Aalto University

See first the CS-C3130 lecture on OS access control

# Network Security: Kerberos in Windows domains

Tuomas Aura

CS-E4300 Network security
Aalto University

Thanks to Dieter Gollmann

# Windows access control summary

- The OS stores security attributes for each process (subject) in an access token
- Token contains a list of SIDs, i.e., user and group identifiers
  - Permissions are decided by comparing the list of SIDs against a DACLs on an object
- The access token is local to the machine, created at login time, and never sent over the network
- How to authorize access to resources managed by a Windows service on a remote server, e.g., over remote procedure call (RPC)?

# Network credentials

- User's username, SID and network credentials are cached on the user workstation
  - Network credetials: username and password, or TGT and $K_{A\text{-}TGS}$
- User's processes can use the network credentials for remote login
  - Two authentication protocols: NTLM and Kerberos V5
  - Neither reveals the password to the server
- Applications can also ask the user for a different username and credentials and store them separately

# Tokens and remote access

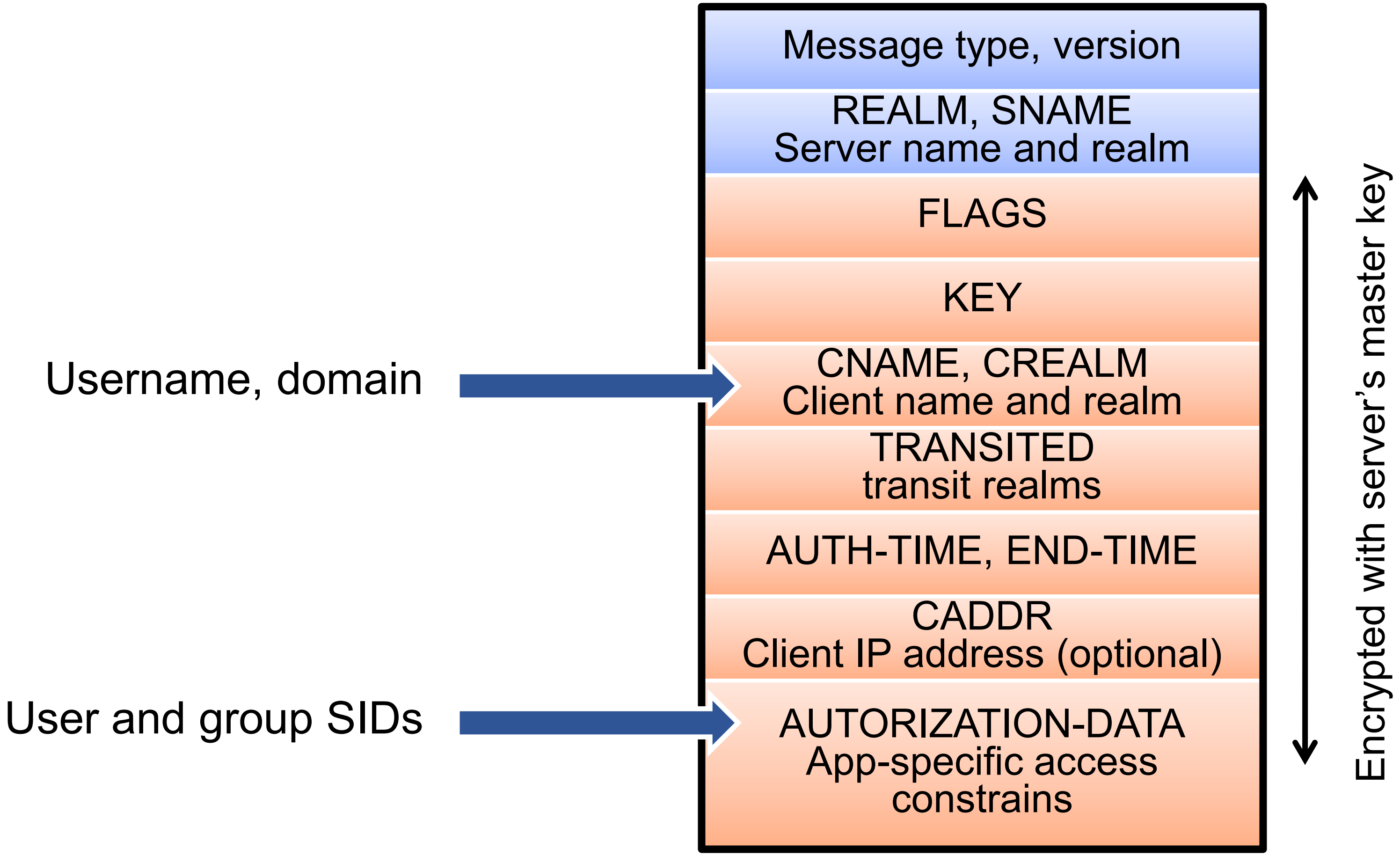
- The client authenticates the user to the remote server with the user's network credentials. The server creates a new login session and a new token (on the server) for Alice
- The service may assign the token to a process or thread (=impersonation)
- The authentication protocols also need to
  - provide the server with Alice's user and group SIDs
  - produce a session key for protecting data between the client and server
- Encryption and authentication of session data is up to the applications
  - Different secure session protocol exist for network logon, RPC, COM

# Kerberos in Windows

- Realm = Windows domain
- Realm hierarchy = domain hierarchy
- KDC = domain controller (DC)
  - Information about users is stored in active directory (AD)
- Kerberos authenticates “principals”, but which principals should be authenticated?
  - Domain and username, e.g., MYCOMPANY\Boss)? Appropriate fields in the ticket for this are CNAME and CREALM
  - However, Windows identifies the user by the user SID and group SIDs. Microsoft put these into the authorization-data field in the ticket

→ This created a major controversy in the early 2000s: incompatible use of a standard protocol. The result is that many IETF standards now require a formal process for any proprietary use of extension fields. The odified Kerberos tickets were later standardized.

# Kerberos ticket in Windows



Message type, version

REALM, SNAME
Server name and realm

FLAGS

KEY

CNAME, CREALM
Client name and realm

TRANSITED
transit realms

AUTH-TIME, END-TIME

CADDR
Client IP address (optional)

AUTORIZATION-DATA
App-specific access constrains

Username, domain

User and group SIDs

Encrypted with server’s master key

# Exercises

- Why are standards needed? For interoperability or something else?
- Should standard protocols include data fields or messages for proprietary extensions? What are the arguments for and against?