통합 인증 시스템 설계 및 구현

연세대학교 전기·전자공학과
정연식, 송홍엽

Contents

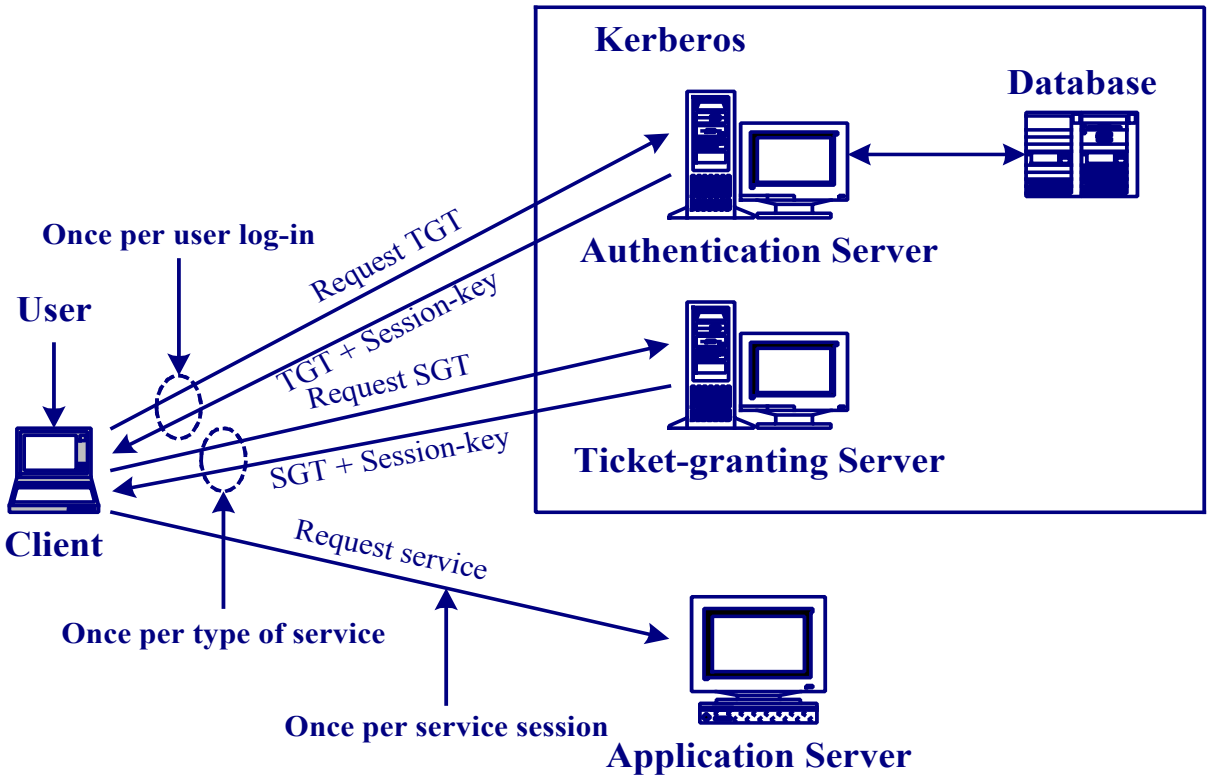
- Introduction
- Previous Works
- Design and Implementation of Public-Key Infrastructure
- Design and Implementation of Single Sign-On
- Conclusion

Introduction

- Multi-Server Environment
  - Problems of multiple passwords
  - Problems of implementation of challenge-response protocol
  - Single Sign-On
- Single Sign-On
  - Integration of authentication schemes inside domain of service
  - A user logs in once using a single password, and gets authenticated access to all servers in the Intranet
  - Without sending any passwords over the network

