Steve Barton

M2 Submit

Data Flow Diagrams (DFDs) and risk assessments are essential tools when designing a system and the security of the system. A DFD helps visualize the processes of a system and how data flows within the system. A DFD includes external entities, the main and supporting processes and the databases. DFDs are broken down into levels, with level 0 being the highest form. Visualizing a system and its data flows is important to understand where attacks may originate.

This understanding and visualization will help in the risk assessment. A risk assessment is meant to evaluate potential threats that may exist within a system. When potential threats are evaluated, an organization can analysis, identify and mitigate risks. Protection of the assets must be first and foremost in the mind of the organization, therefore, DFDs and risk assessments must an integral part of system design and development. Therefore, in this report there is one Level 0 DFD, three Level 1 DFDs, and their corresponding risk analyses.

The Level 0 Data Flow Diagram (see Figure 1) details The Recipe Hub in its highest form. This high-level overview of the application illustrates the main process, The Recipe Hub Web Application, as a circle. It is central to the functioning of the application by handling user registration and authentication, profile management and recipe interaction. It communicates with the two databases (illustrated as open-ended rectangles) and the two external entities (illustrated as squares).

The Recipe Database stores all relevant information concerning the recipes while the User Content Database is responsible for user profile data, user generated content, and the management of said data. The two databases communicate with each other to link user recipes and comments to the corresponding user account. The User is any person interacting with the web application. User interactions include searching for recipes, submitting recipes, creating and managing an account and commenting on other user’s recipes. The Third-Party Email Services are external services providing secure communication between the web application and the end user.

One identified threat to The Recipe Hub Web Application is Denial of Service (DoS). A DoS impacts the availability of a system. It is almost always caused with malicious intent to stop a web application’s service by overwhelming the servers with requests to connect. These requests, however, are not acknowledged by the client (a threat actor) after the web application accepts a connection. If there is an overwhelming amount of connection requests without client acknowledgement, the web application is unable to accept anymore connections from users with legitimate purposes. This can hurt the target’s reputation, financial bottom line and daily operations.

A breakdown of the threat analysis for The Recipe Hub Web Application is provided as follows:

**Process:** The Recipe Hub Web Application

* **Threat ID:** DoS-01
* **Threat Description:** A threat actor could launch a Denial of Service or Distributed Denial of Service attack on The Recipe Hub Web Application to effectively shut down its services.
* **Threat Type (STRIDE):** Denial of Service
* **Impact:** Preventing Users from using its services, The Recipe Hub Web Application could lose revenue and reputation.
* **Likelihood:** Medium
* **Mitigation:** Utilizing a DoS or DDoS protection service from third party vendors or service providers.

**A diagram of a company

Description automatically generated**

Figure 1: Level 0 DFD

The first level 1 DFD (see Figure 2) details a major process, Profile Management, illustrated as a dark circle. The lighter colored circles are sub processes (User Profile Information and Saved Recipes) to the Profile Management major process. The Profile Management major process handles the functions between the external entities and the sub processes of the web application.

The first external entity is Third Party Email Services. This external entity sends the User any profile alert emails, in the context of profile management. These profile alert messages are handled by the major process. The second external entity is the User. The User interacts with the major process by requesting to view or edit their profiles and recipe submissions.

Next, the major process processes the User input and sends it to the appropriate sub process. The sub process then handles the request from the Profile Management process and sends it to the appropriate database. The sub process will either request user data or recipe submissions if the User is requesting to view their profile or recipes or send data to be stored if a user updates their profile or recipe.

One identified threat to the Profile Management process is tampering. Tampering impacts the integrity of a system or user’s data. In a nutshell, tampering occurs whenever a threat actor gains unauthorized access to data and modifies or deletes the information. How the threat actor gains unauthorized access is not necessarily the focus of tampering, but rather the focus on what the threat actor can do when altering data. For example, once unauthorized access is gained, a threat actor may compromise the confidentiality of a user’s information as an inadvertent side effect. They then may disclose said information. But more to the point of tampering, a threat actor can alter the access controls of a system or delete company trade secrets.

A breakdown of the threat analysis for Profile Management is provided as follows:

**Process:** Profile Management

* **Threat ID:** Tamp-01
* **Threat Description:** A threat actor could gain access to a user’s profile (via credential stuffing or social engineering) and alter data and preferences leading to privacy breaches.
* **Threat Type (STRIDE):** Tampering
* **Impact:** The Recipe Hub’s reputation would be impacted. This could result in users not trusting The Recipe Hub and instead using a competitor.
* **Likelihood:** Low
* **Mitigation:** Implementing robust password policies, implementing 2FA before any changes in user profile data can be performed.

