To: Professor Itskovich

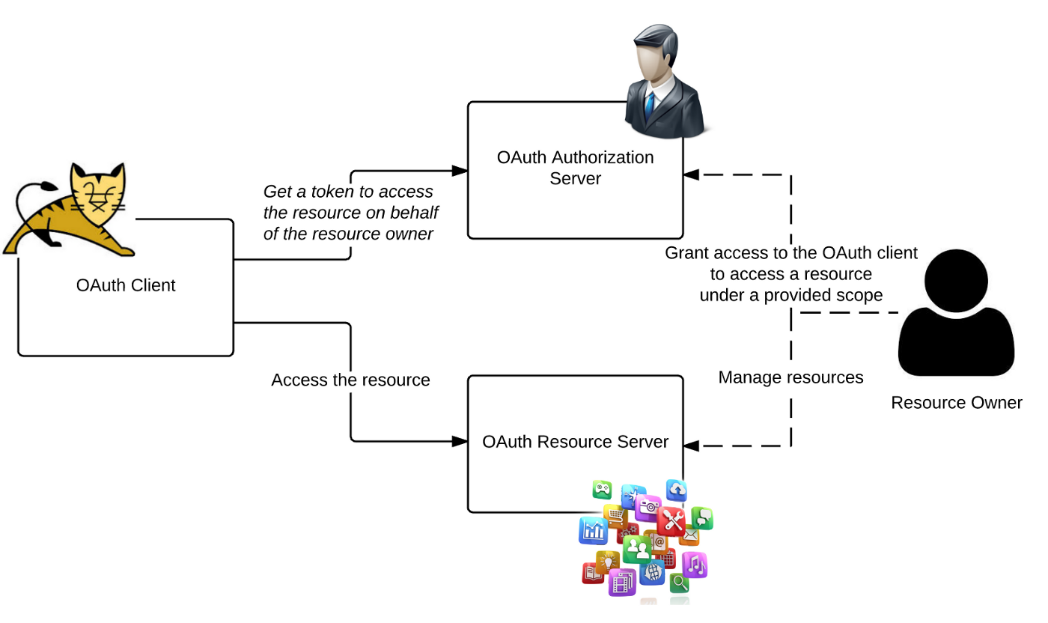
From: Becca Buechle

Date: 26 August 2020

Subject: Security in Microservices Architecture

Hey team, it has come to my attention that security in Microsystem architecture is very important and that we really need to start figure out ways to implement it in our projects. These will project the users and the clients against attacks on the application which is a win win for everyone. Below I will go into detail on a few things that we can do.

One security service that we can use is called OAuth 2.0 which is for user identity and access control. OAuth is a framework for Access delegation. The client or user that wants to access the microservice must get a valid OAuth from the API gateway to be able to access the service. ( Siriwardena, P.) This can be done by the system or by itself on behalf of the user. (Siriwardena, P.) The diagram below shows how this OAuth works source : (Siriwardena, P.).



OAuth doesn’t share any password data but instead uses an authorization token which are used between the consumers and the service providers. OAuth lets you authenticate which approved applications you want interactive with one another on the user’s behalf without sharing the password. (Sobers, R.) Another security option we have would be access control. Each microservice would be able to choose the criteria in which it allows access to the operations. (Siriwardena, P) Things such as if a user is allowed to preform the given action on that particular resource, this shows us if this user has too much or to little power depending on the set permissions. (Siriwardena, P) These checks are considered an authorization check. Each resource can define who can preform which actions and the declaration of the required permissions for each resource can be done in several ways. (Siriwardena, P). Lastly, using embedded PDP. While embedded PDP can have some drawbacks to it such as performance and PIP (policy information points). However, this also has a lot of pro’s because policies are defined centrally and are stored and evaluated at the service level. Whenever there is a new policy update it is then publishing an event to a topic. The microservice acts as an event consumer and then registers for each event that is of interest. (Siriwardena, P). When the topic is received it pulls the corresponding policy for the PAP and then updates the embedded PDP. (Siriwardena, P).

To sum it up I think it might take a little bit of time to implement but I think adding OAuth, access control, and embedded PDP would be greatly beatifical to both the user and the client. It would make sure that passwords and other information wasn’t accessed when it shouldn’t be. It would make sure each of there were only accessed when they are allowed to be. Thanks for taking the time to read this and I look forward to any questions or feedback you might have!

References

Siriwardena, P. (2016, April 11). Securing Microservices (Part I). Retrieved August 26, 2020, from https://medium.facilelogin.com/securing-microservices-with-oauth-2-0-jwt-and-xacml-d03770a9a838

Sobers, R. (2018, August 30). What is OAuth? Definition and How it Works. Retrieved August 26, 2020, from https://www.varonis.com/blog/what-is-oauth/