Distributed Authentication Mesh*

Declarative Adhoc Conversion of Credentials

Christoph Bühler[1]

Spring Semester 2021

Abstract

foo bar stuff

Contents

List of Tables

List of Figures

[1] University of Applied Science of Eastern Switzerland (OST)

*todo

1 Introduction

Modern cloud environments solve many problems like the discovery of services and data transfer or communiation between services in general. One modern way of solving service discovery and communication is a Service Mesh, which introduces an additional infrastructure layer that manages the communication between services (Li et al. 2019, sec. 2).

However, a specific problem is not solved yet: "dynamic" trusted communication between services. When a service, that is capable of handling OpenID Connect (OIDC) credentials, wants to communicate with a service that only knows Basic Authentication that originating service must implement some sort of conversion or know static credentials to communicate with the basic auth service. Generally, this introduces changes to the software of services. In small applications which consist of one or two services, implementing this conversion may be a feasable option. If we look at an application which spans over a big landscape and a multitude of services, implementing each and every possible authentication mechanism and the according conversions will be error prone work and does not scale well[1].

The goal of the project "Distributed Authentication Mesh" is to provide a solution for this problem. TODO.

2 Definitions and Boundaries

This section provides information about the project. It gives an overview of the context as well as terminology and general definitions.

2.1 Context

This project aims at the specific problem of declarative conversion of credentials to ensure authorized communication between services. The solution may be runnable on various platforms but will be implemented according to Kubernetes standards. Kubernetes[2] is an orchestration platform that works with containerized applications.

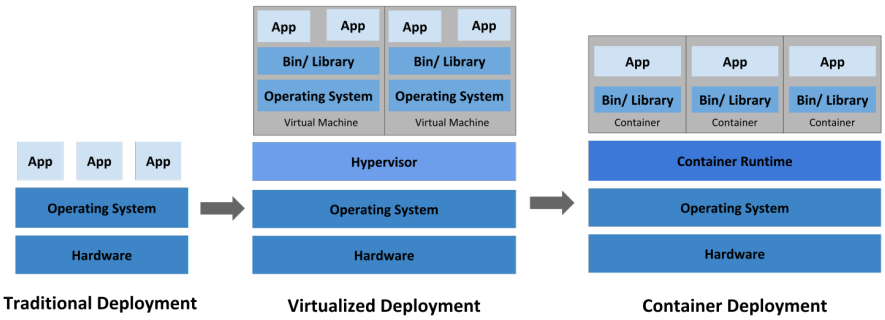
The deliverables of this project may aid services to communicate with each other despite different authentication mechanisms. As an example, this could be used to enable a modern web application that uses OpenID Connect (OIDC) as the authentication and authorization mechanism to communicate with a legacy application that was deployed on the Kubernetes cluster but not yet rewritten. This transformation of credentials (from OIDC to Basic Auth) is done by the solution of this project instead of manual work which may introduc code changes to either service.

To use the proposed solution of this project, no service mesh or other complex layer is needed. The solution runs without those additional parts on a Kubernetes cluster. To provide service discovery, the default internal DNS capabilities of Kubernetes are sufficient.

[1] According to the matrix problem: X services times Y authentication methods
[2] https://kubernetes.io/

2.2 Kubernetes

2.2.1 What is Kubernetes



briefly describe kubernetes
-> fetched from github repo with CCBY4.0 lizenz!

2.2.2 Operator

briefly describe what an operator is

2.2.3 Service Mesh

briefly describe a service mesh.

2.3 Authentication and Authorization

2.3.1 Basic

2.3.2 OpenID Connect (OIDC)

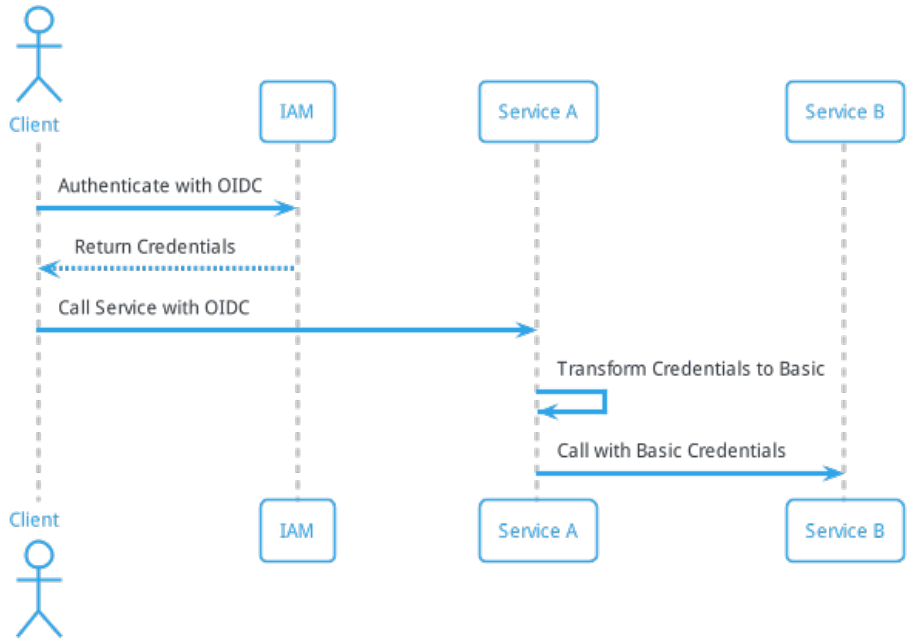
briefly describe OIDC

3 State of the Art, the Practice and Deficiencies

This section gives an overview over the current state of the art as well as the deficiencies according to the author. Following the description of the current situation, a definition of the should situation gives an overview of the purposed solution.

- Describe the IS situation.
- Describe the SHOULD situation.
- describe service mesh (image from referecne could be used)

Test:



4 Grober Roter Faden Projektbericht

1. Introduction
   - Das Erreichte und den Überblick der Arbeit wiedergeben
   - Erklären wie die Arbeit aufgebaut ist und welche Details wo zu finden sind
2. Einführung ins Thema
   - Leser ausbilden
   - Backgroundinformationen liefern
   - Context der Arbeit
   - Begrifflichkeiten erklären
3. IST / SOLL Beschreibung
   - Was gibt es
   - Wie arbeit die Leute
   - Wo sind die Defizite
4. Wie lösen wir die Probleme
   - Planung der Software (Diagramme etc.)
   - Wie die Lösung aussieht
   - Umsetzung der Konzepte
5. Nachweis
   - Beweis anführen, dass das vorgeschlagene Konzept funktioniert
   - Beispiele nutzen um Leute abzuholen
6. Conclusion
   - Ausblick (Referenz auf weitere Projektarbeit)
   - Was man gemacht hat (klassischer Paperdiamant)

5 Todos

Todos:

- nope.

Bibliography

Li, W., Y. Lemieux, J. Gao, Z. Zhao, and Y. Han. 2019. "Service Mesh: Challenges, State of the Art, and Future Research Opportunities." In *2019 IEEE International Conference on Service-Oriented System Engineering (SOSE)*, 122–25. https://doi.org/10.1109/SOSE.2019.00026.