Single Sign-On Authentication

1. Background

Single Sign-On (SSO) has gained widespread acceptance for online authentication because of its user-friendly nature and enhanced security features. However, it also poses a risk as it creates a single point of failure, with all service providers (SPs) relying entirely on the user identity verified by the SSO identity providers (IdPs). Existing research indicates that user identity on IdPs can change frequently, leading to inconsistencies between the user identity and their account. This inconsistency provides a vector for attackers to compromise accounts and steal sensitive information. It is crucial to demonstrate and address this vulnerability to enhance defenses against such a security threat.

2. Project Description

This project aims to implement an SSO authentication framework that demonstrates the aforementioned identity-account inconsistency threat. Specifically, the framework consists of two major components:

1. Service Provider (SP): A web service that enables user account registration and authenticates users to their corresponding accounts using SSO. The SP will also demonstrate how user accounts can be compromised with a reused identity token from the IdP.
2. Identity Provider (IdP): An email provider that supplies user identity tokens to the SP. The IdP should allow users to send and receive emails, as well as create, modify, and delete users email accounts.

2.1. Project Goals

1. Develop an SSO framework by either implementing SPs and IdPs or modifying existing open-source software to simulate both components.
2. Demonstrate that adversaries can easily compromise user accounts by reusing user identity.
3. Design effective mitigations to defend against such security threats, demonstrating their effectiveness and deployability.
4. Provide full documentation of the design process and user guidance on how to use the designed framework.

2.2. Other Requirements

1. The designed framework should be self-contained and able to function without an Internet connection.
2. The SP and IdP should be able to run on both virtual machines and containers.
3. The mitigation should be easily switchable on or off for demonstration purposes, and the overall design of the mitigation should be easily explainable.

2.3. Ethical Requirement

All participating students must consent to an ethical agreement stating that concepts, knowledge, or skills acquired from this project will not be used in any harmful way in real-world scenarios.

3. Desired Skill Set

1. One of the following programming languages: Python, JavaScript, PHP, HTML
2. Taking/Taken related courses is a plus (CSCI 442, 455, 471, 474, 475)
3. Familiar with SSO authentication is a plus
4. Familiar with virtual machines and container is a plus

4. Preferred Team Size: 3-5 students

5. Location of Work: Mines Campus

6. Client Liaison: Guannan Liu (guannan.liu@mines.edu)

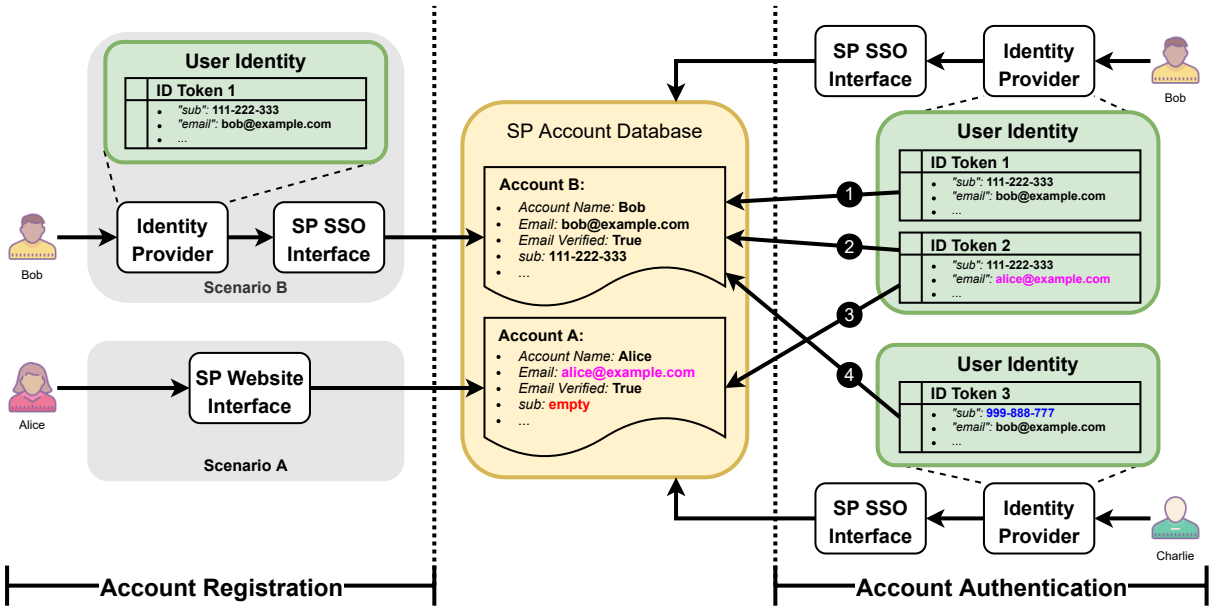

Fig. 1. SSO Framework Overview and Identify-Account Inconsistency Threat Model

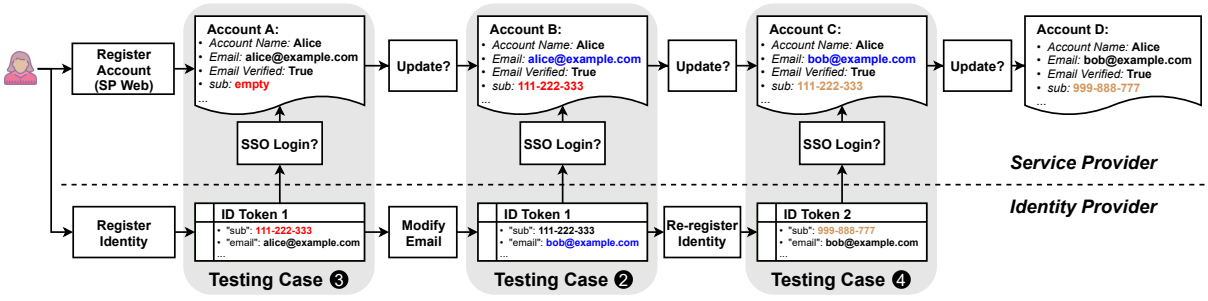

Fig. 2. Attack Vectors of Identify-Account Inconsistency