**User Authentication: Identity-Aware Proxy**

**Overview**

In this lab, you build a minimal web application with Google App Engine, then explore various ways to use Identity-Aware Proxy (IAP) to restrict access to the application and provide user identity information to it. Your app will:

* Display a welcome page
* Access user identity information provided by IAP
* Use cryptographic verification to prevent spoofing of user identity information

What you'll learn

* How to write and deploy a simple App Engine app using Python
* How to enable and disable IAP to restrict access to your app
* How to get user identity information from IAP into your app
* How to cryptographically verify information from IAP to protect against spoofing

Prerequisites

A basic knowledge of the Python programming language will enhance your learning experience.

This lab is focused on Google App Engine and IAP. Non-relevant concepts and code blocks are glossed over and are provided for you to simply copy and paste.

**Introduction to Identity-Aware Proxy**

Authenticating users of your web app is often necessary, and usually requires special programming in your app. For Google Cloud apps you can hand those responsibilities off to the [Identity-Aware Proxy](https://cloud.google.com/iap/) service. If you only need to restrict access to selected users there are no changes necessary to the application. Should the application need to know the user's identity (such as for keeping user preferences server-side) Identity-Aware Proxy can provide that with minimal application code.

What is Identity-Aware Proxy?

Identity-Aware Proxy (IAP) is a Google Cloud service that intercepts web requests sent to your application, authenticates the user making the request using the Google Identity Service, and only lets the requests through if they come from a user you authorize. In addition, it can modify the request headers to include information about the authenticated user.

**Note:**App Engine has its standard and flexible environments which are optimized for different application architectures. Currently, when enabling IAP for App Engine, if the Flex API is enabled, Google Cloud will look for a Flex Service Account. Your lab project comes with a multitude of APIs already enabled for the purpose of convenience. However, this creates a unique situation where the Flex API is enabled without a Service Account created.