UNIVERSITY OF VICTORIA SOFTWARE ENGINEERING

CASE STUDY : APPLICATION SECURITY BIERGARTEN

Braden Simpson
braden@uvic.ca
V00685500

March 10, 2013

1 INTRODUCTION

This is a report regarding the upgrade of the Biergarten online web portal, it will overview the expansion, the new features and requirements, as well as applying the STRIDE[1] threat model, created by Microsoft. It will finally incorporate a DREAD[2] evaluation of an XSS attack on the proposed web application.

2 STRIDE THREAT ASSESSMENT

This section includes a feature listing, including their STRIDE evaluations. The goal of STRIDE is to be intentionally high-level in the thinking, and to simply apply the model to the abstract parts. It is a first step to assessing the threats in a system.

2.1 COMMON SYSTEM COMPONENTS

The common system components between the Tourist Information portal and the Municipal portal are aspects of the system that are most integral to a web application. These include:

- Site Specific Templates
- User Creation and Administration
- Access Rights

- Generic CMS[1] operations

Each of these components will be evaluated with the STRIDE model to assess the threats.

2.1.1 Site Specific Templates

Site specific templates are a good idea for design, since they offer a common look and feel for the user. However they can be a be a potential threat for security. One potential problem with site templates is **spoofing**, by means of phishing users by duplicating a websites' HTML and CSS, on a separate domain and tricking users to enter their credentials.

2.1.2 User Creation and Administration

User creation and administration is obvious to have as a central model. This however, depending on the implementation, could lead to violations of many aspects of the STRIDE model, including: **spoofing**, **elevation**, **tampering**, and **repudiation**. Since the users are very core to the security of the whole system, their security is paramount.

2.1.3 Access Rights

Access rights are another close-knit feature to Section 2.1.2. The method of bootstrapping the users with their automatic permissions for each site, if not developed correctly could have serious threats, most likely in the area of **elevation**, but also in the areas of: **spoofing**, **tampering**, and **repudiation**. This is because if a user is able to elevate their rights, they can manipulate the system.

2.1.4 Generic CMS Operations

This is a problem mainly for the file storage-based threats such as: **tampering**, and **information disclosure**. By storing all the people's data on the same central system, there can potentially be shared data, either by malicious or unintentional means, which can be harmful to the system. The specification states that there will be cookies stored, and kept for ease of login access for users.[3] This is another attack vector, which can be seen as a threat for **spoofing**, as an XSS[2] attack could be used to steal a user's cookie, and login as that user, and perform actions as them, which could also be an **elevation** attack.

2.2 Tourist Information Portal

The Tourist Information portal has some specific features that include :

- Friendly CMS features

[1] A Content Management System is web-based system for uploading, creating, and managing content

[2] Cross site scripting – https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)

- Employees and Business
- Editable profiles
- Access Permissions

2.2.1 Friendly CMS features

This is subject to the same CMS threats as in Section 2.1.4, since it should be using the same backend CMS system. The *friendliness* of the system should be simply a UI change, and should not really affect the threats related to the CMS.

2.2.2 Employees and Business

There are threats here for multiple reasons, one of which is that it is easy to train employees, but it is much harder to train users. For this reason, **spoofing** threats are much more prominent, through attacks on cookies, phishing attacks, and more.

The next reason is that with the business profiles, there is a business entity now setup on the site, so if there were **tampering**, that could reflect very negatively on the business.

2.2.3 Editable Profiles

The users, from the spec "have access to create and edit visible contact such as address, phone number, menus, and pictures".[3] The way this would work is through the central CMS, given in Section 2.1.4 and would suffer the same risks. This method allows users to put up their (potentially sensitive) data on the site, and would suffer threats such as: **spoofing**, **tampering**, and **information disclosure**. When a profile is edited, the content could contains malicious XSS scripts. In an extreme case, false information could be put up to impersonate another person or business, in order to defamate or slander that entity.

2.2.4 Access Permissions

From the specification, "Some administrative details are not to be shown publicly, but are accessible for editing by the business users. "[3] This means that there must be an access model in place, which refers back to the system described in Section 2.1.3. The access system could potentially have the **elevation**, or **information disclosure** threats associated with it.

3 Municipal Service Site

The municipal service site has the following features:

- Governmental Sections
- Department Administrative Users

- Citizen Service Access
- Secure Services

3.0.5 Governmental Sections

The municipal site will be sectioned off into the departments of government, with separate functionality, and separate data. This can pose a threat of **tampering**, **information disclosure**, and **elevation**. The data could potentially be accessed by other departments, and be read by unwanted parties, the separate functionality could potentially elevate the priveledges of users, based on their department. The fact that the people who are not technical could potentially be managing access rights could result in unwanted access rights, or bad access practices.

3.0.6 Department Administrative Users

Having a few users from each department be the *managers* of the content for that department is actually a good security measure. This takes away unneeded priveledges, and allows for most of the users to not worry about securing their logins and reduces the number of possible attack vectors.

