User Authentication Protocols

Information Security – Lecture 09

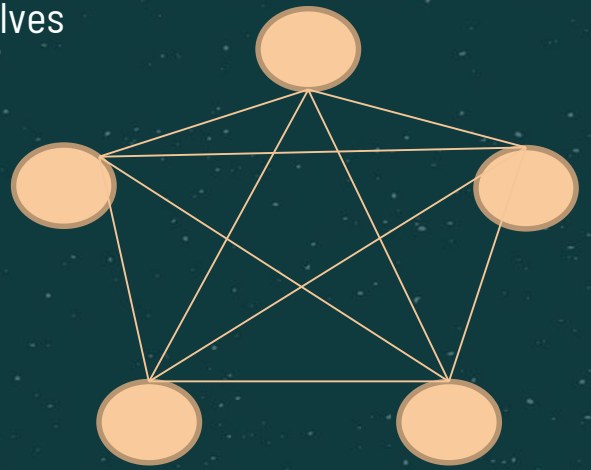
Aadil Zia Khan

Threats in a Networked Environment

- User gains access to a workstation, and pretends to be someone else
- User alters the network address of a workstation to impersonate another workstation
- User intercepts and uses a replay to gain unauthorized access

Why The Need For A User Authentication System

- Enterprise networks may have hundreds of clients and servers
- How can we ensure secure communication between them?
  - Each possible pair of nodes share a secret key between themselves but don't disclose it to others
- How can we ensure access control between them?
  - Each node maintains information regarding who can access it
- Problem with this approach
  - Too much state maintenance at each node
  - Total number of shared keys would be $O(n^2)$
    - Actual figure will be $n(n-1)/2$



Solution: Keep a central server which would be responsible for authentication

User Authentication System Using Private Key

Kerberos

Kerberos: three headed dog in Greek mythology, the guardian of the entrance of Hades

- Trusted key server system from MIT
- Centralized authentication server manages authentication between users and application servers/workstations
- It is symmetric
- Provides centralised private-key third-party authentication in a distributed network
  - Allows users access to services distributed through network without needing to trust all workstations – just trust a central authentication server
  - Much more efficient

Kerberos: Requirements

reliability and scalability imply a distributed architecture

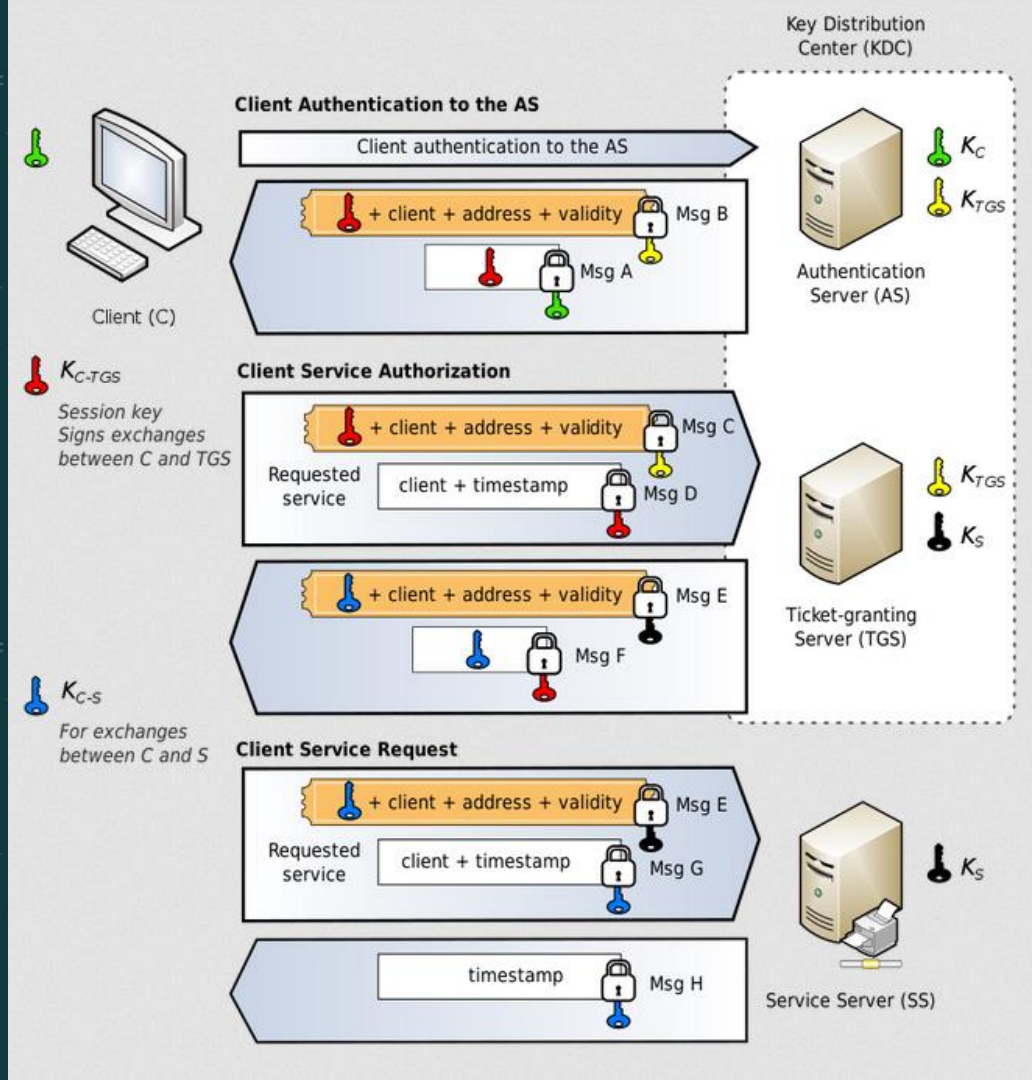
- Security
  - Opponents should not be able to gain access
- Reliability (availability)
  - Kerberos server should be available all the time
- Scalability
  - System should be able to support large amount of users
- Transparency
  - Users should see the system as a username/password system
    - And not be tied down to underlying implementation detail

