Module 4

Entity and User Authentication

Topics to be covered

- To distinguish between message authentication and entity authentication
- To define witnesses used for identification
- To discuss some methods of entity authentication using a password
- To introduce some challenge-response protocols for entity authentication
- To introduce some zero-knowledge protocols for entity authentication
- To define biometrics and distinguish between physiological and behavioral techniques

Introduction

- ***Entity authentication is a technique designed to let one party prove the identity of another party.***
- ***An entity can be a person, a process, a client, or a server.***
- ***The entity whose identity needs to be proved is called the claimant; the party that tries to prove the identity of the claimant is called the verifier.***

Data origin vs Entity Authentication

- ***There are two difference between message authentication (data-origin authentication), discussed in earlier and entity authentication to be discussed***

1) ***Message authentication might not happen in real time; entity authentication does.***
2) ***Message authentication simply authenticates one message; the process needs to be repeated for each new message. Entity authentication authenticates the claimant for the entire duration of a session.***

Verification Categories

- Something Known - password,PIN,secret key
- Something Possessed-id card, credit card, debit card
- Something Inherent-inherited,finger prints,facial characteristics

Passwords

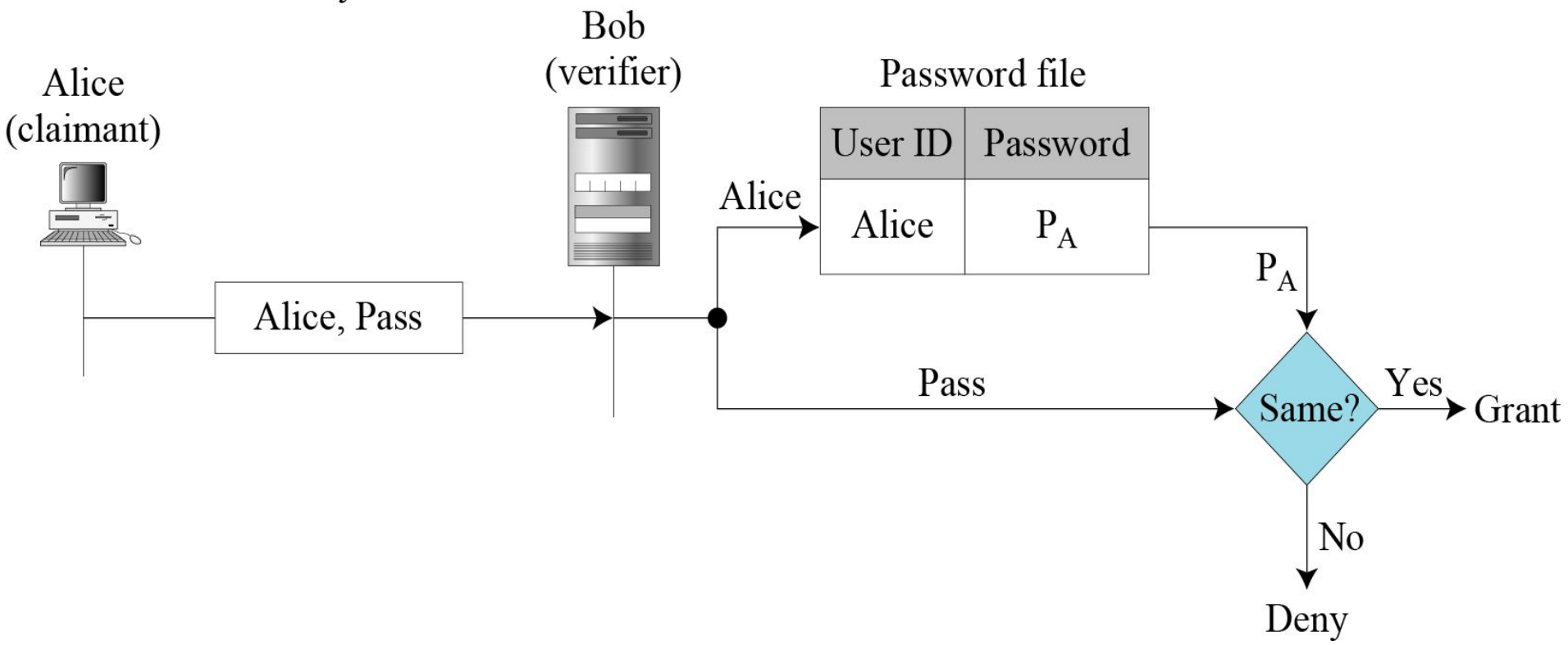
- ***The simplest and oldest method of entity authentication is the password-based authentication, where the password is something that the claimant knows.***