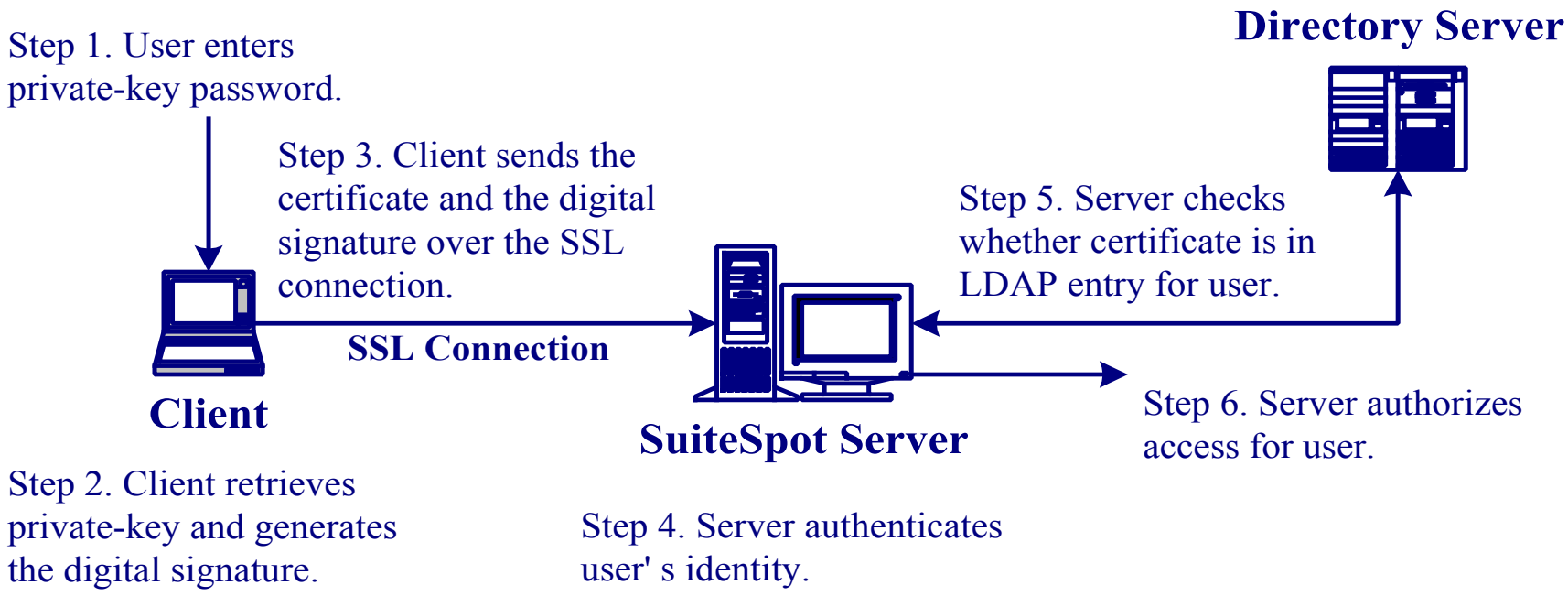
Previous Works: Kerberos

- Kerberos System
  - Based on secret-key cryptosystem
  - Based on Needham-Schroeder's third-party protocol



