Chapter 14

Entity Authentication

Chapter 14

Objectives

- 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

14-1 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.

Topics discussed in this section:

14.1.1 Data-Origin Versus Entity Authentication

There are two differences between message authentication (data-origin authentication), discussed in Chapter 13, and entity authentication, discussed in this chapter.

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.*

14.1.2 Verification Categories

Something known

Something possessed

Something inherent

14.1.3 Entity Authentication and Key Management

This chapter discusses entity authentication. The next chapter discusses key management.

14-2 PASSWORDS

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

Topics discussed in this section:

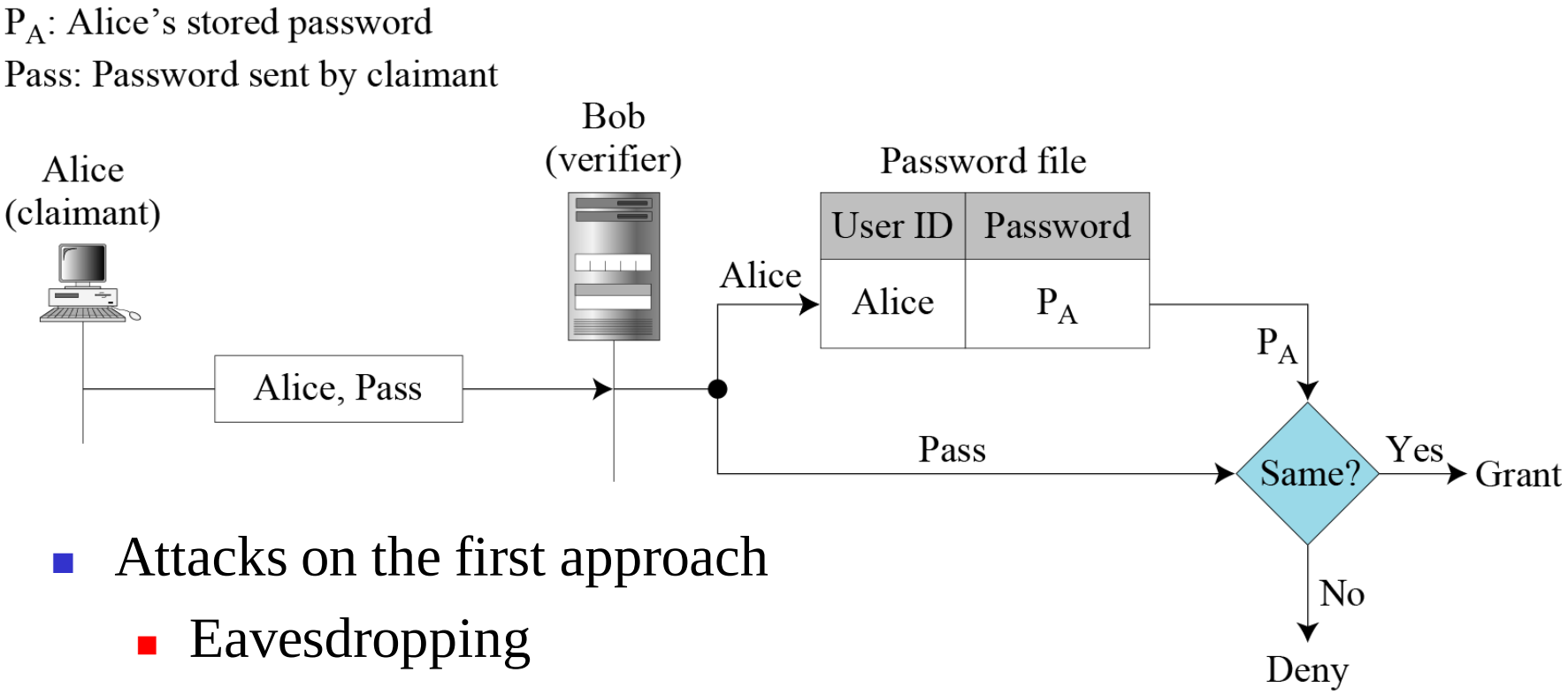
14.2.1 Fixed Password

14.2.2 One-Time Password

14.2.1 Fixed Password

First Approach

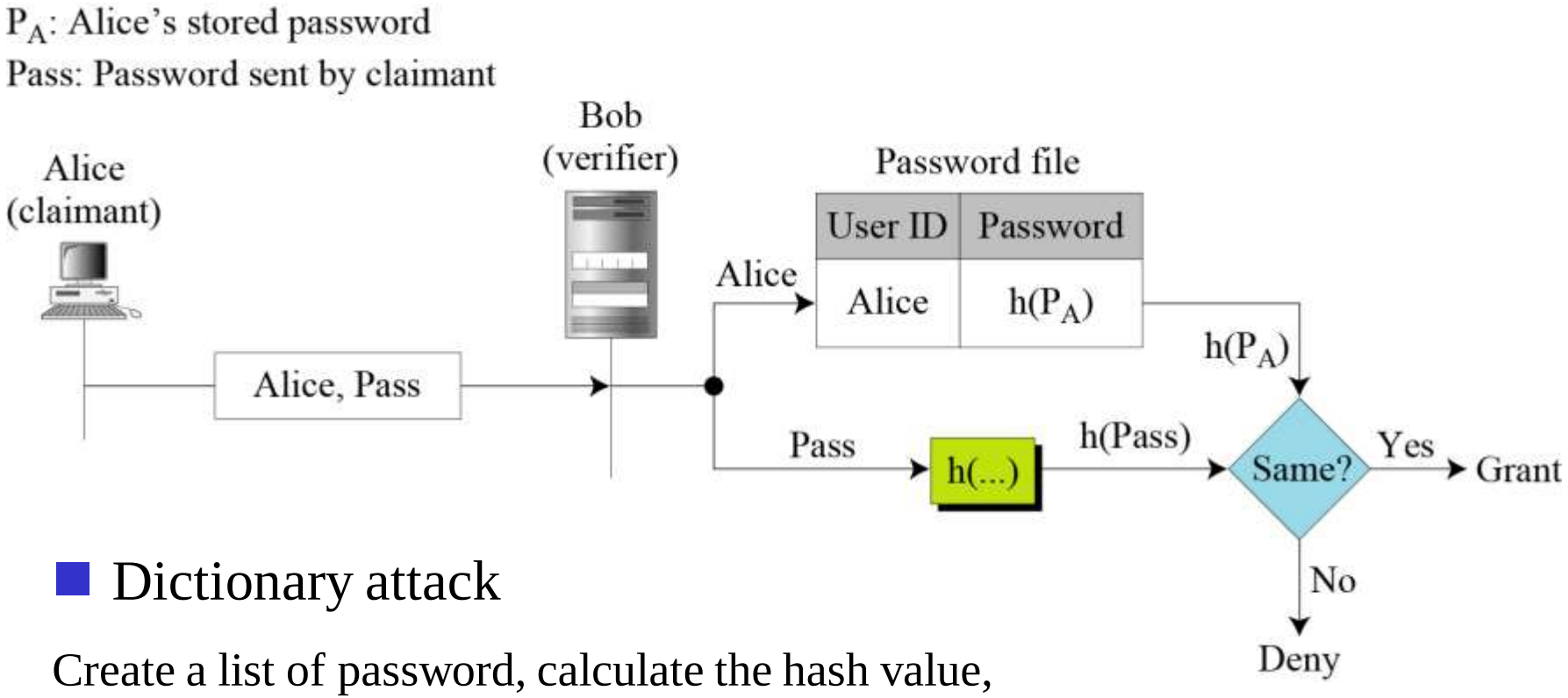
Figure 14.1 *User ID and password file*



- Attacks on the first approach
  - Eavesdropping
  - Stealing a password
  - Accessing a password file
  - guessing

Second Approach

Figure 14.2 *Hashing the password*



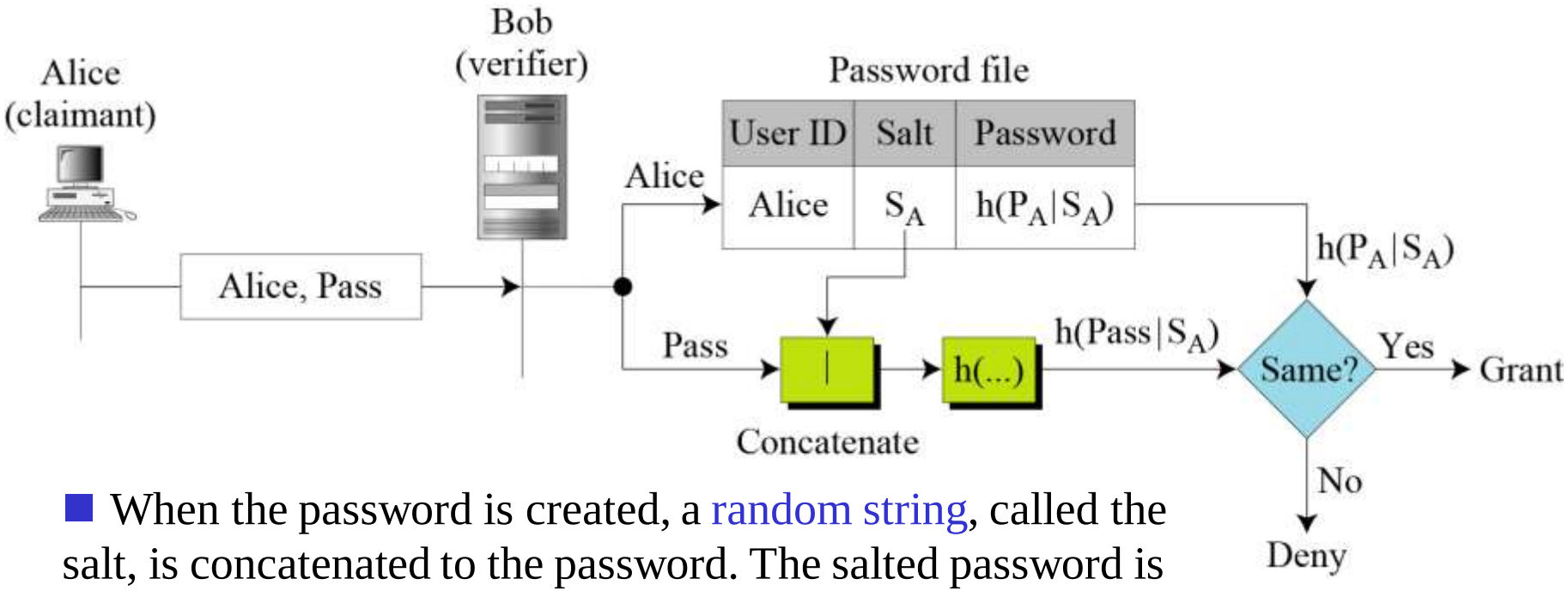
- Dictionary attack

Create a list of password, calculate the hash value, and search the second-column entries to find a match.

Third Approach

P_A: Alice's password
S_A: Alice's salt
Pass: Password sent by claimant

Figure 14.3 *Salting the password*



- When the password is created, a random string, called the salt, is concatenated to the password. The salted password is then hashed.
- The Unix OS uses a variation of this method.

14.2.1 Continued

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 (something possessed) with a PIN (personal identification number) (something known).*

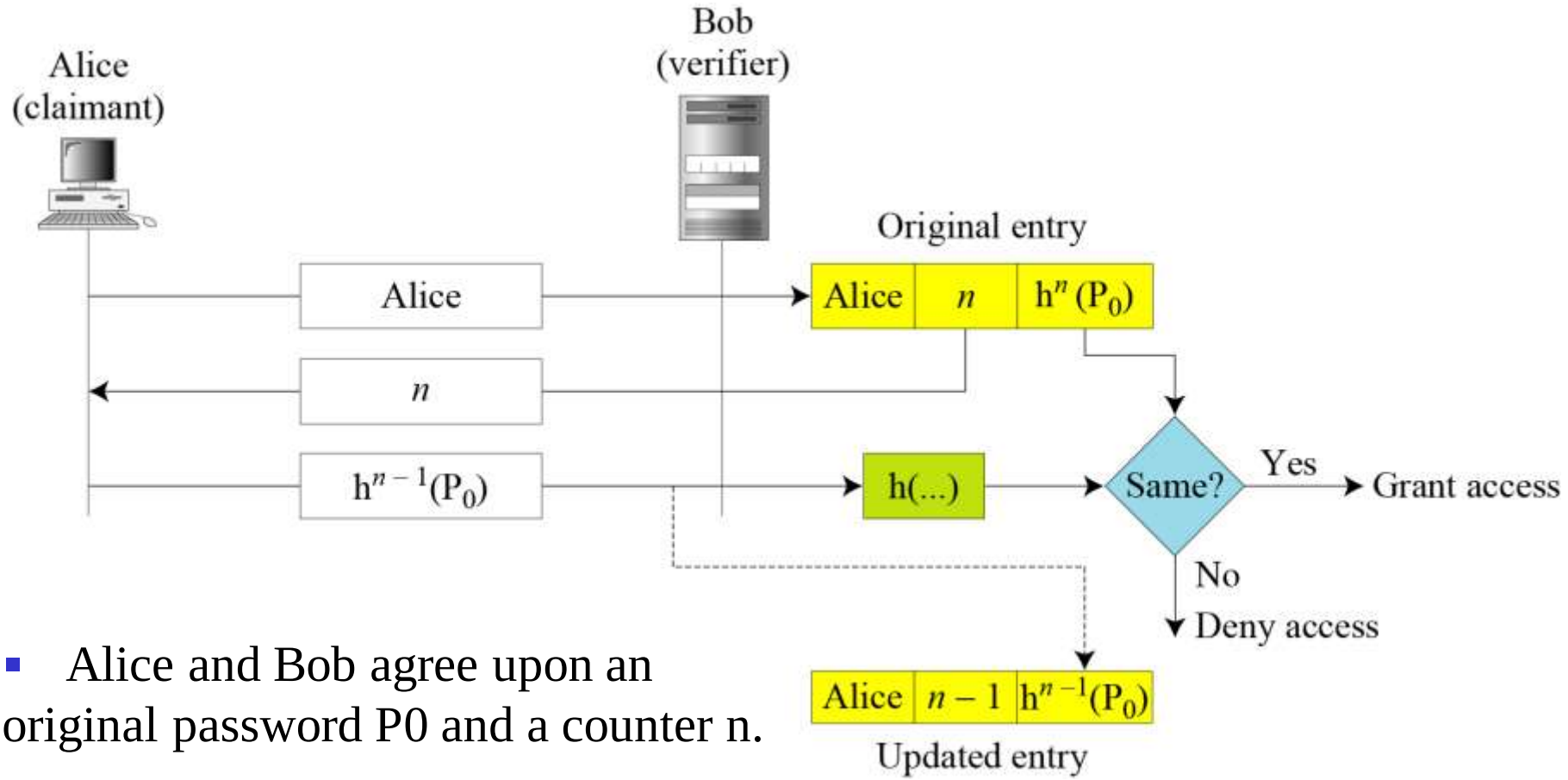
14.2.2 One-Time Password

- *A one-time password is a password that is used only once.*
- *In the first approach, the user and the system agree upon a list of passwords.*
- *In the second approach, the user and the system agree to sequentially update the password.*
- *In the third approach, the user and the system create a sequentially updated password using a hash function.*

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

14.2.2 Continued

Figure 14.4 *the third approach -- Lamport one-time password*



- Alice and Bob agree upon an original password P0 and a counter n.
- The system stores the identity of Alice, the value of n and the hash.

14-3 CHALLENGE-RESPONSE

In password authentication, the claimant proves her identity by demonstrating that she knows a secret, the password. However, because the claimant reveals this secret, it is susceptible to interception by the adversary. In challenge-response authentication, the claimant proves that she knows a secret without sending it.

Topics discussed in this section:

14-3 Continue

Note

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

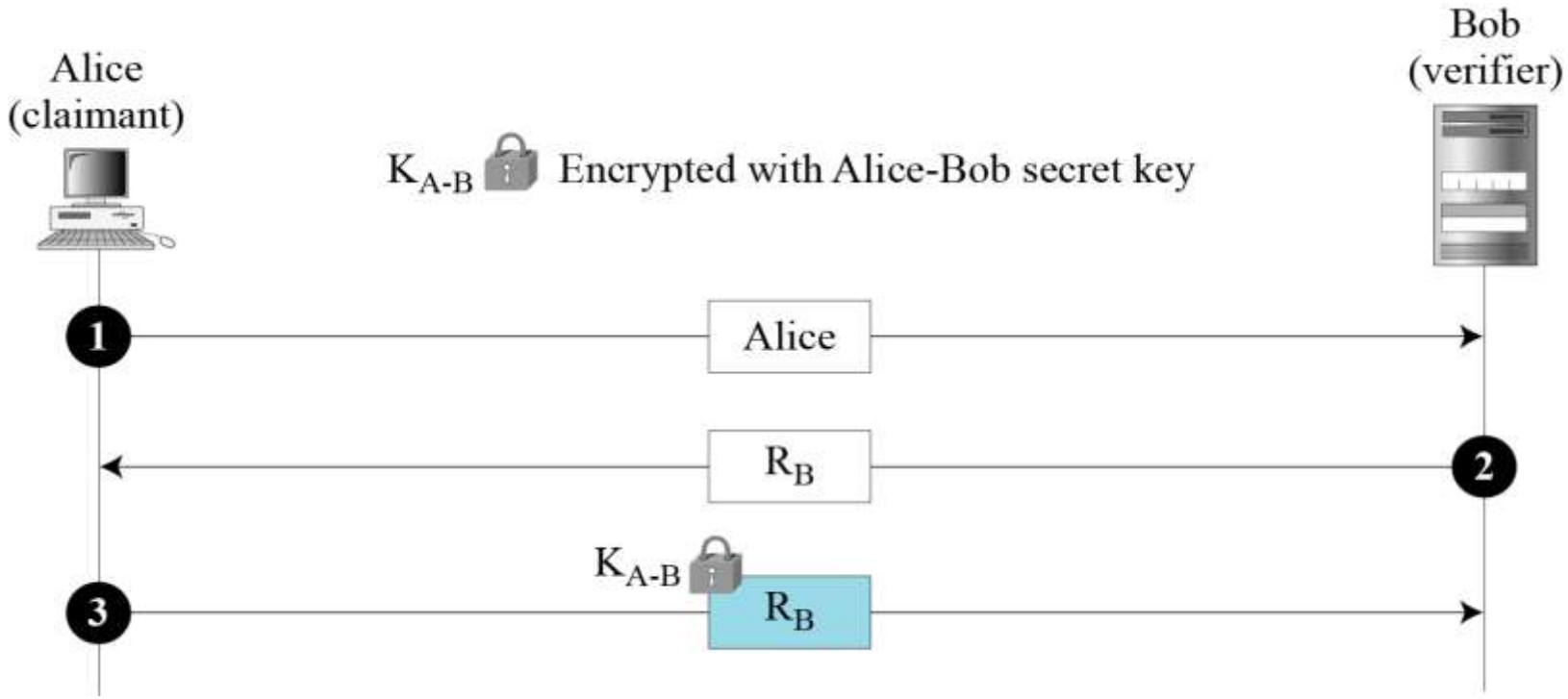
Note

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

14.3.1 Using a Symmetric-Key Cipher

First Approach

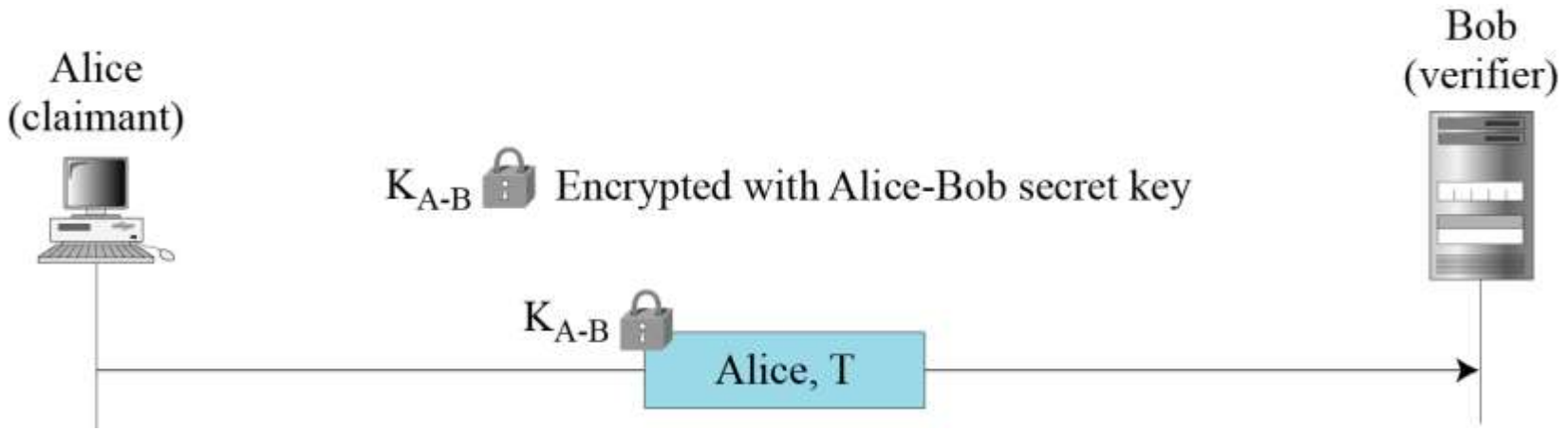
Figure 14.5 *Nonce challenge*



1. ID of claimant
2. The challenge: RB is the nonce randomly chosen by the Bob to challenge Alice
3. Alice encrypts the nonce using the shared secret key known only to Alice and Bob. Bob decrypts the message. If the nonce obtained from decryption is the same as the one sent by Bob.

Figure 14.6 *Timestamp challenge*

Second Approach

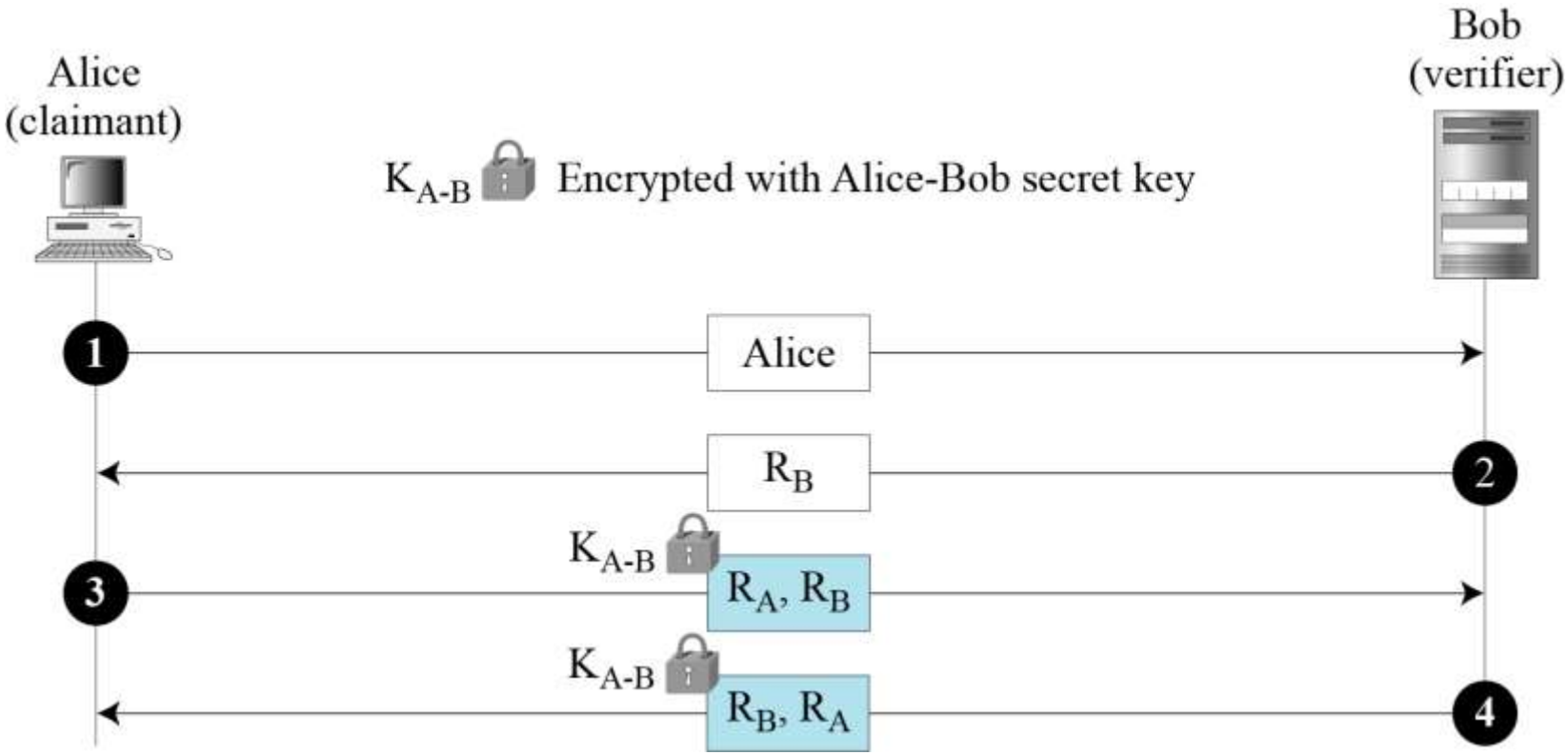


The challenge message is the current time sent from the verifier to the claimant.

The claimant encrypt Alice ID and time with Alice-Bob secret key.

14.3.1 Continued

Third Approach. **Figure 14.7** ***Bidirectional authentication***

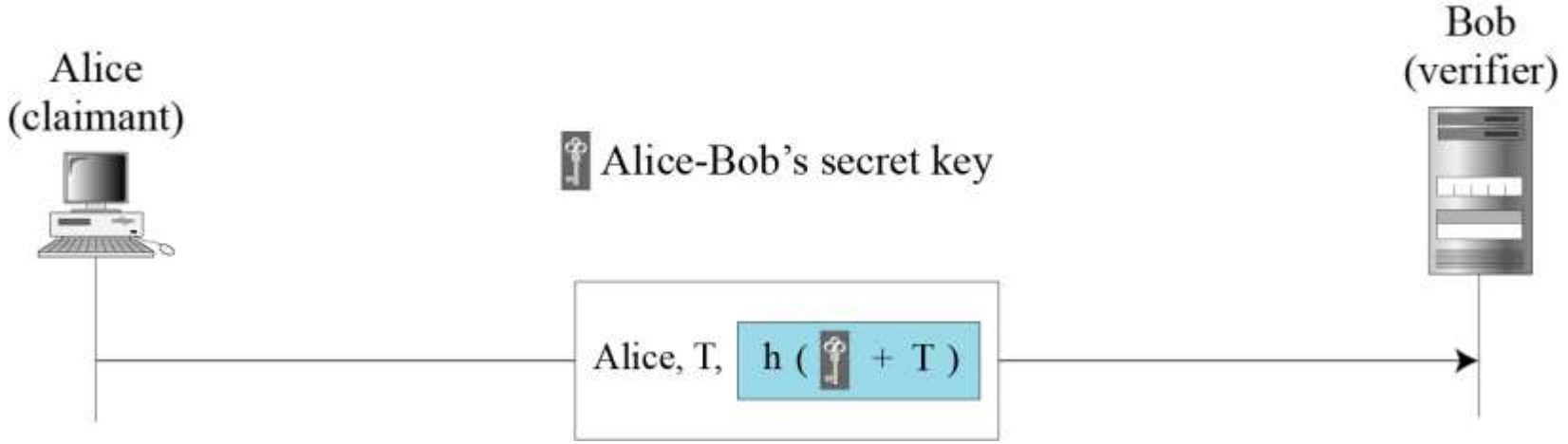


1. Alice ID
2. The challenge from Bob to Alice R_B
3. Alice respond and send her challenge R_A
4. Bob's response. R_A and R_B are switched to prevent a replay attack.

14.3.2 Using Keyed-Hash Functions

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

Figure 14.8 *Keyed-hash function*

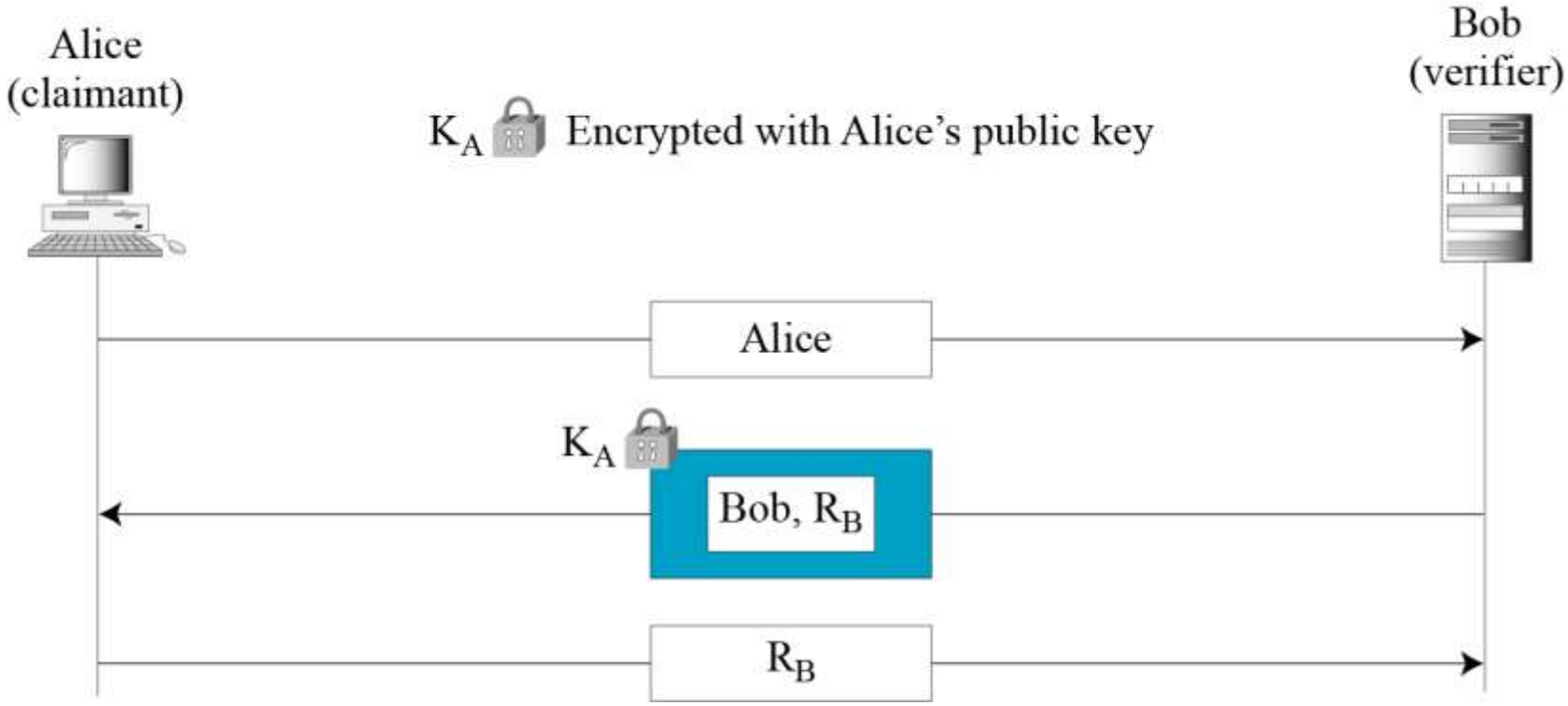


1. The challenge message is the current time sent from the verifier to the claimant.
2. The timestamp is sent both as plaintext and as text scrambled by the keyed-hash function.
3. Bob compares his calculation with what he received.

14.3.3 Using an Asymmetric-Key Cipher

First Approach

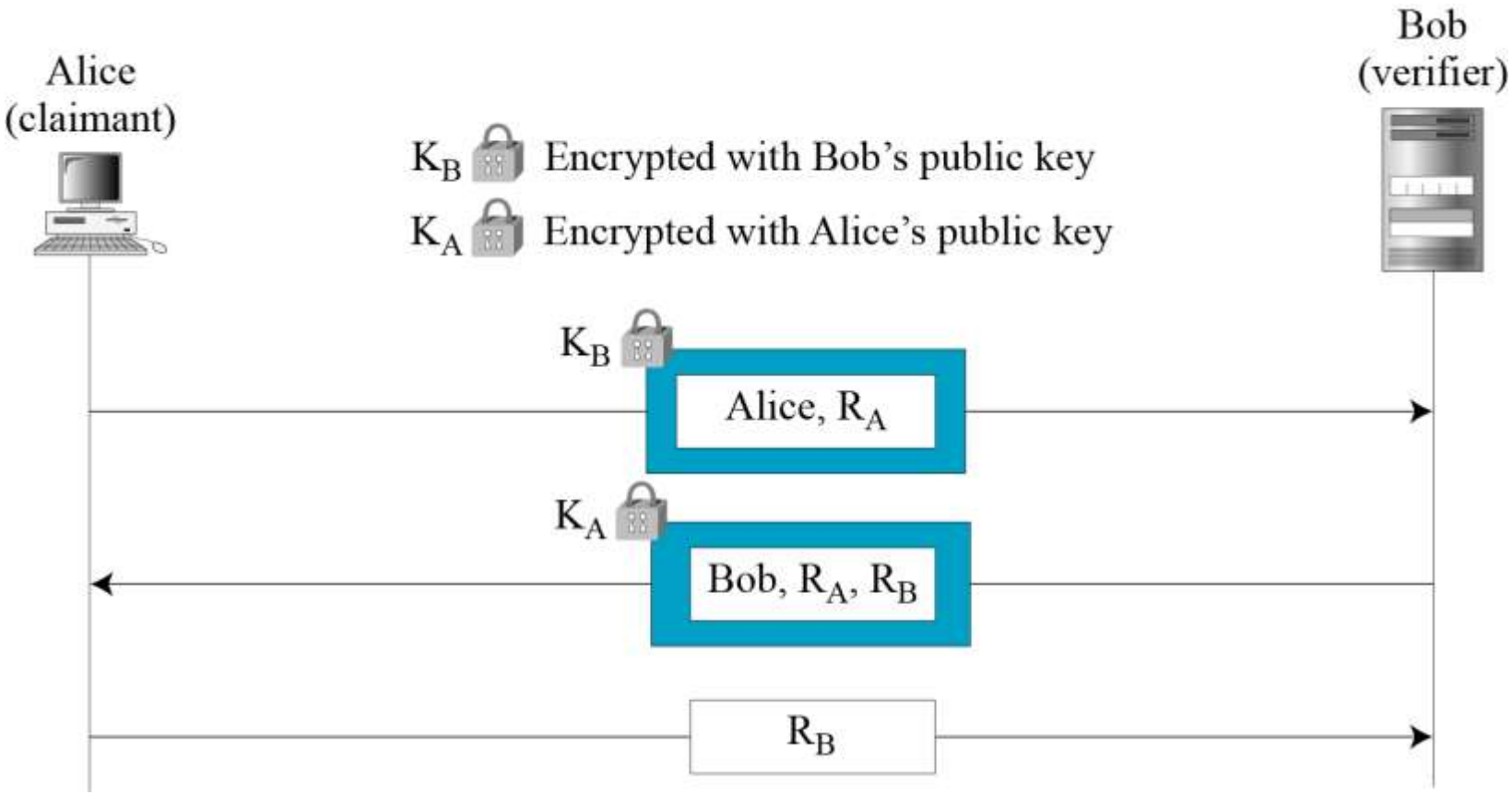
Figure 14.9 ***Unidirectional, asymmetric-key authentication***



- Ownership of claimant's private key

Second Approach

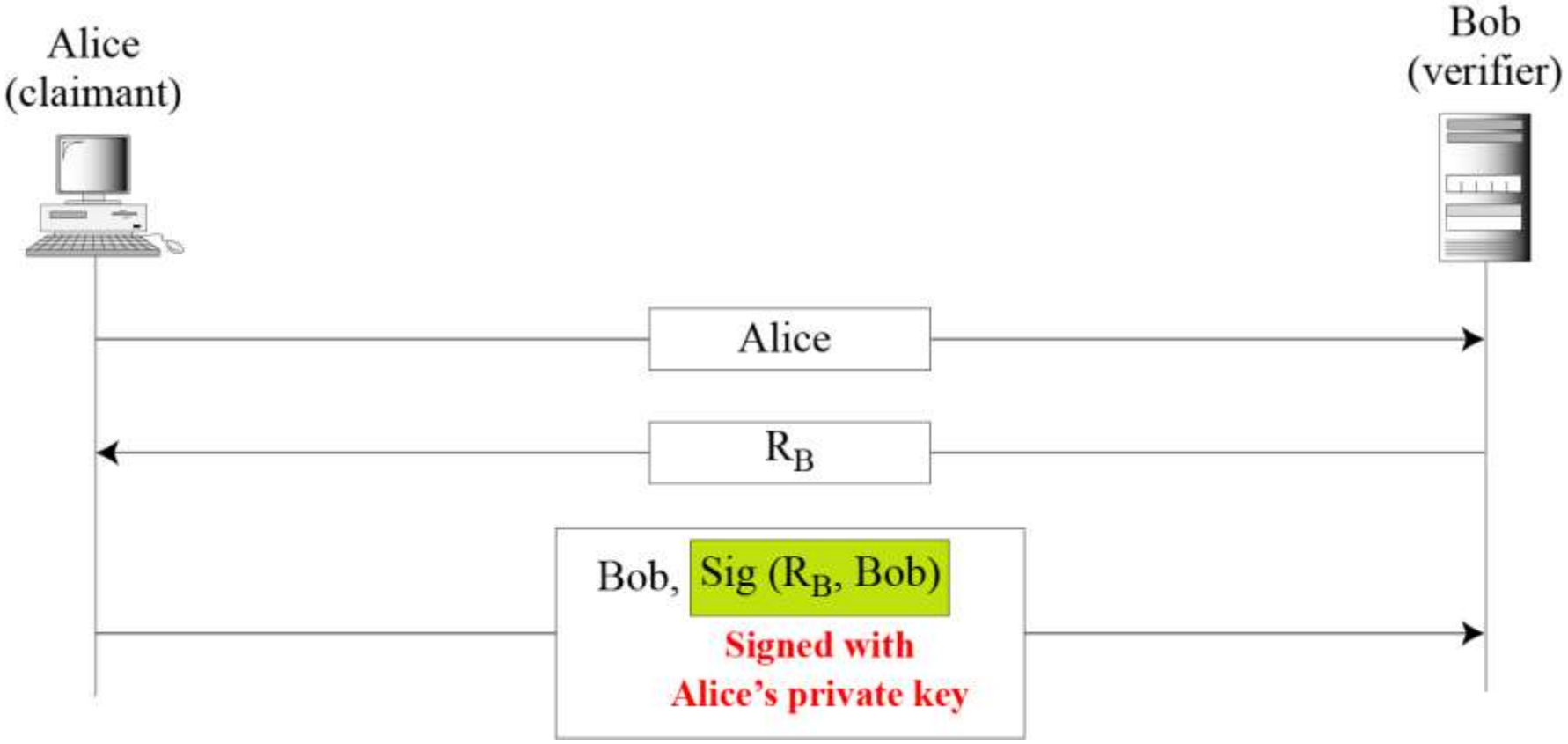
Figure 14.10 *Bidirectional, asymmetric-key*



14.3.4 Using Digital Signature

First Approach

Figure 14.11 *Digital signature, unidirectional*

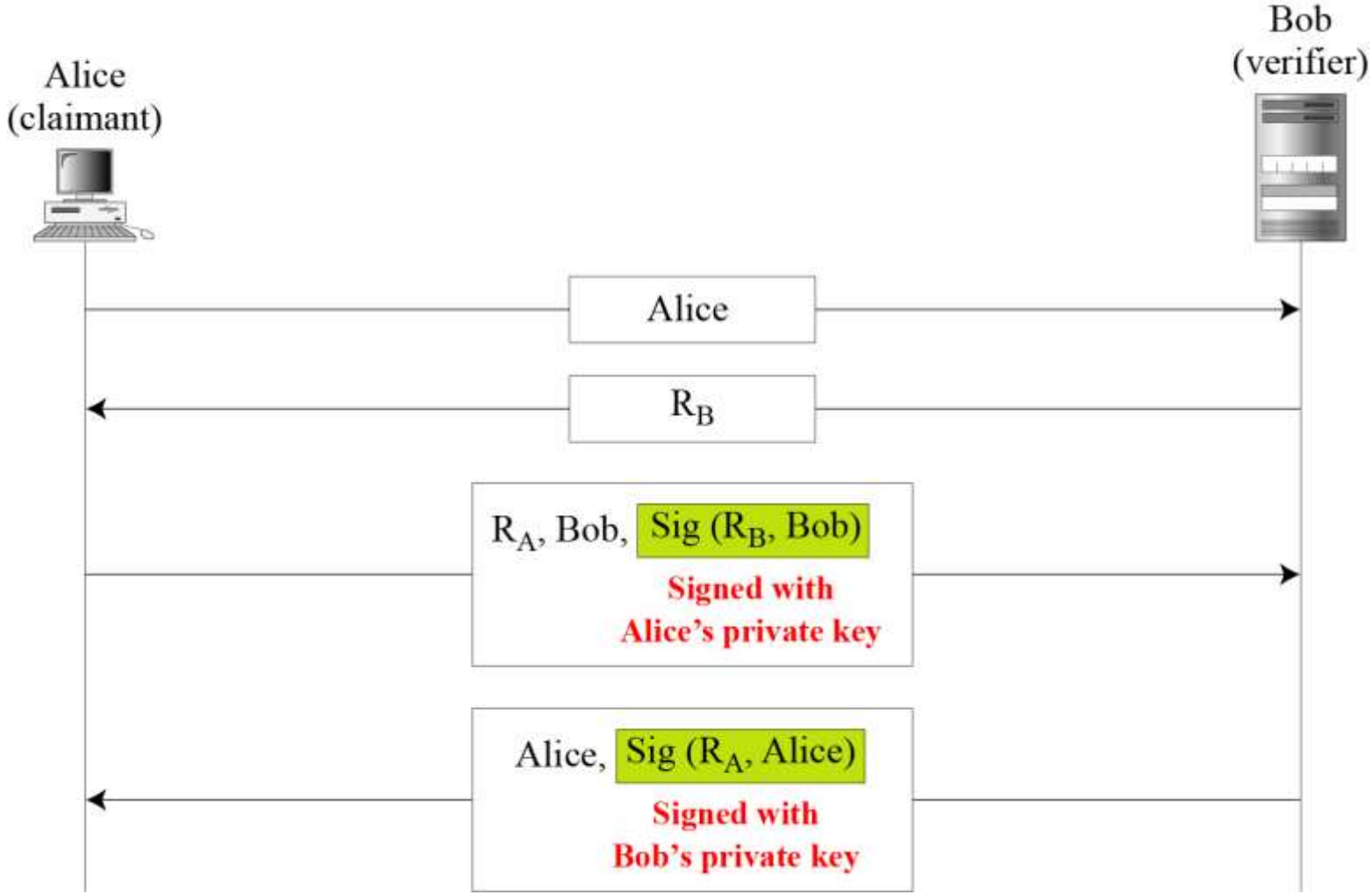


- Ownership of claimant's private key

14.3.4 Continued

Second Approach

Figure 14.12 *Digital signature, bidirectional authentication*



14-4 ZERO-KNOWLEDGE

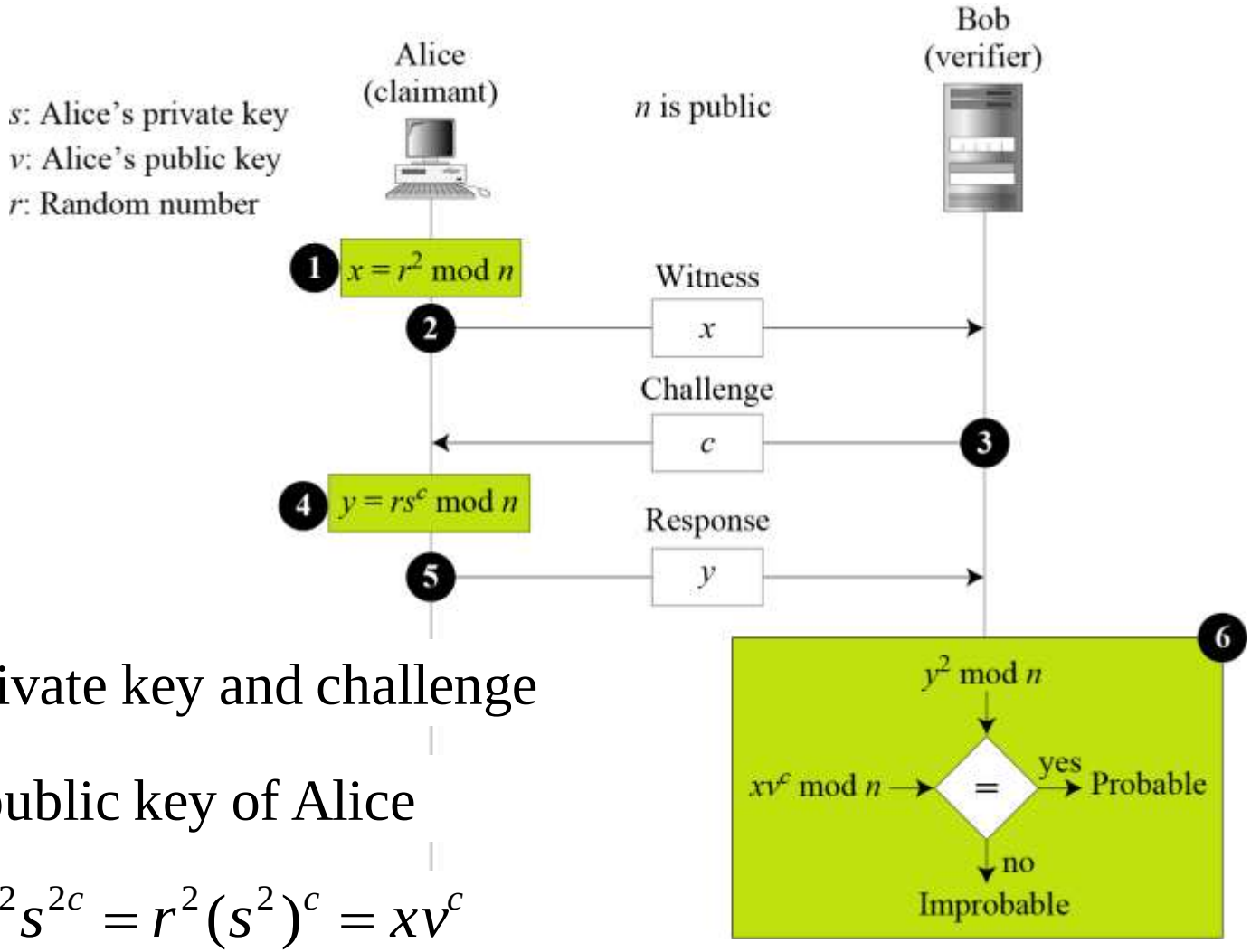
In zero-knowledge authentication, the claimant does not reveal anything that might endanger the confidentiality of the secret. The claimant proves to the verifier that she knows a secret, without revealing it. The interactions are so designed that they cannot lead to revealing or guessing the secret.

Topics discussed in this section:

14.4.1 Fiat-Shamir Protocol
14.4.2 Feige-Fiat-Shamir Protocol
14.4.3 Guillou-Quisquater Protocol

14.4.1 Fiat-Shamir Protocol

Figure 14.13 *Fiat-Shamir protocol*



y^2 is from private key and challenge

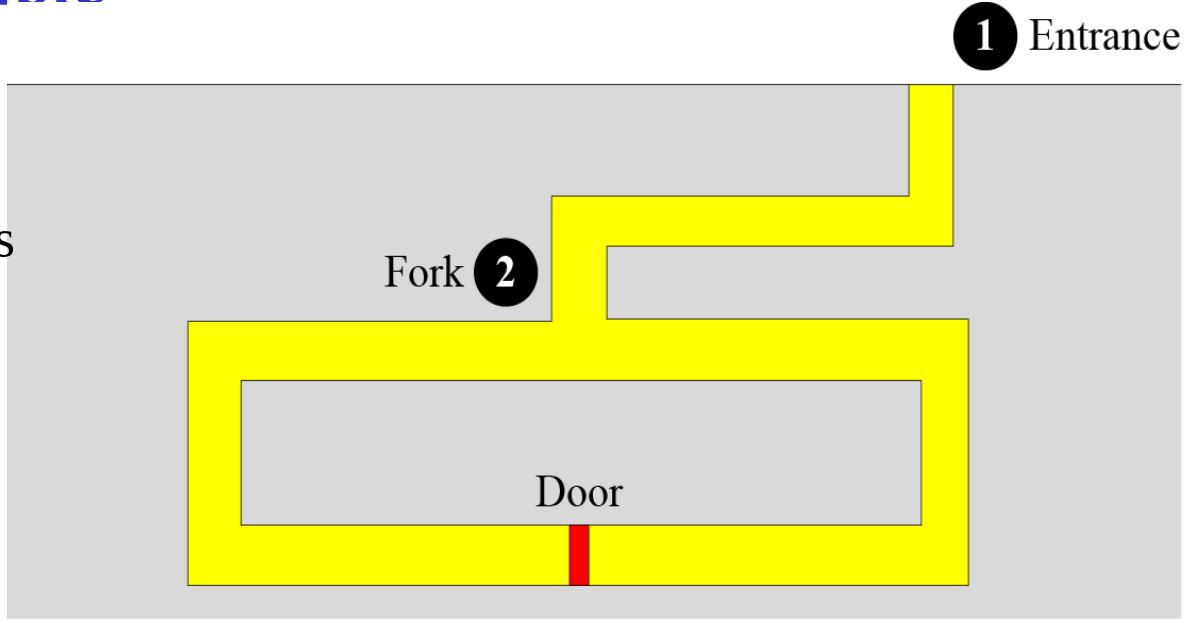
xv^c is from public key of Alice

$$y^2 = (rs^c)^2 = r^2 s^{2c} = r^2 (s^2)^c = xv^c$$

Cave Example

The door can only be opened with a magic word. Alice claims that she knows the word and that she can open the door. Bob and Alice are at pint 1. Alice enters eh case and reaches the point 2.

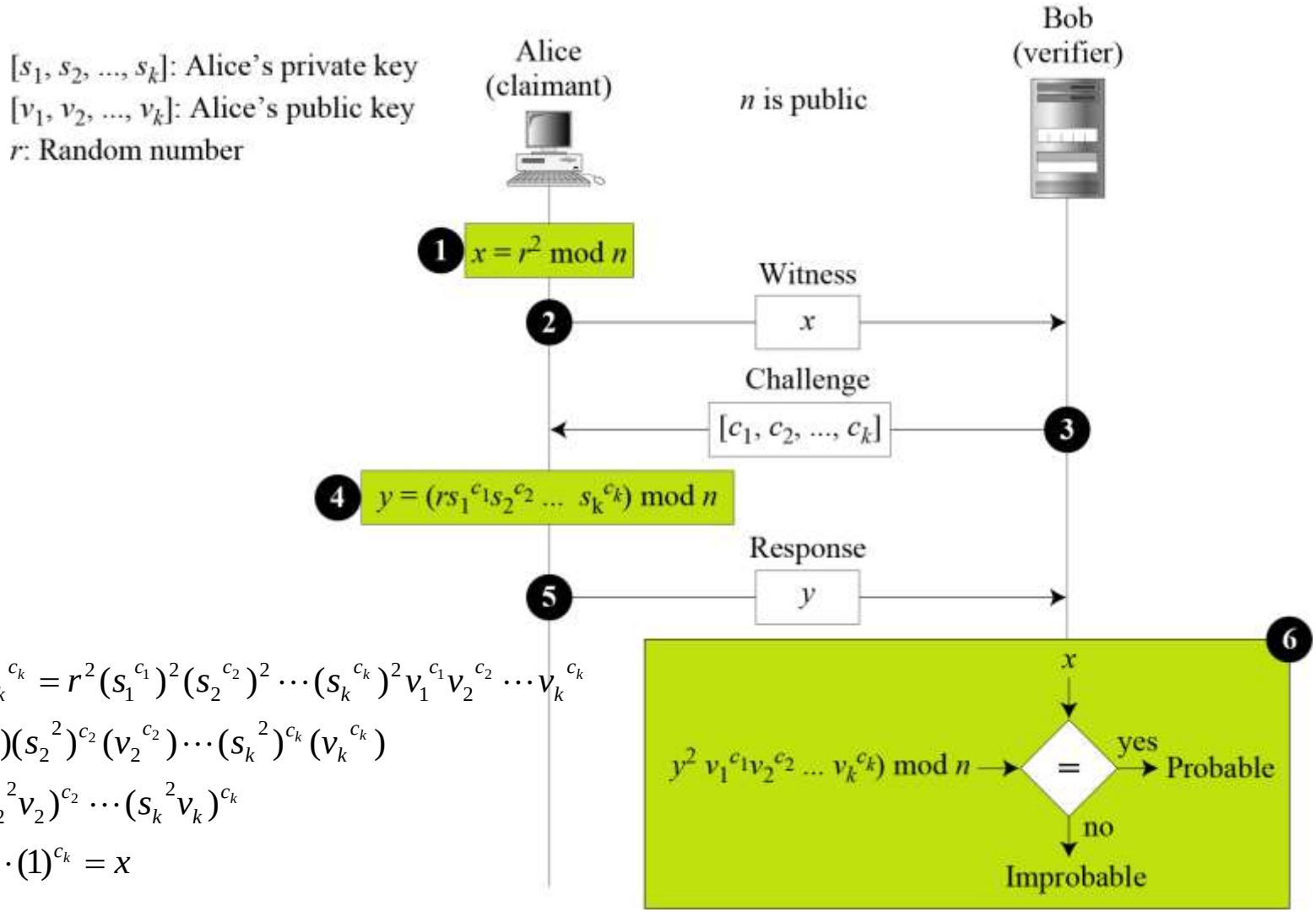
Figure 14.14 *Cave example*



1. Alice chooses to go either right or left. After Alice disappears, Bob comes to point 2 and asks Alice to come up from either the right or left.
2. if Alice knows the magic word, she will come up from the right direction. If she does not know the word, she comes up from the right direction with ½ probability.
3. The game will be repeated many times.

14.4.2 Feige-Fiat-Shamir Protocol

Figure 14.15 *Feige-Fiat-Shamir protocol*



$$y^2 v_1^{c_1} v_2^{c_2} \cdots v_k^{c_k} = r^2 (s_1^{c_1})^2 (s_2^{c_2})^2 \cdots (s_k^{c_k})^2 v_1^{c_1} v_2^{c_2} \cdots v_k^{c_k}$$

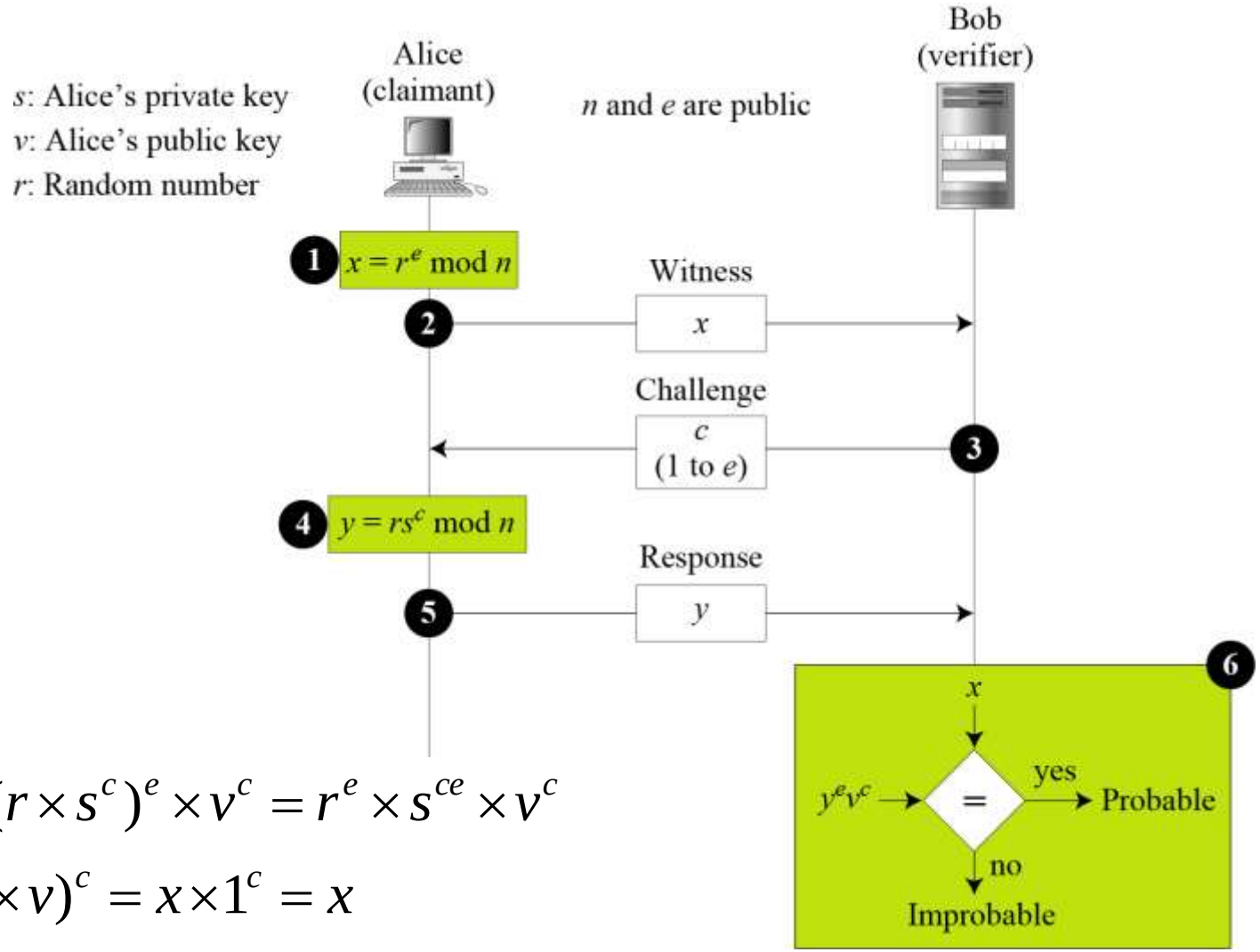
$$= x(s_1^2)^{c_1}(v_1^{c_1})(s_2^2)^{c_2}(v_2^{c_2}) \cdots (s_k^2)^{c_k}(v_k^{c_k})$$

$$= x(s_1^2 v_1)^{c_1}(s_2^2 v_2)^{c_2} \cdots (s_k^2 v_k)^{c_k}$$

$$= x(1)^{c_1}(1)^{c_2} \cdots (1)^{c_k} = x$$

14.4.3 Guillou-Quisquater Protocol

Figure 14.16 *Guillou-Quisquater protocol*



$$y^e \times v^c = (r \times s^c)^e \times v^c = r^e \times s^{ce} \times v^c$$

$$= r^e \times (s^e \times v)^c = x \times 1^c = x$$

14-5 BIOMETRICS

Biometrics is the measurement of physiological or behavioral features that identify a person (authentication by something inherent). Biometrics measures features that cannot be guessed, stolen, or shared.

Topics discussed in this section:

14.5.1 Components

Several components are needed for biometrics, including capturing devices, processors, and storage devices..

14.5.2 Enrollment

Before using any biometric techniques for authentication, the corresponding feature of each person in the community should be available in the database. This is referred to as enrollment.

14.5.3 Authentication

Verification

Identification

14.5.4 Techniques

Figure 14.17 *Techniques*