- ***Fixed passwords***
- ***One Time passwords***

Password-first approach

P_A: Alice's stored password

Pass: Password sent by claimant



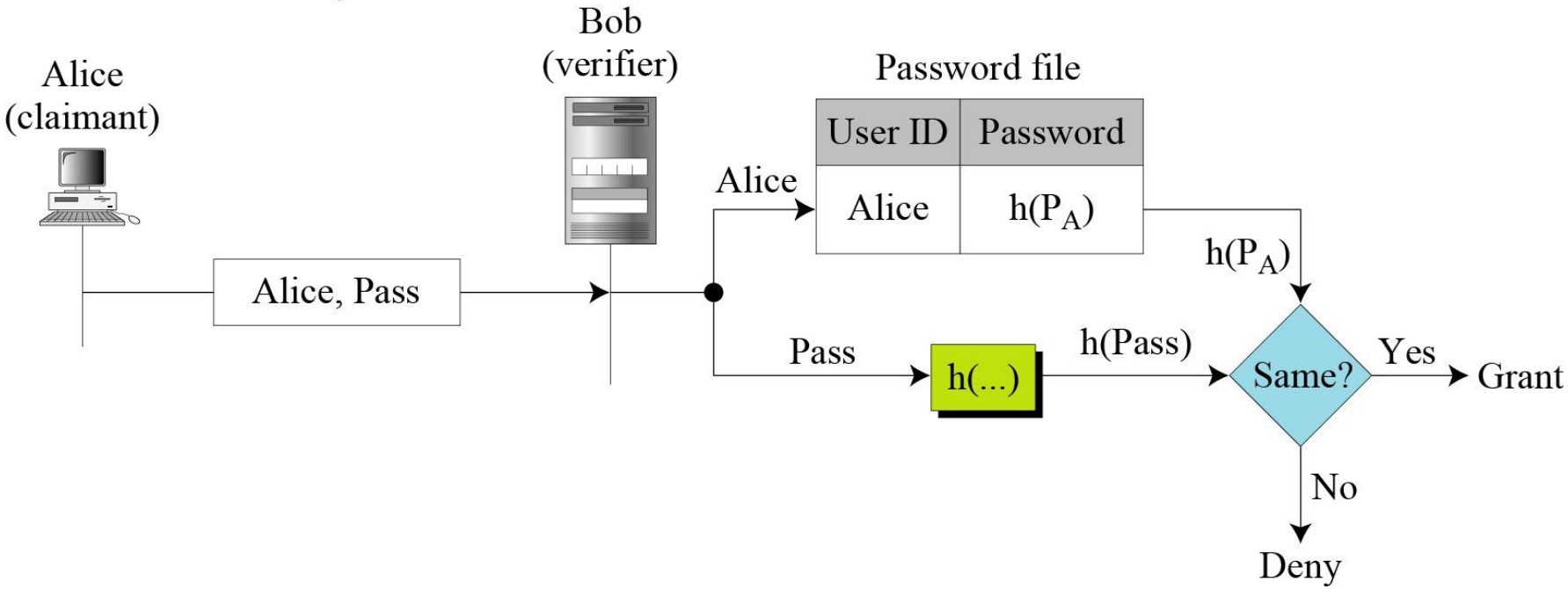
Attacks on password- first approach

- Eavesdropping
- Stealing a password
- Accessing a password file
- Guessing

Password- second approach(hashing)

P_A: Alice's stored password

Pass: Password sent by claimant



Attacks on password- Second approach

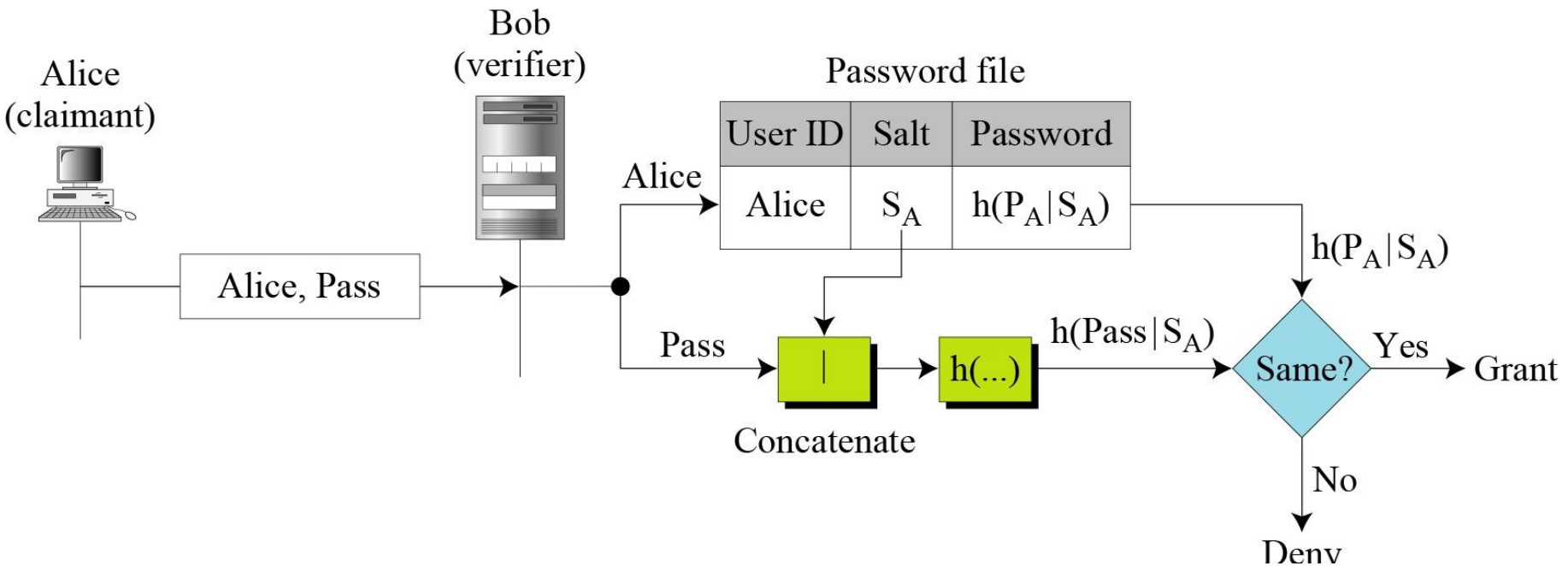
- There can be a possibility of dictionary attack
  - A dictionary attack is **a method of breaking into a password-protected computer, network or other IT resource by systematically entering every word in a dictionary as a password**.