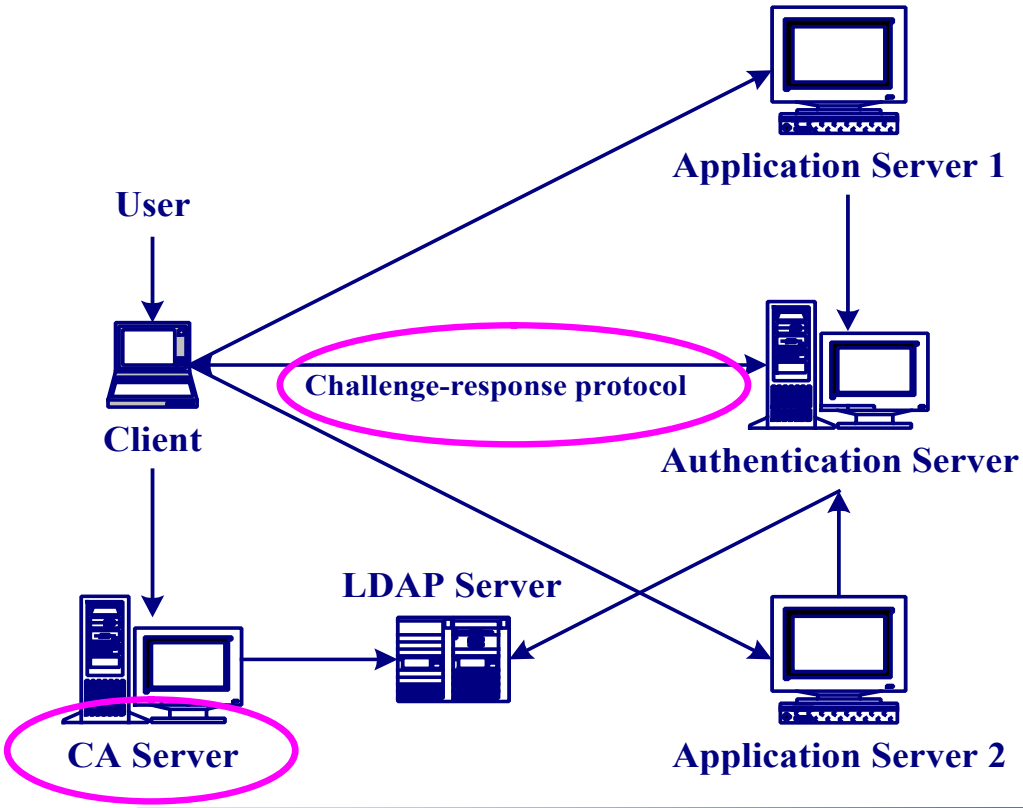
Previous Works: Netscape's SSO

- Netscape's SuiteSpot Server
  - Based on digital signature
  - Based on Secure Socket Layer(SSL)



The Proposed Single Sign-On

- The Proposed Single Sign-On
  - Based on Public-Key Infrastructure
  - Use of challenge-response protocol



