Raymond Elward

CSC 438

Prof. Massimo Dipierro

October 30, 2011

Assignment 5

1)

Models are the data representation layer of the MVC pattern. Models typically hold the database connections and computations. It extracts the data from being held by views or controllers so it can be more modular.

Views are the visual representation of information. This is what is represents the model for the user and gives them a representation that they can interact with.

The controllers handle requests made by the users. This deals with changing the model state or pulling from the model to display via the view.

All 3 of these work together to keep the data and the views as abstract as possible from each other so any number of user interface views can be dropped in on top of a data model and still be used.

2)

1)Cross site scripting: This is a attack where the attacker runs scripts through the browser on your server. This can result in hijacked sessions or defaced web sites and various other attacks. This vulnerability occurs when user data is taken and displayed on the browser without being properly escaped.

Example: By default web2py will properly escape things shown in the view. Using helper methods such as DIV('content here') properly escapes everything for the browser so that it won't have the browser running scripts that could be hidden in varibles from the user.

2)SQL injection: This is typically caused when user submissions are directly put into a database. This can change the sql execution or add extra queries, that can result in corrupting your data.

Example: in web2py using the database abstraction layer to do queries will make this impossible. The DAL generates all the SQL so it properly escapes all the data being sent to it. Your code will be vulnerable if you don't use the DAL and you do a statement like "db.executesql('SELECT \* FROM person where id = %s' % userinputvar)"

3)Malicious File Execution: This vulnerability occurs when an attacker is able to get access to the files directly on a users server and then they can execute any file they wish. Using remote file inclusion allows them to include hostile date and code that could be devastating.

Example: Web2py only exposes functions to the users. In the application administration interface a user can see under the controller files exactly what is exposed to the user.

4)Insecure Direct Object Reference: This is when an attacker gets to an object that they are not meant to get to. The developer showing how internal objects are represented and accessed often achieves this.

Example: Generally web2py does not expose any internal objects. web2py also validates all URLs. You can still have the case where a user sees he is directed to:

…./app/default/show/2

To show him the id=2 item in a database. They can manually change the number.

…/app/default/show/3

But if this item is out of scope web2py will complain and say its invalid or even if the programmer doesn’t validate it will issue a ticket error that the object couldn’t be reached.

5) Cross Site Request Forgery: This is where unauthorized commands are transmitted, to a website, from a user that that website trusts. For example attackers hijacking people’s sessions for banks or other sensitive web pages are guilty of this.

Example: As I said before people can steal sessions from scripts in attackers pages, but Web2py specifically fights against this by assigning a one-time random token to each form as well as using a UUID for session cookies.

3)

1) T. T() is used to translate strings to different languages for the users. T() will determine a translation file and use that to display in a different language.

2) URL. This function creates a URL based on the parameters. It is useful in creating a URL and passing variables to it by formatting properly. This will give the correct URL to a controller and function based on what is passed to it.

3) redirect. This function call will redirect the user to a different page. The basic usage of this function is to pass it a URL object and it will take the user to that URL in the application. You can also pass the url of a site outside the application and it will still redirect the user's browser to that site.

4)SQLFORM.factory. This generates a nice looking and CSS friendly form for you. This is best use when you are collecting data from a form but do not have that data represented in a database.

5)IS\_NOT\_EMPTY. This is for use in the DAL. It is an attribute that says this column in a row can not be null. It must be filled or it will be rejected.