Password- third approach-Salting

P_A: Alice's password

S_A: Alice's salt

Pass: Password sent by claimant



Password –Fourth approach

- ***In the fourth approach, two identification techniques are combined. A good example of this type of authentication is the use of an ATM card with a PIN (personal identification number).***

One time password

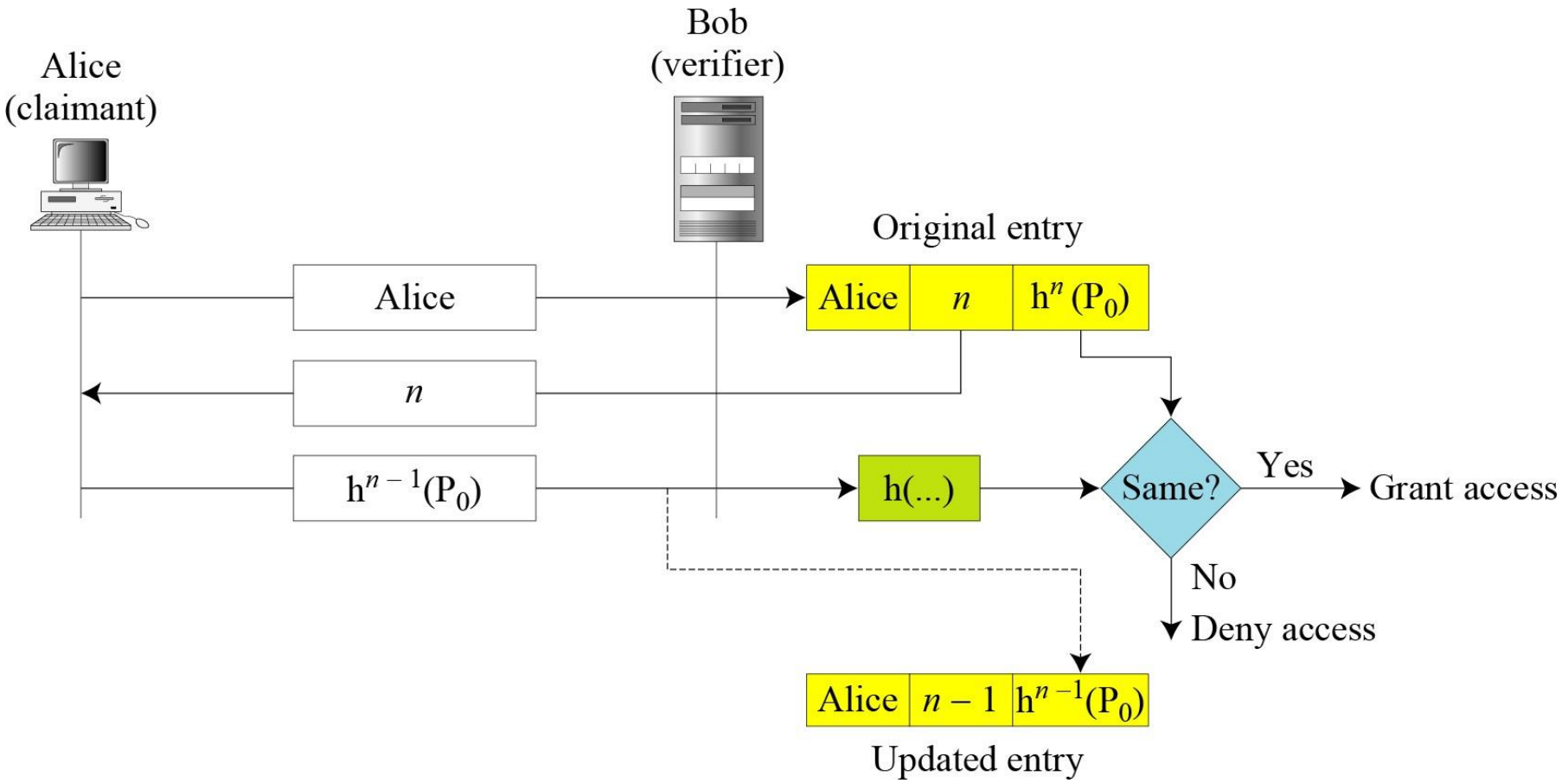
- First Approach
- In the first approach, the user and the system agree upon *a list of passwords.*

