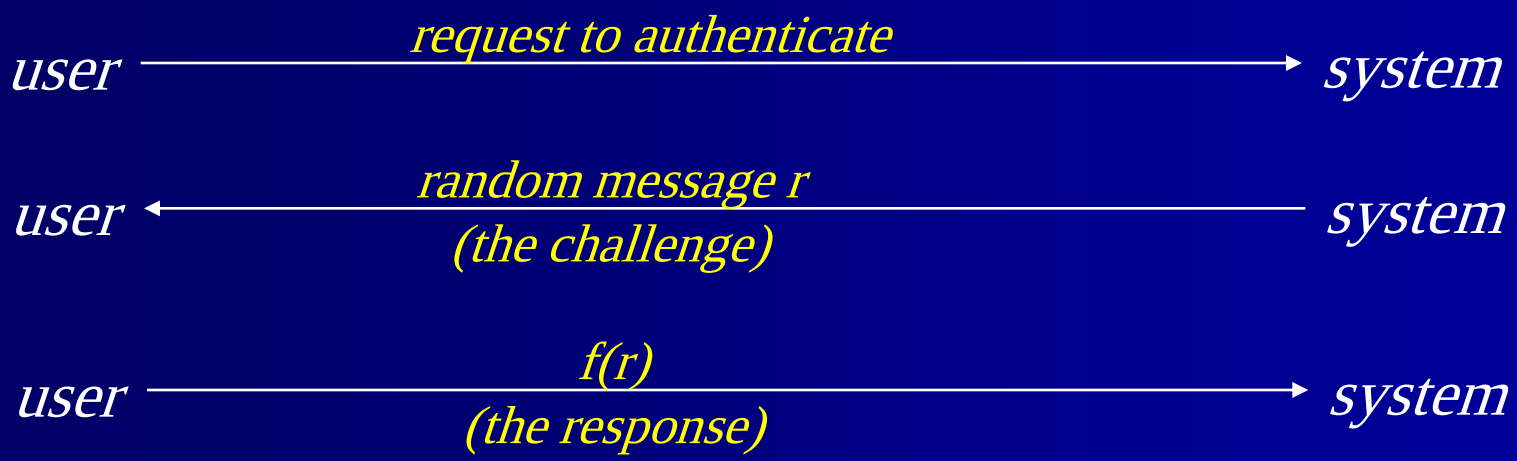
2.3.3. Challenge-Response
(Strong authentication)

- Another alternative is to authenticate in such a way that the transmitted password changes each time
- Let a user *u* wishing to authenticate himself to a system *S*. Let *u* and *S* have an agreed-on secret function *f*.
  - A ***challenge-response*** authentication system is one in which *S* sends a random message *m* (the *challenge*) to *u*, and *u* replies with the transformation $r = f(m)$ (the *response*). *S* then validates *r* by computing it separately.
- The *challenge* may be a ***nonce***, ***timestamp***, ***sequence number***, or any combination.

Challenge-Response

(by symmetric-key techniques)

- The user and system share a secret function f (in practice, f can be a known function with unknown parameters, such as a cryptographic key).
- This called challenge-response by symmetric-key techniques.



Challenge-Response
(by public-key techniques)

- A identifies B by checking whether B holds the secret (private) key KR_B that matches the public key KU_B
- A chooses a random challenge (nonce) r_A. B uses its random nonce r_B. B applies its public-key system for authentication
- Message sequence:

  1. $A \rightarrow B$: r_A.
  2. $B \rightarrow A$: r_B, $E_{KR_B}(r_A, r_B)$

3. Cryptographic Key Infrastructure

- Goal: bind identity to key
- Symmetric Cryptography:
  - Not possible as all keys are shared
- Public key Cryptography:
  - Bind identity to public key
  - Crucial as people will use key to communicate with principal whose identity is bound to key
  - Erroneous binding means no secrecy between principals
  - Assume principal identified by an acceptable name

Certificates

- A certificate is a token (message) containing
  - Identity of principal (e.g., Alice)
  - Corresponding public key
  - Timestamp (when issued)
  - Other information (perhaps identity of signer)
  - Signature of a trusted authority (e.g., Cathy)

$$C_A = D_{kv}(K_{u\_a} \,||\, \text{Alice} \,||\, T)$$

Kv Cathy's private key

C_A is A's certificate

Certificate Use

- Bob gets Alice's certificate
  - If he knows Cathy's public key, he can validate the certificate
    - When was certificate issued?
    - Is the principal Alice?
  - Now Bob has Alice's public key

- Problem:
  - Bob needs Cathy's public key to validate *Alice*'s certificate
  - Many solutions:
    - Public Key Infrastructure (PKI),
    - Trust-based certificates (PGP)

X.509 certificate

- Key certificate fields in X.509v3:
  - Version
  - Serial number (unique)
  - Signature algorithm identifier: hash algorithm
  - Issuer's name; uniquely identifies issuer
  - Interval of validity
  - Subject's name; uniquely identifies subject
  - Subject's public key
  - Signature:
    - Identifies algorithm used to sign the certificate
    - Signature (enciphered hash)
- Issuer is called ***Certificate Authority*** (CA)

PKI

- ***A Public Key Infrastructure*** (PKI) is a set of services and policies that lays the framework for binding a public key to an identity and distributing that binding
  - Or in other words, a PKI is a system that provides authentic public keys to applications
- A PKI has three basic processes: Certification, Validation, and Certificate revocation.

1. Certification
   - Binding identities (or attributes) to a public key by a trusted third party to form a certificate.
     - A trusted third part in a PKI is called Certificate Authority (CA),
     - A CA digitally signs a certificate means it vouches for the correctness of the certificate's contents.

PKI

2. Validation
   - The process of verifying the authenticity of the certificate
     - This means that the certificate's contents can be believed.
   - This involves verifying the signature of the CA by using the CA's public key, by checking the validity period contained in the certificate itself, and by checking the certificate against a CRL
   - A CRL (Certificate Revocation List) contains certificate that have been revoked (ie, not valid) by the CA

3. Certificate revocation
   - The process of disavowing a previously issued certificate before its expiration date. This may happen if some of the information contained in the certificate changes.

4. Conclusion

- Entity Authentication
  - Something you have: rarely used alone
  - Something you are: Biometrics
  - Something you know (passwords and secrets)

- Cryptographic Key Infrastructure:
  - authentic Public key
    - entity authentication (based on Challenge-response mechanism)


Questions?