Kerberos: Working

Note: User enters a username and password on the client machine - the client transforms the password into the key of a symmetric cipher using hash function. Authentication Server already has passwords.

Note: Authentication Server maintains a list of access rights of all nodes

Note: Session keys expire after a short while



Kerberos: Usage

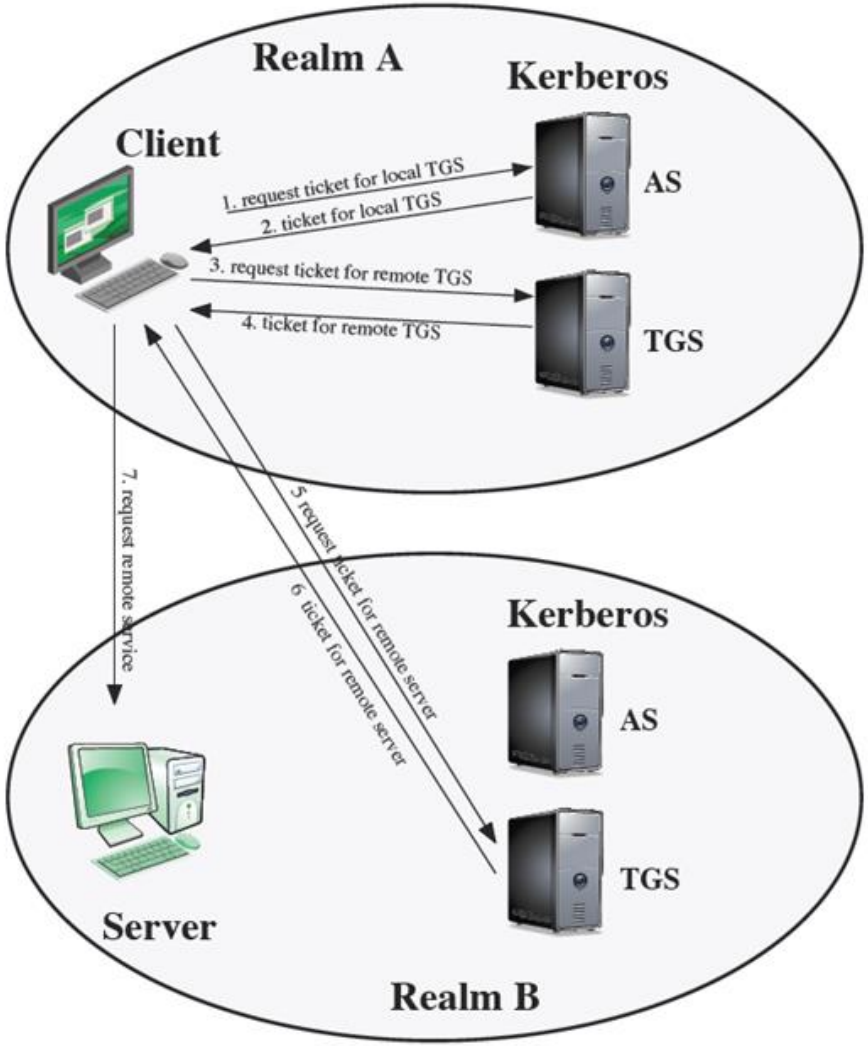
- Windows
- Unix-like operating systems, including FreeBSD, OpenBSD, Apple's macOS, Red Hat Enterprise Linux, Oracle's Solaris
- Non-Unix like operating systems such as OpenVMS
- Embedded implementation is also available

Kerberos: Limitations

- Kerberos has strict time requirements, which means the clocks of the involved hosts must be synchronized within configured limits
- Each network service which requires a different host name will need its own set of Kerberos keys - this complicates virtual hosting and clusters.
- Kerberos requires user accounts and services to have a trusted relationship to the Kerberos token server