- Second Approach
- In the second approach, the user and the system agree to *sequentially update the password.*

- Third Approach
- In the third approach, the user and the system create a *sequentially updated password using a hash function.*

$$h^n(x) = h(h^{n-1}(x)) \quad h^{n-1}(x) = h(h^{n-2}(x)) \quad \ldots \quad h^2(x) = h(h(x)) \quad h^1(x) = h(x)$$

First approach



Challenge-Response

- ***In password authentication, the claimant proves her identity by demonstrating that she knows a secret, the password. In challenge-response authentication, the claimant proves that she knows a secret without sending it.***

1. **Using a Symmetric-Key Cipher**
2. **Using Keyed-Hash Functions**
3. **Using an Asymmetric-Key Cipher**
4. **Using Digital Signature**

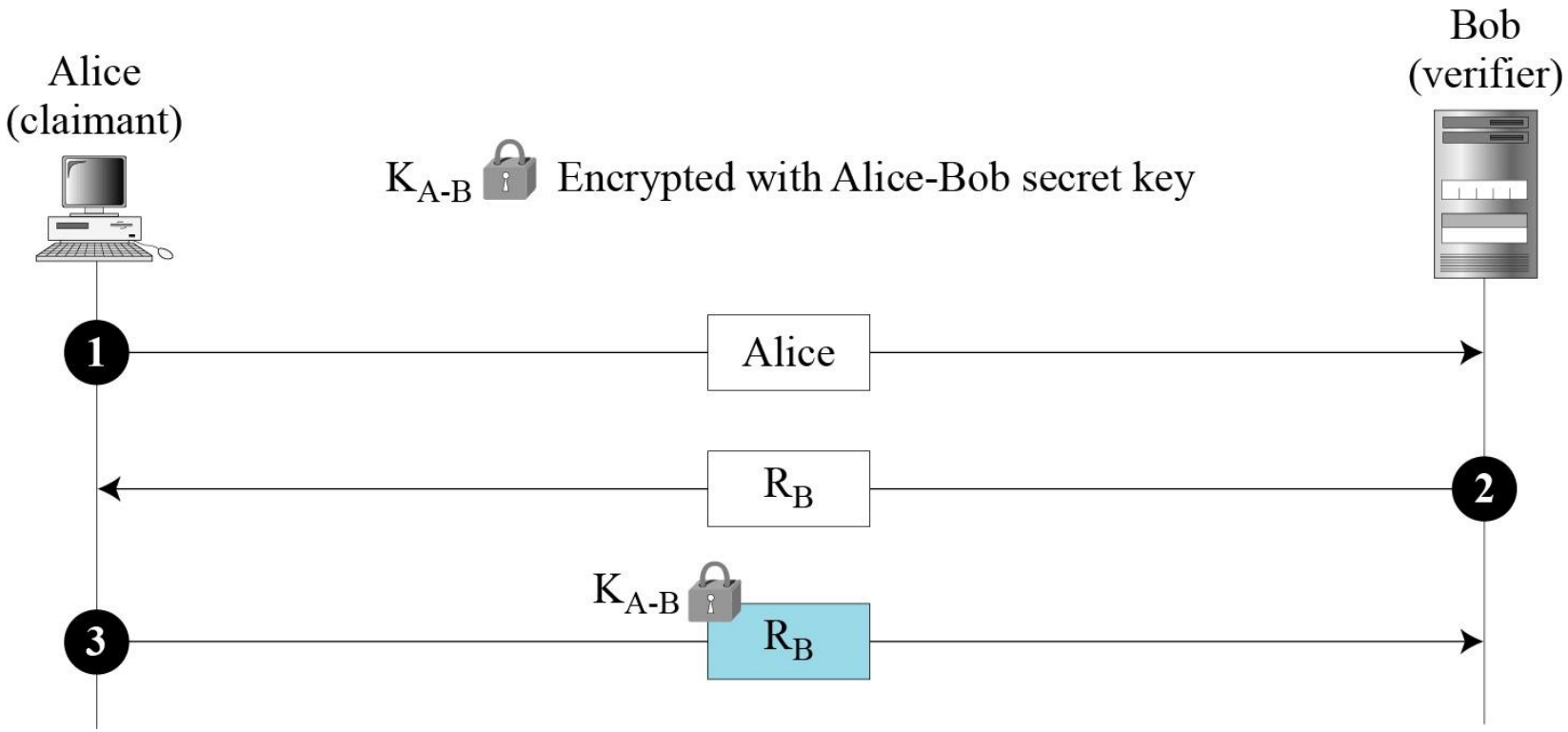
Note

In challenge-response authentication, the claimant proves that she knows a secret without sending it to the verifier.

