SAMS Project

Security Plan

Thomas Travers
Grantham University
CS406

SAMS Project

Thomas Travers

Security Plan

Context/Overview:

Use the security-related activities in the life cycle shown in figure 24.4 to create at security test plan.

Resources to consult:

Chapter 24

Specific questions or items to address:
1. Domain Model from Figure 24.8

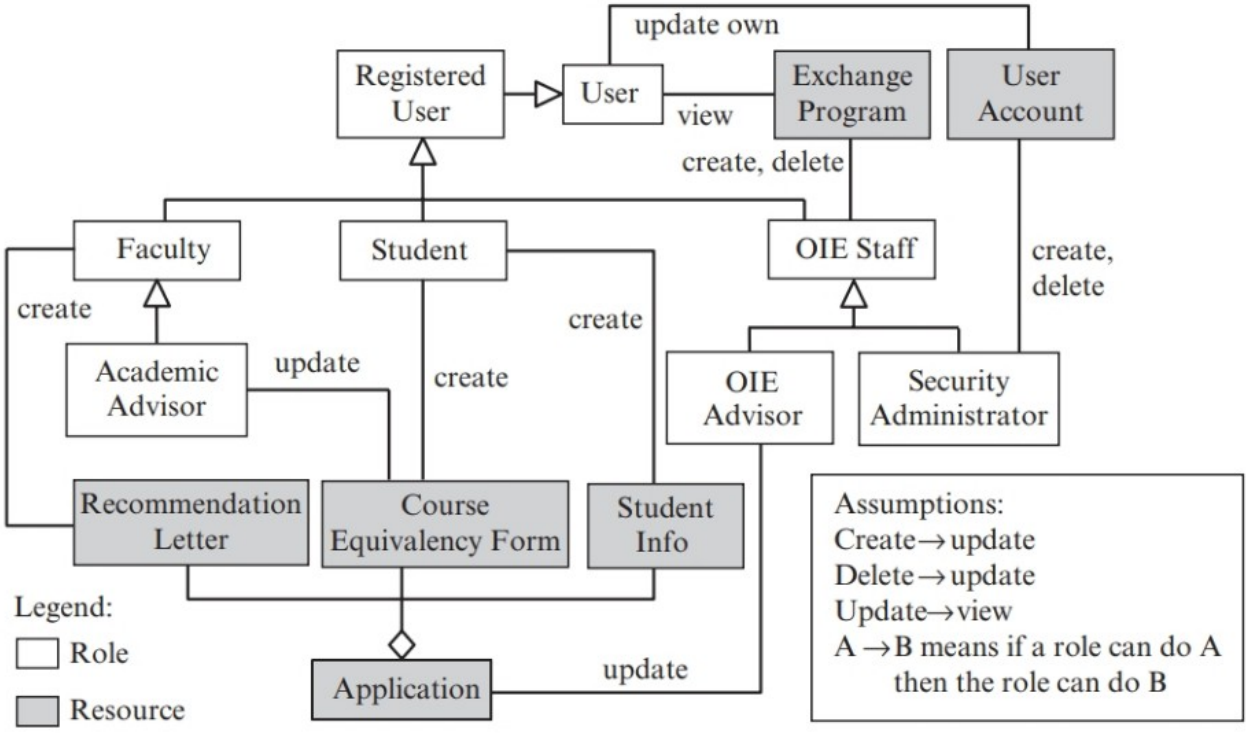

FIGURE 24.8 Part of a domain model showing security information

2. Identify security threats and derive security requirements and misuse cases. See examples 24.2 and 24.3

Security Threats

Every aspect of the system can be miss used if security measures are not properly implemented. Here are the threats that I can identify for this project…

- *Brute force attack on servers (Web Logins, SSH and SQL Services)*

- *Exploit of default user names and passwords*
- *Man in the middle attacks*
- *Social Engineering attacks*
- *Phishing Attacks*
- *Denial of Service Attacks*
- *Domain Name hijacking / ransom holds*
- *Input manipulation*

To combat these misuse cases… the following requirements shall be set

(a) All servers in use shall deploy a properly configured IDS/IPS + firewall to combat brute force attacks, vulnerability scanning and port scanning.