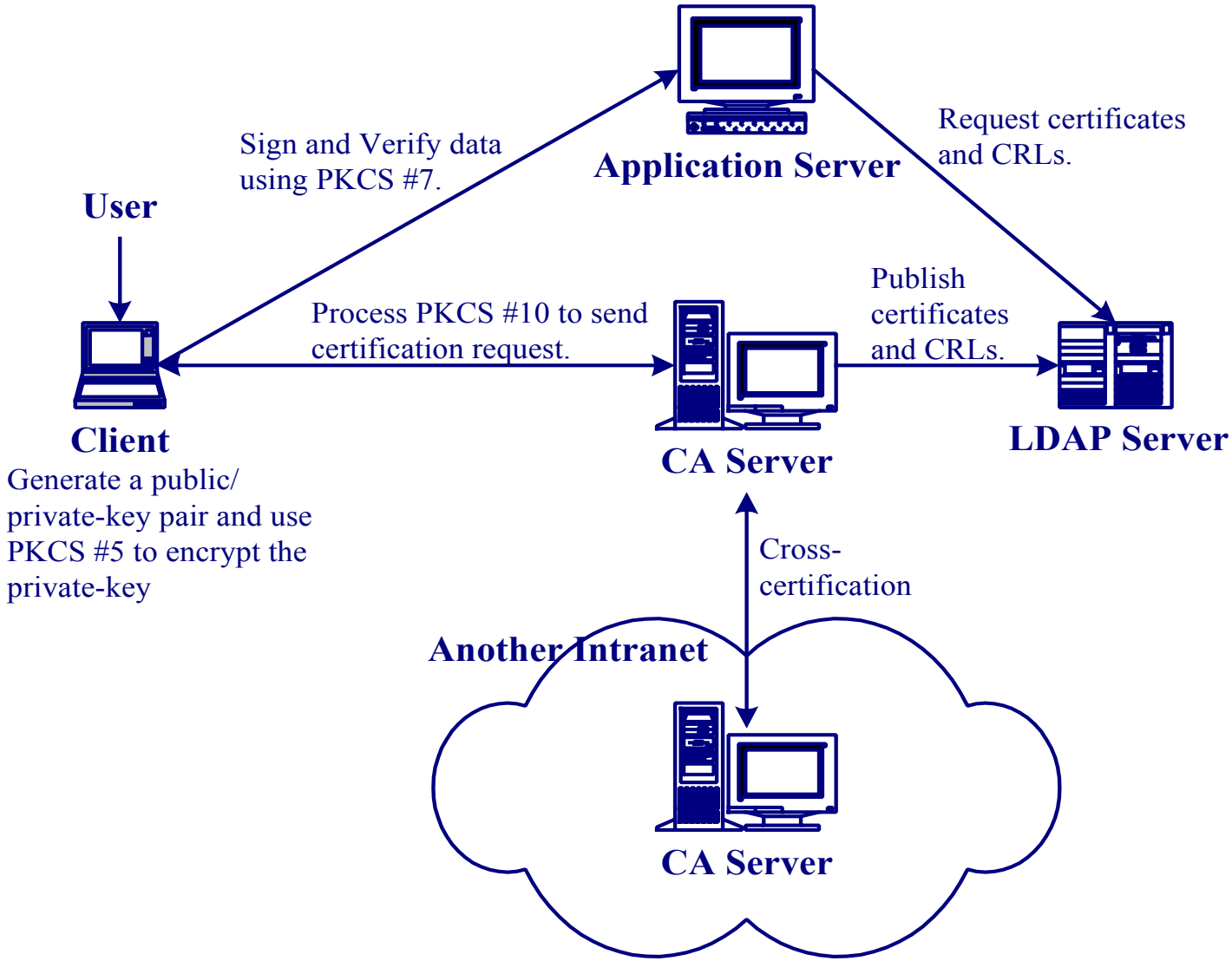
Contents

- Introduction
- Previous Works
- ✓ Design and Implementation of Public-Key Infrastructure
- Design and Implementation of Single Sign-On
- Conclusion

Public-Key Infrastructure

- Concept
  - Enables the use of public-key encryption and digital signature in a consistent manner
  - Certificate management infrastructure
    - Issues and provides access to public-key certificates
  - Network of certificate authorities
- Components
  - Certificate Authority(CA) server
  - Directory server
  - Client modules