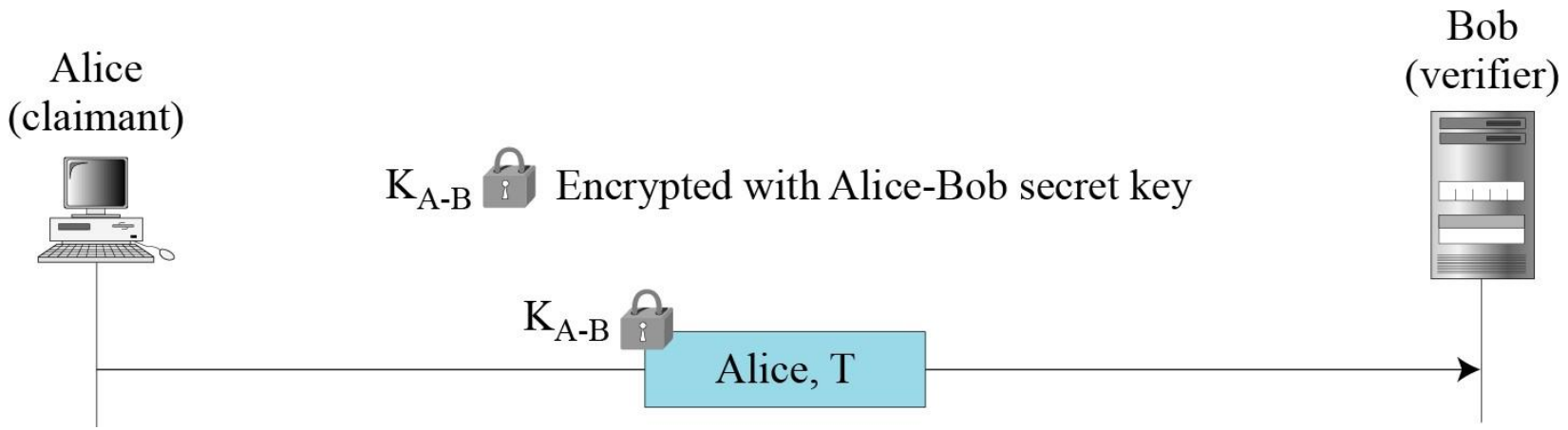
Note

The challenge is a time-varying value sent by the verifier; the response is the result of a function applied on the challenge.

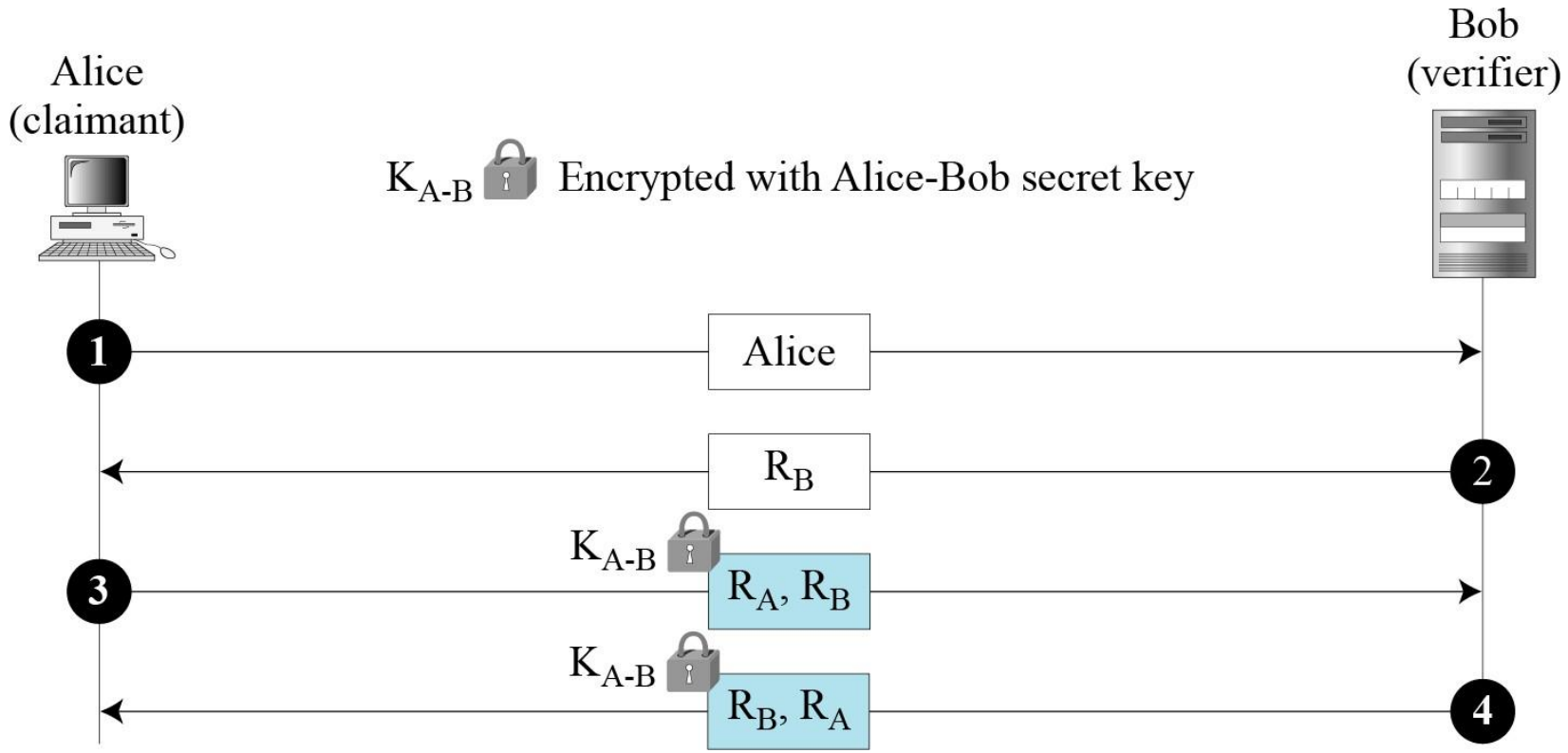
Using Symmetric Key cipher-nonce Challenge



