.sql, setup.sh, Config folder
* Contoller and Utility were added htaccess files that do not allow the access to the
* Delete all VIM files, .index.php.swo, .pagesmodel.php.swp
  + Those are temporary files created by the **vim editor**. The .**swp** is created when a file is opened in the editor. The .swo is created if the file is edited and the .**swp** already exists. Edit it a third time and you get a .**swn**, and so on. When vim is closed, those files should go away. If vim crashes or is killed, they may be left behind. They are safe to ignore.
* Img, view and all others are done using Options –Indexes, which doesn’t allow you to view the list, but allows the website to access it for basic work

## XSS

**Just put in the BEFOREROUTE and CLEAN there**

* http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/admin/category
* http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/admin/category/edit/1
* http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/admin/comment/edit/1
* http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/admin/page
* http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/admin/page/edit/rob
* http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/admin/blog/add
* http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/admin/settings MARQUEE just everything
  + But still need to do email validation
  + USE STRIP\_TAG for already inputted html tags
  + MAIL DOESN’T WORK
* http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/admin/user/edit/1 MARQUEE url and form
* http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/blog/view/1234 problem XSS
* http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/blog/index/1234 XSS
* http://linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/user/view/1234 XSS
* BUT still all of them has this problem with ERROR Page
* The title and Comment in COMMENTS section … CLEAN doesn’t work for them, apparently you need to do Strip\_tag as well
* Can’t delete pages with HasTag at the beginning
  + Solved <http://stackoverflow.com/questions/2668854/sanitizing-strings-to-make-them-url-and-filename-safe>
* USE CLEAN FOR THAT - http://fatfreeframework.com/base#clean

USE H FUCNTION instead of CLEAN

## Insecure Upload

* Emily and Aaron has it
* Still not sure

## Cross-Site Request Forgery

* <http://greenash.net.au/thoughts/2010/05/taking-php-fat-free-framework-for-a-test-drive/>
* <https://github.com/bcosca/fatfree/blob/master/lib/changelog.txt> Starting from **version 3.2.0** there is an automatic CSRF protections being implemented
* Done Using Token and Session

## Authorisation Bypass

* From blog/user/view/2 to blog/blog/view/1 and then to blog/admin/comment/edit/1 you can go straight to
* You need to encrypt (MD5) your password, for now they are just VarChar
* Solved … using check the $access

## Internal Information

* Use HTaccess for that
* Speak about foo as well

## Parameter Manipulation

* Have I done it already
* Category/edit/30000

## Application Logic

* You still can access Register page even if you are logged in
* PROMOTE??? Tipo because using grep I’ve found out that it was never used before, I’ve decided to comment it out, it wasn’t deleted for the development purposes so it can be used later on
* Admin.admincontroller.php

## Out of Date Software

* For now we know it is 3.3.0
* Changelog.php contains all the versions
* git command running patch updates for fat free

## File Inclusion

* Apparently it the DEBUG stuff
* Plus the Promote as well

## Insecure Cookies and Sessions

* Randomly generate Key for Users

## Open Redirects

* It might be the PAGE.php, because it has URLdecode, but will see
* linuxproj.ecs.soton.ac.uk/~ea2g12/cyber-security-cw2/blog/user/login?from=%2Fuser
* So you could say that, I still could remove it, but why not to make secure
* <http://fatfreeframework.com/audit> was used

## Brute-force

* <http://fatfreeframework.com/plug-ins>
* <http://www.laurencegellert.com/2014/01/recaptcha-getting-hard-to-read-found-streamlined-substitute-in-fatfree/>

Comment-Aaron

H function, through beforeROute

Brute-force

Open Redirect

* Kind of done, but ask people check for vulnerability

Insecure Cookie and Sessions

Out of Date Software

Upload finish

**Vulnerability number 15**

Those are temporary files created by the **vim editor**. The .**swp** is created when a file is opened in the editor. The .swo is created if the file is edited and the .**swp** already exists. Edit it a third time and you get a .**swn**, and so on. When vim is closed, those files should go away. If vim crashes or is killed, they may be left behind. They are safe to ignore.