Kerberos: Realms

- Kerberos environment consists of:
  - Kerberos server
  - Multiple clients, all registered with server
  - Application servers, sharing keys with server
- This is called a Kerberos realm
  - Usually a single administrative domain
- Inter-realm authentication possible
  - Requires mutual trust



User Authentication System Using Public Key

Problem With Public Entities

- If you visit a webpage – how will you determine its authenticity?
  - You will use its public key to decrypt some data
- How will you know what its public key is?
  - You will need to get the public key from a trusted source

X.509

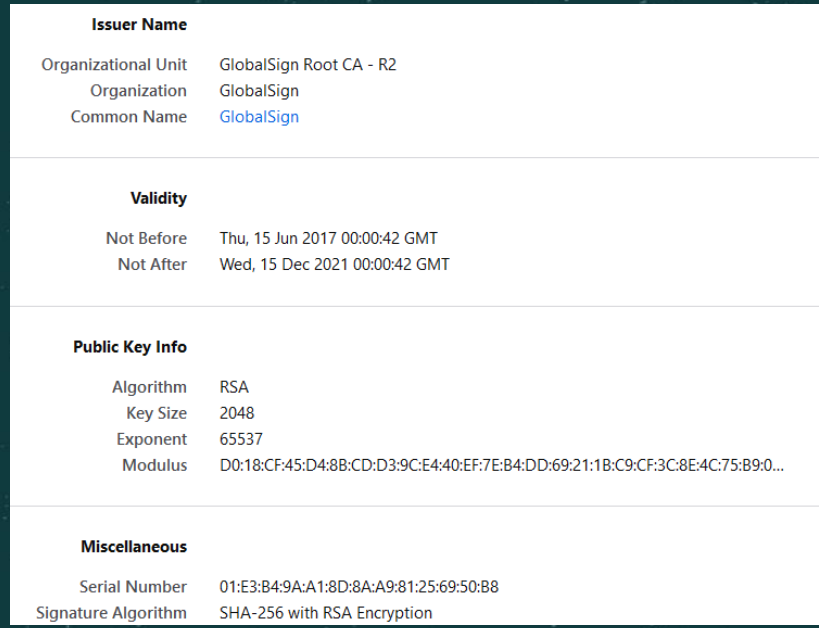
- X.509 is a standard format for public key certificates
  - Public key certificates: digital documents that securely associate cryptographic key pairs with identities such as websites, individuals, or organizations
- A distributed set of servers maintain a database about users
- Certificate contains public key of a user - signed with private key of a Certification Authority
- Used in S/MIME, IP Security, SSL/TLS and SET

X.509: Players

- Certification Authority
- Entity that needs to share its public key – e.g., webpage
- Entity that needs to obtain another entity's public key

X.509: Certificates

- Issued by a Certification Authority (CA)
- Certificates contain
  - version (1, 2, or 3)
  - serial number (unique within CA) identifying certificate
  - signature algorithm identifier
  - issuer name (CA)
  - period of validity (from - to dates)
  - subject name (name of owner)
  - subject public-key info (algorithm, parameters, key)
  - issuer unique identifier
  - subject unique identifier
  - extension fields
  - signature (of hash of all fields in certificate)



Issuer Name

Organizational Unit	GlobalSign Root CA - R2
Organization	GlobalSign
Common Name	GlobalSign

Validity

Not Before	Thu, 15 Jun 2017 00:00:42 GMT
Not After	Wed, 15 Dec 2021 00:00:42 GMT

Public Key Info

Algorithm	RSA
Key Size	2048
Exponent	65537
Modulus	D0:18:CF:45:D4:8B:CD:D3:9C:E4:40:EF:7E:B4:DD:69:21:1B:C9:CF:3C:8E:4C:75:B9:0...

Miscellaneous

Serial Number	01:E3:B4:9A:A1:8D:8A:A9:81:25:69:50:B8
Signature Algorithm	SHA-256 with RSA Encryption

- Registration Authority verifies the identity of entities

X.509: Obtaining User Certificate

- Properties of certificates generated by CA
  - They are signed using private key of CA
  - User with access to the public key of the CA can recover the entity's certified public key
  - Nobody other than the CA can modify the certificate without this being detected
- Once the public key is obtained, it can be used for encryption and authentiation

X.509: CA Hierarchy

- If both parties use the same CA, they know its public key and can verify each others certificates
- If not, then there has to be some means to form a chain of certifications between the CA's used by the two parties, by the use of client and parent certificates
    - It is assumed that each client trusts its parents certificates

X.509: Certificate Revocation

- A certificate may need to be revoked for the following reasons
  - The user's private key is assumed to be compromised
  - The user is no longer certified by this CA
  - The CA's certificate is assumed to be compromised
- The CA maintains a revocation list – which is also digitally signed
- Note that revocation is different from expiry – an expired certificate may not be a security threat but a revoked certificate will be

Public Key Infrastructure Based On X.509

PKI - the set of hardware, software, people, policies, and procedures needed to create, manage, store, distribute, and revoke digital certificates based on asymmetric cryptography