Using Symmetric Key cipher-time stamp Challenge

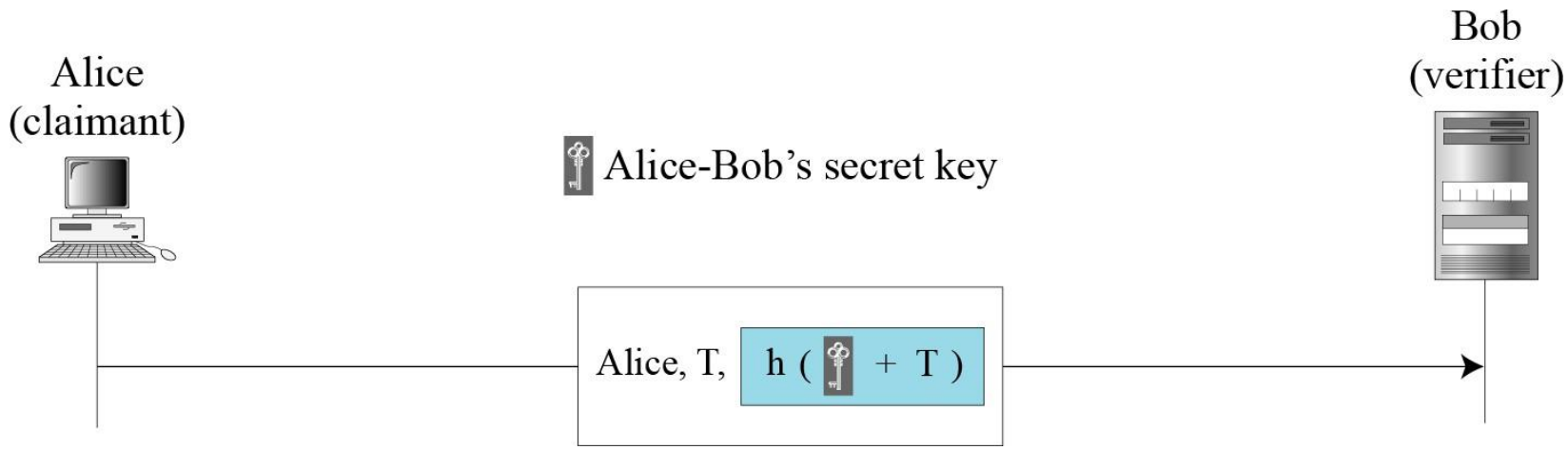


Using Symmetric Key cipher- Bidirectional Authentication

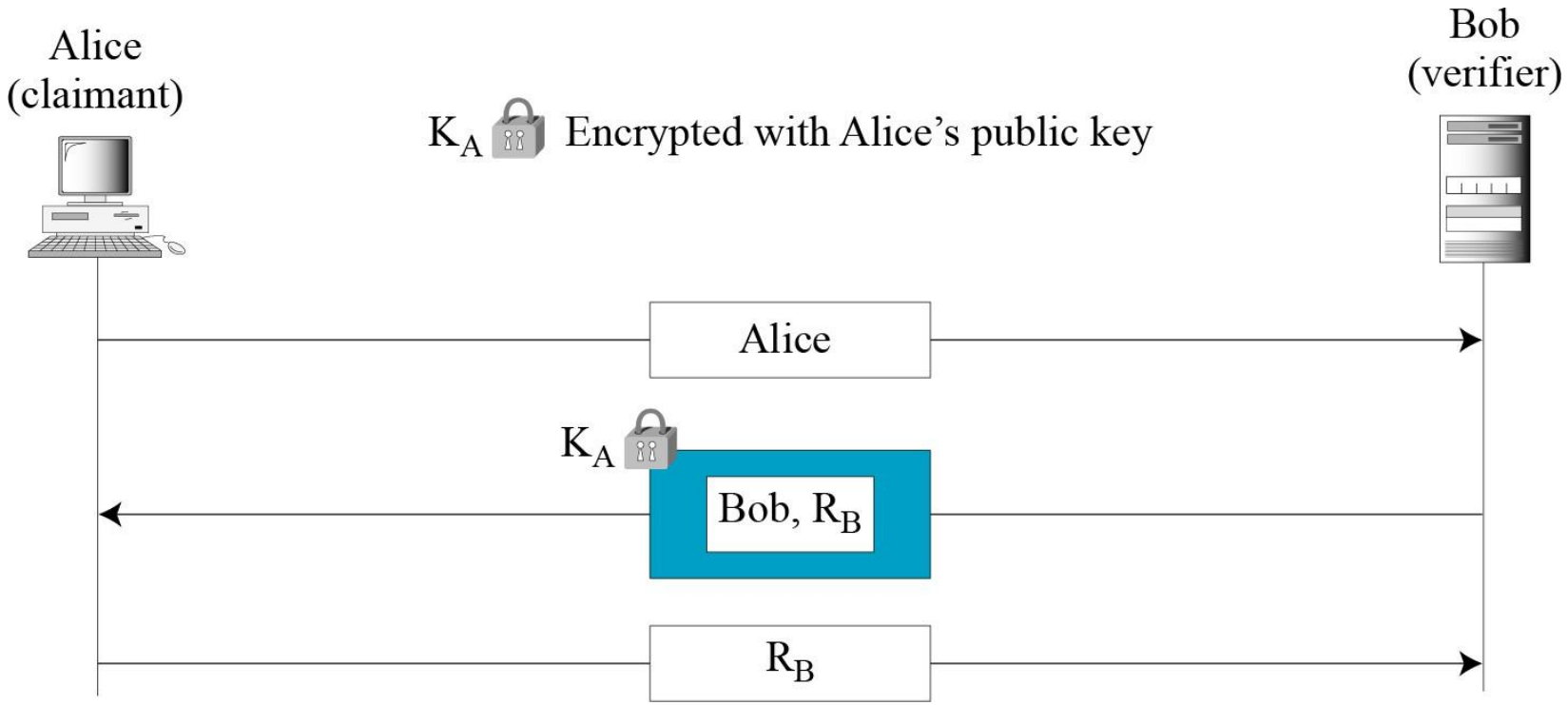