Generation of Certificate

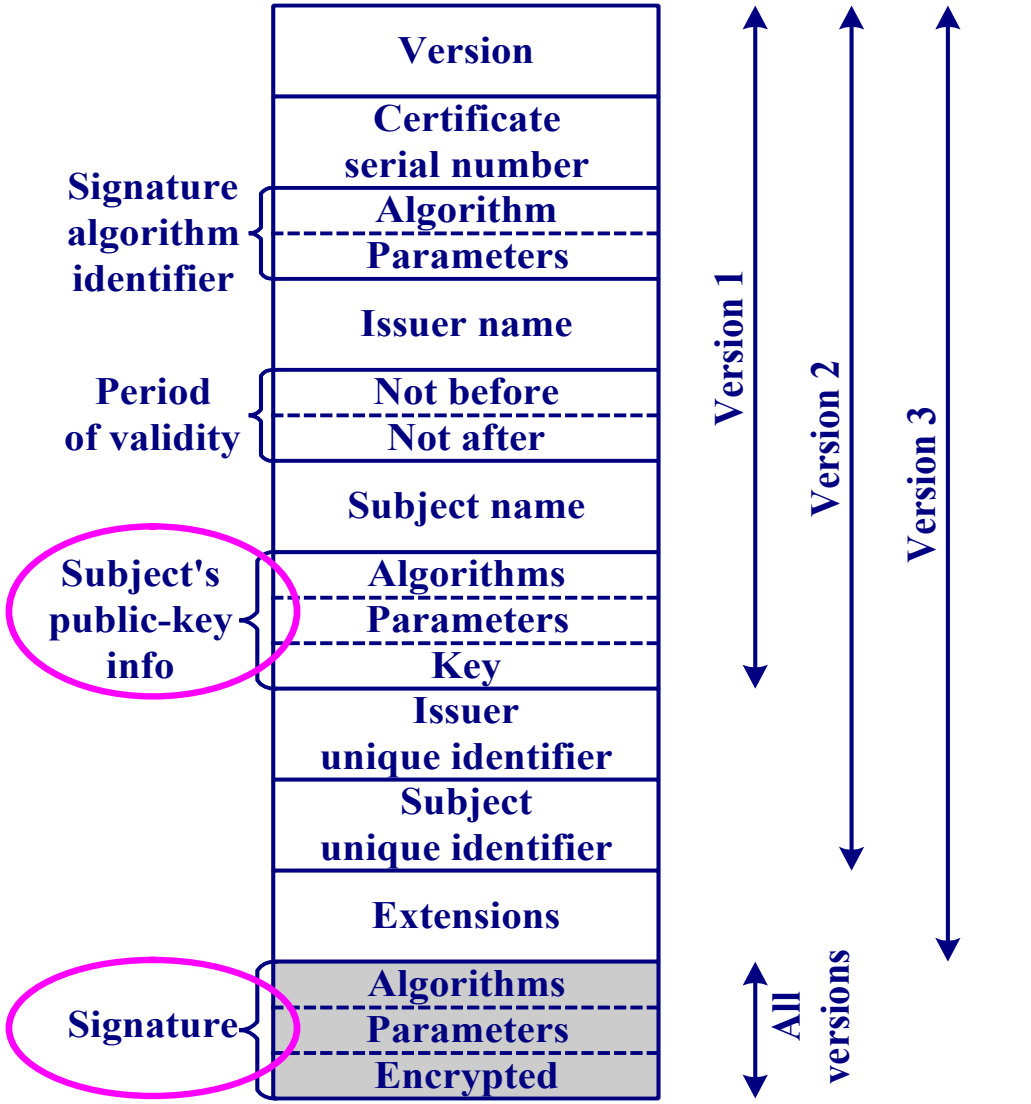


CA Server

- Functions
  - Generates certificates
  - Registers the certificates into the directory server
  - Issues certificates
    - Using PIN
  - Revokes certificates
    - Simply deletes a user entry and its attribute from the directory
    - Does not need the use of certificate revocation list(CRL)
- Implementation
  - Apache web server
  - CGI programming

Certificate

- X.509 Certificate Format
  - Version
    - ver.1
  - Signature algorihtm identifier
    - MD5 with RSA encryption
  - Issuer name
    - Root CA(Yonsei CA)
  - Period of validity: Not before
    - Present date
  - Period of validity: Not after
    - Present date + 1 year
  - Subject's public-key info
    - RSA algorithm