**A diagram of a diagram

Description automatically generated**

Figure 2: Level 1 DFD; Profile Management

The second Level 1 DFD (see Figure 3) details the Recipe Interaction major process. Shown as a dark colored circle, the major process of this Level 1 DFD acts as the mediary between its light-colored circle subprocesses (Search Recipes, Submit Recipes, Comment on Recipes) and the two square external entities (User and Third Party Email Services).

The major process handles input from a User wanting to search for recipes, add new recipes, or comment on other user’s recipes. These inputs are then processed and validated before being sent to the appropriate sub process. From there the sub processes will send the request to the Recipe Database, which will then sync the data with the User Content Database (both illustrated as open-ended squares). Depending on what was requested, the sub processes will either retrieve recipe search results or comments on the User’s recipes or send the inputs to be stored as either recipe data or comment data.

Once retrieved, the major process will get the request ready to be displayed to the User. Furthermore, if one user makes a comment on another user’s recipe, the major process will process the new comment and make a notification. Then the notification will be sent to the Third-Party Email Service where the User can retrieve the notification.

One identified threat to the Recipe Interaction major process is Information Disclosure. Information Disclosure directly impacts the confidentiality of data. A threat actor may have many avenues of approach to compromise confidentiality, but one possible method is through Cross Site Scripting (XSS). XSS involves the creation of a malicious script that is embedded in a recipe submission. An unsuspecting user could then interact with said recipe and become infected. The infection could happen when the script is executed in the user’s browser, leading to the stealing of session tokens, user credentials, or other sensitive data.

A breakdown of the threat analysis for Recipe Interaction is provided as follows:

**Process:** Recipe Interaction

* **Threat ID:** Info-01
* **Threat Description:** A threat actor could inject malicious code within a recipe submission to perform XSS. The malicious script could be used to infect a user and steal information.
* **Threat Type (STRIDE):** Information Disclosure
* **Impact:** Theft of data and system compromise for a user
* **Likelihood:** Medium
* **Mitigation:** Implement secure coding practices that allow for strict input validation and sanitization of all content submitted by users.

**A diagram of a company

Description automatically generated**

Figure 3: Level 1 DFD; Recipe Interaction

The last Level 1 DFD (see Figure 4) details the User Registration and Authentication major process as a dark colored circle. The major process of this Level 1 DFD acts as the mediary between its light-colored circle subprocesses (Register New User and Account Manager) and the two square external entities (User and Third-Party Email Services).

The major process handles input from a User wanting to make a new account, login to an existing account, or request to reset their password if forgotten. These inputs are then processed and validated before being sent to the user Content Database (open ended square). If a user is making a new account, the major process will process and validate the data and send it to the Register New User subprocess. This subprocess will then send it to the User Content Database. If the User is performing a login request or a password reset request, the major process will validate the User’s credentials or new password and send it to the Account manager subprocess. If the User requested to reset their password, the Account Manager will process a 2FA email to authenticate the User and the new password. After this step, the Account manager will store the new password in the User Content Database. However, if the User is attempting a login, this subprocess will retrieve the appropriate User data from the database and authorize or deny the User based on the stored data and the User’s input. If the User is authorized, a 2FA email will be processed for login.

One identified threat to the User Registration and Authentication major process is Elevation of Privilege. Elevation of Privilege directly impacts the confidentiality and integrity of data and even may include the availability of a system, depending on what the threat actor does. There are different ways a threat actor may accomplish gaining elevated privileges, but the attack vector outlined in the threat analysis (shown below) is that of SQL injection. In this attack vector, malicious code is injected into query fields (e.g. username and password input fields). The purpose is to have said malicious code be sent to the database where data could be disclosed to the attacker, or in this scenario, altered to allow the attacker administrative privileges due to access controls being altered. The attacker could then delete or modify data.

A breakdown of the threat analysis for User Registration and Authentication is provided as follows:

**Process:** User Registration and Authentication

* **Threat ID:** Elev-01
* **Threat Description:** A threat actor uses SQL injection in username and/or password fields, bypassing authentication controls and gains administrative access.
* **Threat Type (STRIDE):** Elevation of Privilege
* **Impact:** Compromise of system integrity and user confidentiality. Personally Identifiable Information could be stolen and leveraged.
* **Likelihood:** High
* **Mitigation:** Good validation and sanitization of entry fields and utilize penetration testing for the registration and authentication processes.

A diagram of a company

Description automatically generated

Figure 4: Level 1 DFD; User Registration and Authentication

In conclusion, the DFDs and risk analyses have shown areas of potential threats. Common to all the processes outlined in this report is one very important risk mitigation strategy. This strategy is ensuring that there is strict input validation. Another mitigation strategy is to implement security measures during development and not implemented as an after thought once the web application is completed.