SE4950 – Fall 2020
Project 1 – Threat Modeling

Background

The purpose of this project is to provide you hands on experience in developing a Threat Model for a system.
According to the Microsoft Threat Modeling methodology, threats are categorized into the following categories:

- Spoofing
- Tampering
- Repudiation
- Information Disclosure
- Denial of Service
- Elevation of Privilege

Threats are modeled by first developing a Data Flow Diagram (DFD) for the system. Then, the STRIDE categories are used to look at each of the DFD elements and identify the threats that are possible against that element.

System To Model

For this project, you can choose one of two systems to model. The first choice is the Dropbox cloud-based storage system. The second choice is a Online Reviews Website. Please read the system descriptions below, then choose one to model.

Dropbox

Dropbox is a cloud-based storage system for your desktop and mobile devices. Store a file in a folder on your desktop, and it gets immediately uploaded to the server, and synchronized across all of your registered devices. Users can share folders in their Dropbox, allowing other users or the general public a view of the files. Also, each change to a file is kept in a version history, so that users can revert to old copies of their files. All of this can be managed via a desktop client, a mobile client, and a webapp.

One feature you may not know about Dropbox is their *de-duplication engine*. The clients maintain a local list of hashes of the files in your folder. When a file is changed, Dropbox re-computes the hash, and sends the hash to the server. The server then uses this logic on each hash:

- **New change.** If the server doesn't have the hash, upload the file.
- **No change.** If the server has the hash associated with the user's account already, and no files are uploaded.

- **De-duplicate.** If the server has the hash already, but another user owns it, then make a record that a user is storing a copy of the file associated with that hash (i.e. no files are uploaded, just some bookkeeping)

De-duplication can provide a huge speedup in upload times and a gives big savings in storage for Dropbox. (The actual algorithm that Dropbox uses is even more complicated than stated, but for the purposes of the project let's leave it at this. We can discuss the actual algorithm in class after the project).

Dropbox servers are also spread across multiple physical sites, and data is replicated across those sites for reliability and performance. This replication is handled by a single database management system. Mobile and desktop clients (even the webapp) all interact with a common API.

Online Reviews Website

The basic characteristics of the Online Reviews Website include:

A. A web server for handling users' requests. After a user posts or removes a review, the user will get an e-mail acknowledging the operation performed. Also, if the administrator removes a review, the author of that review will get a notification by e-mail. There's an e-mail server for that.
B. The System handles three types of users: anonymous users, identified users, and an administrator user.
C. Anonymous users can read reviews. Identified users can do what anonymous users do plus write reviews and remove their own existing reviews. Administrators can perform privileged operations (the usual stuff an administrator should be able to do).
D. Reviews stored in a Relational Database Management System.
E. The system is configurable. The administrator has access to a configuration file where the values for different options for the Review Website are specified.
F. Every operation that the web server performs for a user is logged. That means that the web server will generate log requests. There's a log server (logger) that supports system monitoring and debugging. Some logged information is private and can only be accessed by the administrator. The non-private information can be viewed by any authenticated user.

The Activity

For the activity, it is helpful to do it in groups of 2-3 to allow discussion on the diagrams and threats. However, I understand we are in a unique environment this semester, so you may complete the project by yourself as well. I would encourage you to ask me questions, and bounce ideas off of me if you are doing the later.

1. Discuss as a group what the Architecture of the system will look like.
2. Start the Microsoft Threat Modeling tool. Choose the Create a Model option, and save the blank model in a convenient location using the name Dropbox.tm7.

3. From the File menu, select the Threat Model Information… menu item, and enter the names of your group in the Contributors box. Note, this is important to receive credit and your grade for the assignment.
4. Start the model by adding the **Data Stores**, **Processes**, and **External Interfaces** for the system. This is done by dragging the icons from the Stencils window to the Diagram.
5. Next add the **Data Flow** relationships by selecting two of the other elements and selecting Connect or Bi-Directional Connect.
6. Next, add **Trust Boundaries**.
7. Change from the Design View of the Tool, to the Analysis View.
8. Notice the large number of potential threats in the System.
9. Eliminate some of the threats. Discuss as a group which threats you believe are not possible in the system you are modeling. Select those threats and change their state to "Not Applicable", and enter the reason for this in the Justification box.
10. Select threats that you believe may be mitigated, set the status to mitigated, and write the mitigation strategy in the Justification field for the threat.
11. Test your model as well. Try to think of a security concern in the system. This can be a design concern or something specific to the domain. Where does it fit in the model? If it doesn't, then revise the model.
12. When you have mitigated at least 5 threats, then from the Reports menu, select Create Full Report and save the report HTML file.
13. You will turn in both the *.tm7 file and the *.html file for the assignment.
14. Have one group member turn the project in on Blackboard (making sure the other's names are in the model properties in the .tm7 file). As I am letting you select your own groups, I have not made this a group assignment in Blackboard, but will manually override the grade for the other group members.

Note, for thinking of threats and mitigations, the Threat Trees in Chapter 22 of the Microsoft Security Development Lifecycle ebook may be helpful.

Grading

The project will be worth 50 points, and graded as follows:

- 25 points – DFD Diagram completed
- 10 points – At least 5 threats with identified mitigations
- 15 points – Thoughtful analysis of the threats and mitigations.