To: Teresa Butterfield

From: Adam Donner

Date: 20 August 2019

Subject: Securing Microservices

As an organization we realized that based on our explosive growth, we need to move away from our current monolithic application to a microservices design. Microservices takes out the aging monolithic application and separates the application’s functions in a microservice architecture. Each one of these services will perform a task for our current application. An example of this is our current ordering system. With the new design, it will be broken up into orders, the basket and the credit card processing. All of these services will of course be communicating with one another to ensure that we do not lose functionality. We will in fact increase scalability, development time and failover. When implementing a change to microservices it is very important to discuss security, as each service talks to one another there needs to be a security layer for that exchange of information. As a development team we have outlined the key features which will be integrated into the microservices architecture.

To make a microservices application usable we will needed to establish how users will access services within it. With this in mind, we have established a dedicated server to function as an API gateway, providing a single point of entry and directing traffic into the different services. By using this technique we can secure all of our services behind a firewall, allowing the API gateway to handle the external requests and then talk to the microservices which will reside behind the firewall.

Figure 1 shows how the API gateway will communicate with the microservices from the outside world. This will be the entry point to all of the services that our application is providing. From the security point of view, the API gateway will handle the authentication and authorization from the external callers to the microservice level.

We will implement OAuth for user identity and access control. The majority of our services will require some level of access control. OAuth2 is the industry standard for user authorization. It is a framework that allows users to obtain access to a resource from a resource server using tokens. These tokens are responsible for access to the resource. The advantage of using OAuth is that we can leverage existing libraries and platforms which will accelerate the development phase of this project.

We will also be using a ‘defense in depth’ strategy to prioritize key services. Defense in depth is defined as “information assurance concept in which multiple layers of security controls (defense) are placed through an information technology system.” In other words we have identified what the most sensitive services are and will apply a number of different layers of security to them. The reason for doing so is that potential attackers to our site will need to figure out a way to beat all of our other defenses to get to the critical layers.

As discussed we will be making this change in security a priority. We feel that by implementing the above strategies we will be successful in our rollout. This will ensure we protect ourselves and our clients. We have developed a strategy, which I feel, will secure the data exchange as well as a strategy to prioritize the key services.

Sincerely,

Adam D. Donner

References

Newizze. (2019, June 14). 7 Best Practices of Microservices Security. Retrieved August 20, 2019, from <https://newizze.com/7-best-practices-of-microservices-security/>

Siraj, M. (2017, September 20). Securing Microservices: The API gateway, authentication and authorization. Retrieved August 20, 2019, from https://sdtimes.com/apis/securing-microservices-the-api-gateway-authentication-and-authorization/

Sumo Logic. Improve Security in Microservices Architecture. Retrieved August 20, 2019, from <https://www.sumologic.com/insight/microservices-architecture-security/>

Troisi, M. (2019, January 22). 8 best practices for microservices app sec. Retrieved August 20, 2019, from https://techbeacon.com/app-dev-testing/8-best-practices-microservices-app-sec