Using Keyed-Hash Functions

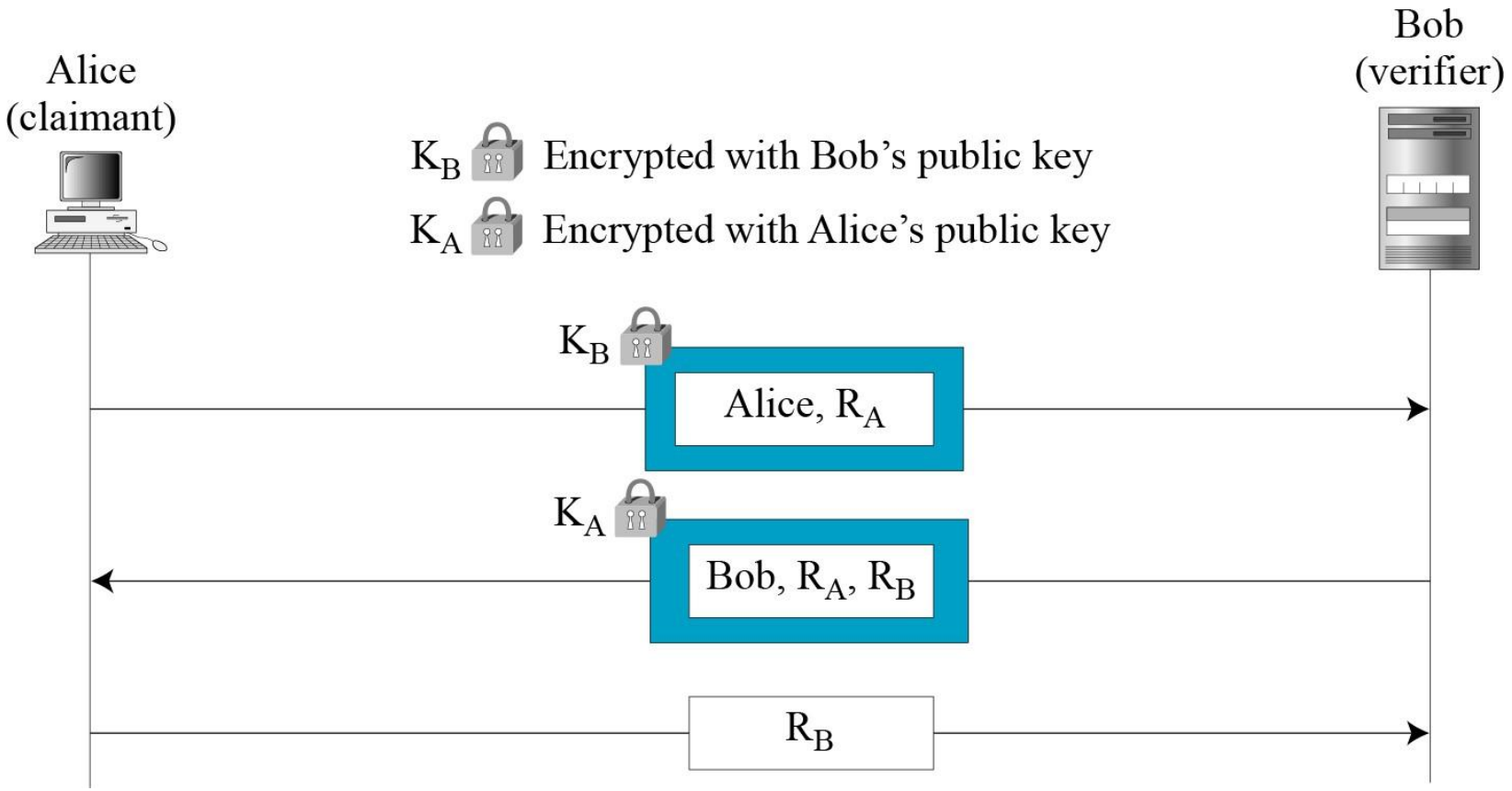
Instead of using encryption/decryption for entity authentication, we can also use a keyed-hash function (MAC).