(b) All servers in use shall deploy a properly maintained antivirus and antimalware system to combat rogue payloads that can compromise the network.

(c) The web forms and interfaces shall validate all input types.

(d) No default username's and passwords will be in use. All of these will be disabled immediately after an admin account is created with a unique username and password.

(e) Root login will not be allowed on any SSH server. An admin will use their user account and escalate with su or sudo during remote connections.

(f) All SSH connections will use a generated SSH key. No password authentication will be allowed.

(g) Domain names will be set to auto renewal to prevent laps where the name can be hijacked and held for ransom.

(h) No links will be sent to any messenger or email. Any notice will require the user to manually login from a browser or approved web application. This will help prevent phishing.

(i) No staff will ever ask for personal information or account information via any communication type. The user will be required to contact staff directly in order to prevent social engineering attacks.

(j) All users will be made aware of staff procedures by receiving notices of what the staff will not do in order to maintain privacy. Copies of this will also be a part of the privacy notice.

(k) All users will be required to read and agree to the user agreement, privacy notice, and security policy in order to use their account prior to first login.

(l) All users will be required to read and agree to any privacy policy, security policy, or user agreement updates, changes, and notices upon login before being allowed to proceed using the app.

(m) Changes to user data will require 2-factor validation of the following by sending a code that must be entered into a data field on the web app...

1. *user email*
2. *user phone textbook*

(n) Users will ONLY be assigned access and privileges based on their job duties and roles. They will not receive any higher access to the system than what is needed. Access not needed will be stripped away after (see section 3 for details)

3. Specify role-based access rights

Users will ONLY be assigned access and privileges based on their job duties and roles. They will not receive any higher access to the system than what is needed. Access not needed will be stripped away after.

The main user roles will be…

(1) Administrator – has access to all accounts as well as code, database, logs, and analytic data required to maintain the site.

(2) Student – the student can browse available abroad courses and open seats. They can apply to them, submit documentation in the portal, and check status of pending applications along with view the account actions history including documents submitted and dates submitted.

(3) Advisor / Counselor – A student advisor will be given access rights to log into a student's account to assist the student in setup and search. The student will have the ability to grant and revoke access from the settings. Any changes an advisor makes will appear in the students dashboard on the portal and will require the student to review and submit manually themselves giving them the opportunity to review the changes / additions made to their application.

(4) Staff – Staff will have the ability to upload recommendation letters to the students portal, as well as have access to analytical data and insights as to the status of applications and documents.

(5) Admissions – This type of user will have the ability to register classes with the system as well as approve or deny "their applicants". They will have the ability to review and request pending applications as they come in. They also have the ability to suspend application submissions when their seats are filled. They can even have one click access for direct contact with the students and staff that recommended them and send recommendation letters in order to request more information or address followup questions or concerns.

(6) Outsiders – This type of access can be granted without login in the form of a single web page that gives statistics and analytical data without any personal information attached. This could be useful for board members, staff, and students to view real time analytical data about applications submitted, applications approved, and they can even sort data by location allowing them to see where their specific school or desired location fits into the analytical data. This data along can be used to determine funding, or popularity of a program. For example, if students are not applying and the cost to maintain the program is too high, a board might vote to cut a particular program that isn't receiving well. A school can decide to grant more funding if they find that they are constantly overbooked with foreign students and need to gain more seating in the class rooms in order to accommodate more studnets.

4. Discuss what will need to happen when requirements change. See section 24.7.2 Security in the Iterative Phase.

When requirements change, a change proposal must first be made. Before the change can be implemented, we must first draft a new domain model to show security related relationships and formulate a list of security threats. Me also assess how this could effect the components of the system and the role based access rights.

5. Discuss security in implementation, testing, and deployment

Security will be implemented just as stated. It will be tested and deployed in a physical work environment. If a security related test fails, the system will not deploy until it is fixed. If a security related bug or vulnerability is found after deployment, a new security requirement will be made, and a test will be made from this requirement. A bug fix will be coded and tested against this requirement and once it is met, an update will be pushed to correct the issue. The same goes for the human element as well, anyone who's account has been compromised due to fishing and social engineering will receive an email explaining how these type of exploits work and how to avoid them as well as their account requiring them to review and agree to the privacy policy all over again.