Security systems in Data-Driven Applications

Kit Symmon Kessey L. Rojas

Introduction

Data-driven apps operate on a diverse set of data (spatial, documents, sensor, transactional, etc.) pulled from multiple different sources, often in real-time and create value from that data in very different ways to traditional applications. For example, they may use Machine Learning to make real-time recommendations to customers or detect fraudulent transactions. Or use Graph analytics to identify influencers in a community and target them with specific promotions or perhaps use spatial data to keep track of deliveries.

These apps are also frequently deployed on multiple platforms, including mobile devices as well as standard web browsers, which means they need a flexible, scalable and reliability deployment platform. Given the demands on these apps, they need to be continuously developed to adapt to new use cases or user needs, and all updates must happen online as they have to be available 24×7.

Now Security Systems will come and will play a big role in securing these set of data's. It's a concept that encompasses every aspect of information security from the physical security of hardware and storage devices to administrative and access controls, as well as the logical security of software applications. It also includes organizational policies and procedures.

Objectives

Defining Security Objectives

Security Objectives are the targets the customer establishes for their security program. Without security objectives, they do not know what they are trying to accomplish for security and therefore will not reach any goals. Security Objectives are one of those areas requiring the customer's involvement, and so the assessment team cannot make up the information.

List of security system obejctives

- ### Confidentiality

  This prevents the disclosure of sensitive information to unauthorized users or systems on computer networks. Sensitive information refers to the information that should be kept confidential. Loss of confidentiality leads to the unauthorized disclosure of sensitive information. In literature, confidentiality is used to provide data confidentiality and privacy. Data confidentiality prevents unauthorized entities from accessing confidential information whereas privacy ensures entities can control or influence information related to them.

- ### Integrity

  In computer networks and systems, the term integrity covers both data and systems. Generally, integrity assures the accuracy and consistency of data and systems, which means guarding against improper modification or destruction of data and systems in an unauthorized or undetected manner. A loss of integrity is the unauthorized change or destruction of data or systems.

  Data integrity assures that data are modified only in a specified and authorized manner on computer networks and systems.

- Availabilty

  This objective ensures that computer networks and systems work properly and services are accessible and are not denied for authorized users. Specifically, availability ensures timely and reliable access to information and services on computer networks and systems. A loss of availability leads to the disruption of access to the information and services on the systems.

Functionality and Features

- security-related features, functions, mechanisms, services, procedures, and architectures implemented within organizational information systems or the environments in which those systems operate.

What types of security features should good application development software include?

Here are the some of the key security features that are absolutely essential in professional business application development software:

- Application-level security

Application-level security is pretty straightforward: It lets you control application access on a per-user role, or per-user basis. This typically includes a role-based menuing system, which displays different menu options to different users based on their role.

Why is this so important? Unless every employee in your organization should have access to every application, application-level security is a must-have. For instance, your CEO might have access to all applications, while your HR department can only access applications related to HR.

- Single sign-on

Single sign-on (SSO) is a session/user authentication process that lets users enter their name and password in only one place, and access multiple related applications. It authenticates the user for all the applications they're authorized to access and eliminates login prompts when switching between applications in a single session.

Why is this so important? SSO reduces the number of passwords end users must remember, and cuts down on "forgotten password" support requests. It also improves end user productivity as users no longer need to log in to each new application.

- User privilege parameters

User privilege parameters are used to personalize features and security to individual users or user roles. These user privilege parameters are saved to a user's profile and accessible throughout every application.

Why is this so important? User privilege parameters are incredibly flexible. They can control an application's look and feel, add or hide user options, limit user capabilities, and more. For instance, suppose your company had a customer listing application. User privilege parameters could be set to display an "Update Customer Info" button only when accessed by a manager.

- User-specific data sources

This security feature is similar to row-level security, but on a database level. It means you can build a single application that accesses different data sources depending on the user.

Why is this so important? This security feature provides flexibility, as it lets developers dictate which database each user can access. For instance, suppose two companies are merging. While employees from each company must now use the same applications, employees from Company A might need access to a local database, while employees from Company B might need access to data in a completely different database. With user-specific data sources, the application will point to the correct database based on the user.

- Application activity auditing

Application activity auditing lets developers log end-user activity for signon/signoff activities. This lets IT departments quickly see when a user has logged in, which application they accessed, and when they logged off.

Why is this so important? When managing application security, it's quite useful to know who is logged in to your system. On a non-security note, activity auditing also helps your company understand which applications are being used and which are being ignored.