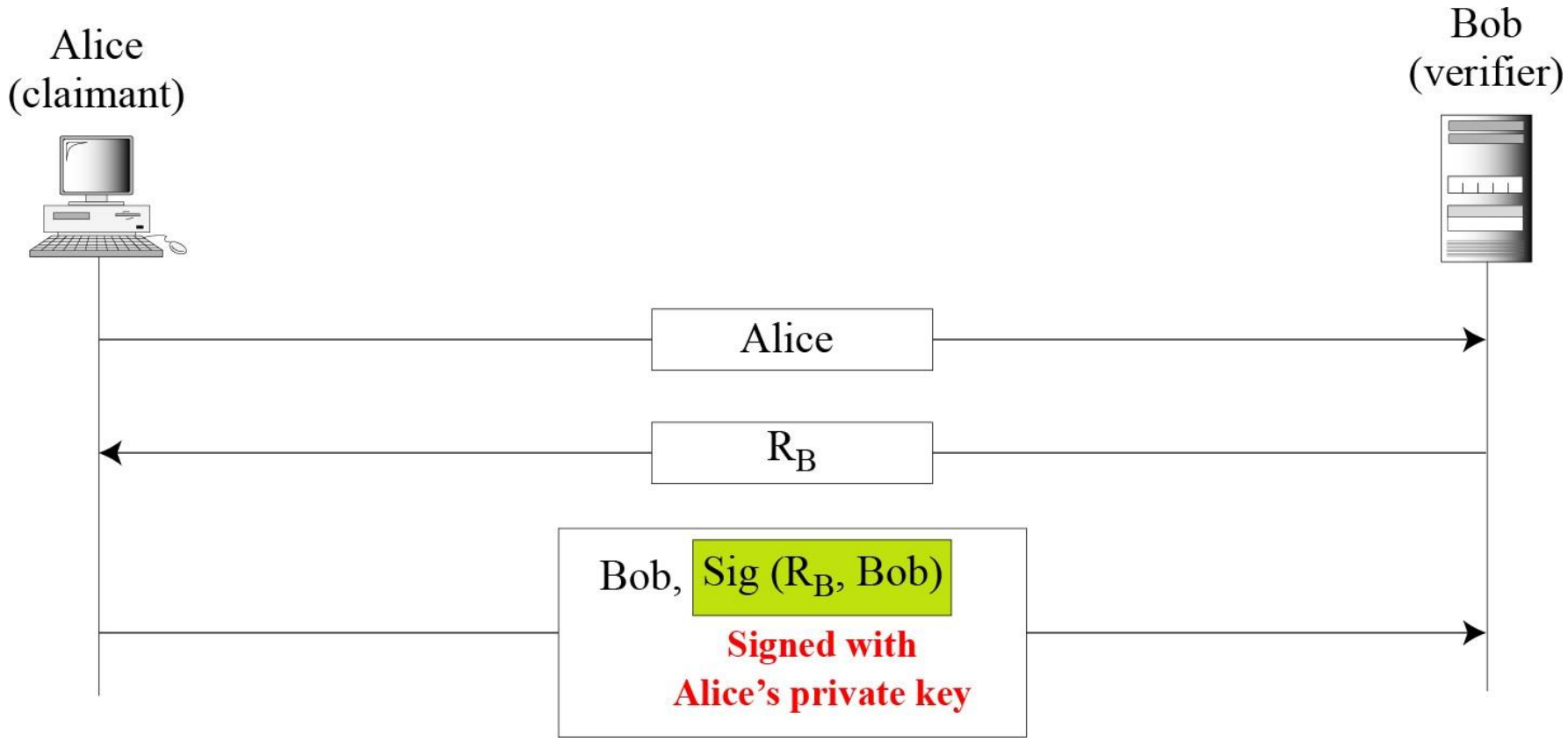
Using an Asymmetric-Key Cipher



Using an Asymmetric-Key Cipher



Using Digital Signature-unidirectional



Using Digital Signature-bidirectional