3.0.7 Citizen Service Access

This is a feature with a large amount of responsibility. Once users are uploading very sensitive information such as tax forms, application permits, and other official documentation, the stress on data security increases dramatically. The threats here include: **tampering**, and **information disclosure**. The difference is, with this feature, the information being accessed is of much higher value than less sensitive personal information such as name, location, or email.

The service offered seems like it might be unnecessary and unless absolutely necessary and funded properly, I would recommend removing it from the system. There are multiple examples of online systems that store this fundamentally sensitive information being compromised, such as [4].

3.0.8 Secure Services

To secure the services discussed in Section 3.0.7, would require a large amount of good security practices and testing. The users would need to be ensured that their sensitive tax and other data is safe. The threats related to insecure data would be **tampering**, and **information disclosure**.

4 Mitigating Threats

In order to mitigate the threats outlined in this report, the tables in [3] should be followed. They provide methods to dissolve the risk associated with these threats.

For example, to mitigate **spoofing**, the system must provide *authentication*, *secret protection*, or *not store secrets*. The rest of the solutions may be found at [3], P. 8.

5 D.R.E.A.D ASSESSMENT

D.R.E.A.D is a model to prioritize risks [2]. In this report we will be applying D.R.E.A.D to the proposed Biergarten web portal on the XSS attack risk.

Damage Potential The damage potential of an XSS is quite high, especially with a system such as the proposed one. One of the most common XSS attacks to do is to use shared content (ex. uploaded forms or text) to store a hidden script and then once other, legitimate users visit that content, they could have their cookies stolen. A stolen cookie can result in spoofing that victim's user account and performing any action they could, this is one of the downfalls of using session cookies. This could be even more devastating if the victim was an administrator user. The damage depends on the user attacked, but this attack can still be classified as a **high** level risk – 3.

Reproducibility XSS is a very common attack, and it is mitigated via two factors: good programming practices to stop attackers embedding script into sites. As well XSS can be done by baiting users to click on a link which appears to be from a trusted source, but can contain a malicious script. This can be mitigated by having users trained to be careful for such links, and to not trust anything, even if it appears to be from a trusted source. Depending on these two factors, the reproducibility can be from low to high, but would probably stay around **medium**.

Exploitability The exploitability of an XSS has esstentially two parts: writing the malicious code, and delivering that malicious code to an unsuspecting user. The latter is by far more difficult, since the malicious code needed to do an XSS attack can be found by doing a quick google search. From the OWASP top ten vulnerabilities in 2010, XSS was number 2[5], making it very exploitable. This D.R.E.A.D evaluation will rate it at **High**.

Affected Users The affected users of an XSS attack can range. If the attack is done on a shared section of the website, for instance the front page of the website, then all users would be affected. However a targeted email based XSS could be less extreme, and only affect the users that click on the malicious link. Therefore the rating will be determined as **medium**.

Discoverability As stated previously, XSS is among the top two most common vulnerabilities in web applications, this makes the risk very accessible even to attackers that might not be experienced. The theory behind XSS is simple, and the code does not require much knowledge. For example, [6] contains a large list of example XSS vulnerability testing scripts for checking if the server escapes HTML correctly. There are published methods to perform these XSS attacks, and for this reason, discoverability is ranked as **high**.

Performing the final calculation for total normalized risk level based on the D.R.E.A.D framework described in [3], we take the mean of the risk level after applying numerical values to it.

$$\frac{Damage + Reproducibility + Exploitability + Affected + Discoverability}{Total} \tag{5.1}$$

$$\frac{12+8+12+8+12}{5} = 10.4 \tag{5.2}$$

This result indicates that XSS is indeed a **medium-high** risk threat that needs to be addressed in the making of the Biergarten web application.

References

[1] OWasp, "OWasp Threat Modeling Guide," Mar. 2012. [Online]. Available: https://www.owasp.org/index.php/Application_Threat_Modeling#STRIDE

[2] ——, "OWasp DREAD Threat Risk Ranking Model," Mar. 2012. [Online]. Available: https://www.owasp.org/index.php/Application_Threat_Modeling#DREAD

[3] A. Law, "Case Study, Biergarten," Mar. 2012. [Online]. Available: http://www.ece.uvic.ca/ henrylee/2013/08/08AppSecurity-Case.pdf

[4] BankInfoSecurity.com, "Stolen Password Led to South Carolina Tax Breach," Mar. 2012. [Online]. Available: http://www.bankinfosecurity.com/stolen-password-led-to-south-carolina-tax-breach-a-5309/op-1

[5] OWASP, "OWASP Top Ten Vulnerabilities, 2010," Mar. 2012. [Online]. Available: https://www.owasp.org/index.php/Top_10_2010-Main

[6] ——, "OWASP XSS Cheat Sheet," Mar. 2012. [Online]. Available: https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet