User Authentication with Public Key Technology

A Sevan White Paper

How are users identified and authenticated over the Internet? In the scenario where Bob and Alice are attempting to do business over the Internet, how does Alice know that the "other guy" is Bob?

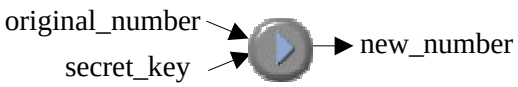
The conventional solution is with user names (for identification) and passwords (for authenticating the identification). The password is based on the assumption that the user and only the user knows his or her password. Therefore, if the user purporting to be Bob actually provides Alice with Bob's password, then Alice assumes that the user is indeed Bob.

Notice that the password scheme requires that Alice knows Bob's password. Therefore the user's password is not an exclusive secret of the user. For this reason, the password scheme is known as a "shared secret" system, where both the user and the user's partner know the password: the shared secret.

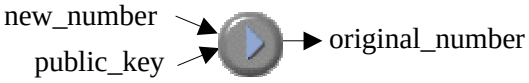
Public Key Cryptography

Public key authentication is based on public key cryptography and does not use a shared secret. Rather, each user possesses a pair of authentication *keys*. The user's *secret key* is a big number (typically a few hundred digits) and is available only to the user (it is the user's secret information). The user's *public key* is mathematically related to the secret key and it is known by anyone wishing to authenticate the user (it is the user's public information).

As mentioned previously, the secret and public keys are mathematically related to one another in such a manner that the following is true. If you apply any number and your secret key to the public-key mathematical function, you get another number.



If you then apply the new*_number* to the same function with your public key, you get the *original number*:



In other words, your public key "undoes" the mathematical operation caused by your secret key.

There are two other mathematical properties that we need to discuss before applying this to user authentication.

First, the mathematical relationship between a user's secret and public keys makes it practically impossible for someone knowing a user's public key to compute the user's secret key. This means that your secret key indeed remains a secret to you.

Second, the mathematics makes it practically impossible for someone knowing the *other_number* and *original_number* in equation 1 to compute the secret key. This again is necessary to protect the value of your secret key from all other parties.

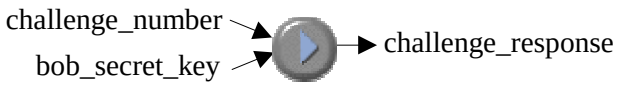
Now, to Authentication...

As before, we desire a method for Alice to authenticate Bob. If Bob has a pair of authentication keys:

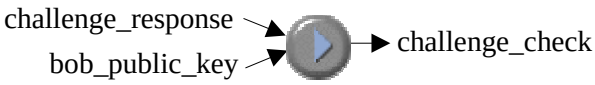
bob_**secret**_key & bob_**public**_key,

Alice can authenticate Bob using public-key cryptography through the following steps:

- Alice must first obtain the value of Bob's **public** key. Note that Alice does not have and will never have the value of Bob's **secret** key.
- When someone claiming to be Bob appears at Alice's web site, Alice does the following:
- Alice sends the alleged Bob (this person might not be Bob) a challenge number. This is just a long, random number.
- Bob receives the challenge number and computes a challenge response by:
- Bob sends the challenge response to Alice



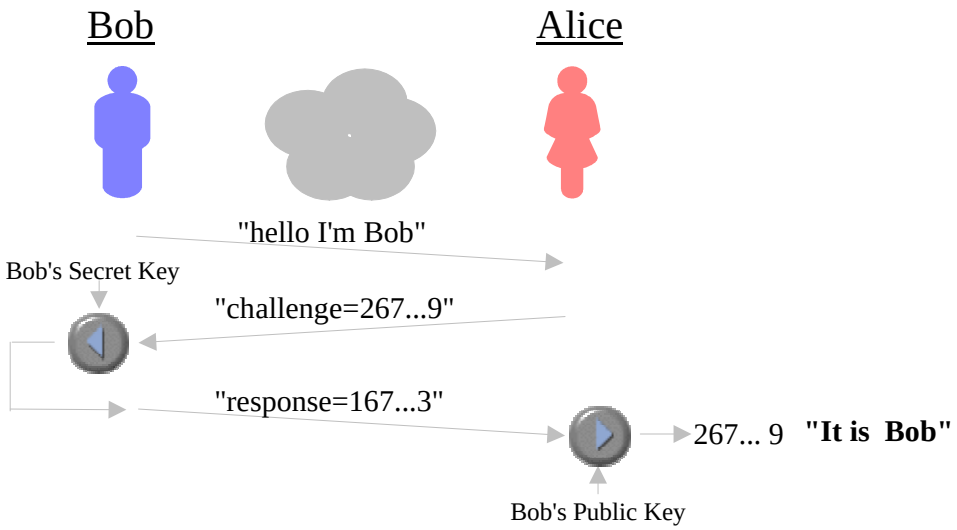
- Alice checks Bob's challenge response by:



- If the *challenge_check* computed by Alice is the same as the *challenge_number* that Alice had sent to Bob, then Alice knows that the person purporting to be Bob has access to Bob's secret key.

On this basis Alice authenticates the user as Bob.

This public key authentication process is illustrated in the figure below.



What About Passwords?

Unlike password schemes, public-key cryptography does not require Bob to share his secret with others. Therefore, Bob's secret remains safe. Since Bob's partners do not know his secret_number, Bob can use the same pair of authentication numbers for all of his partners. Using the same password with different partners is risky, since each partner knows Bob's password and could therefore masquerade as Bob.

Public Key Cryptography and Web Browsers

These public key operations are a standard feature in every modern web browser. The browsers use public key cryptography to generate Secure Socket Layer, SSL, connections. These SSL connections allow the browser to do the following:

- authenticate web servers,
- generate keying material for encrypting the information flowing between the browser and the server, and
- optionally allow the server to authenticate the browsers.

The interesting point is that the browsers perform these complex functions transparently to the users. Even the most inexperienced user can browse the web securely without even noticing the public key mathematics being executed by the browsers.

sevan networks

Sevan Networks, Inc
1310 Hollenbeck Ave, Ste F
Sunnyvale, CA 94087
Tel: 408.830.1000
Fax: 408.830.1001
www.sevannetworks.com