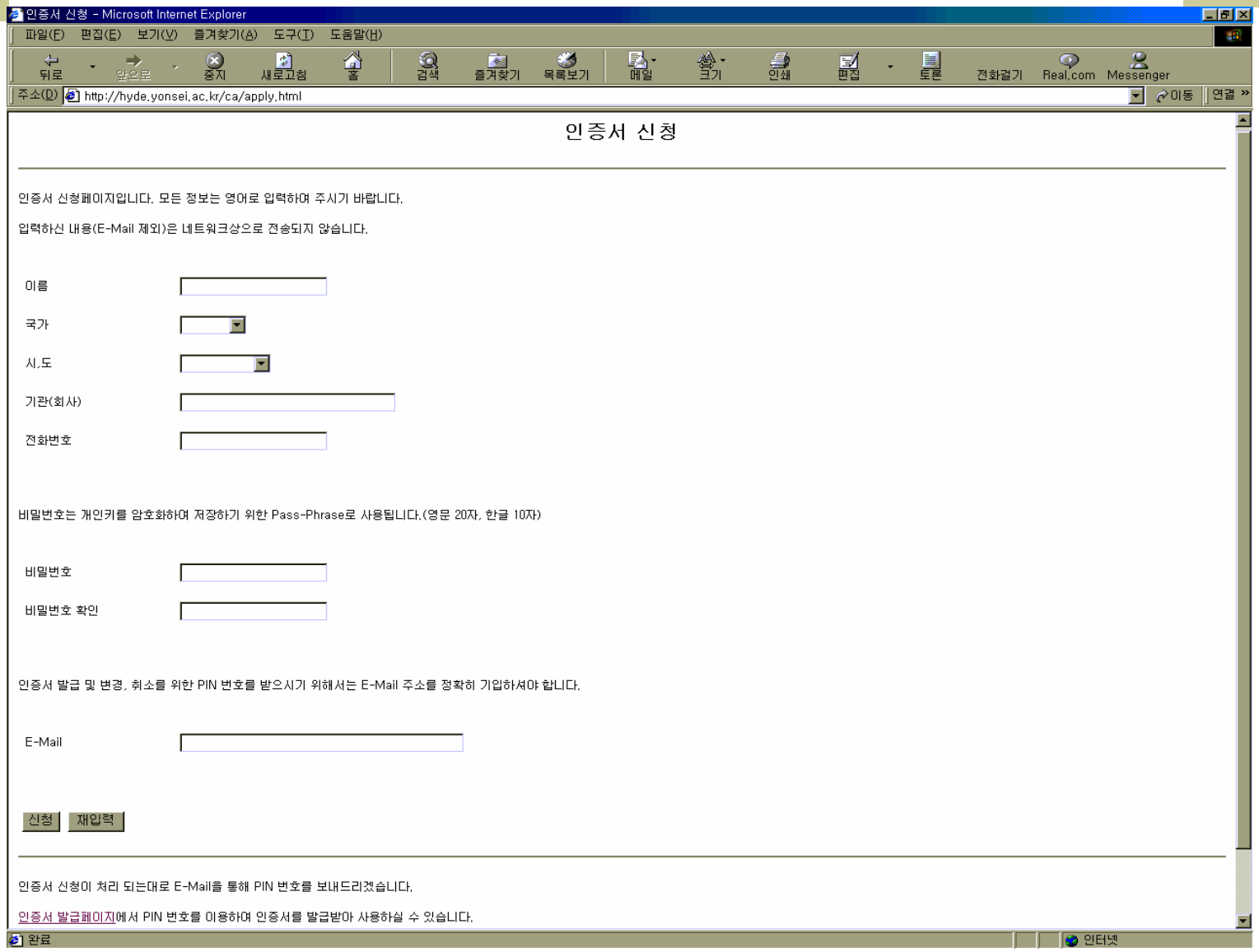
Directory Server

- Functions
  - Stores the certificates and CRLs
  - Makes all stored certificates available on request
- Lightweight Directory Access Protocol (LDAP)
  - Simplified DAP(X.500)
  - Runs directly over TCP or other reliable transport layer
- Implementation
  - OpenLDAP release distribution
    - Stand-alone LDAP daemon: SLAPD server
  - Binary to ASCII conversion
    - Base-64 encoding/decoding algorithm

Client Module

- Functions
  - Generates public/private-key pairs
    - RSA key pair
    - Size of modulus: 1024bit
  - Encrypts the private-key and saves into the local machine
    - Password-based encryption
  - Generates certification requests
    - PKCS #10
  - Communicates with the CA server to send certification request
- Implementation
  - Visual Basic Script language
  - Microsoft ActiveX control programming
  - Socket programming

CA Server

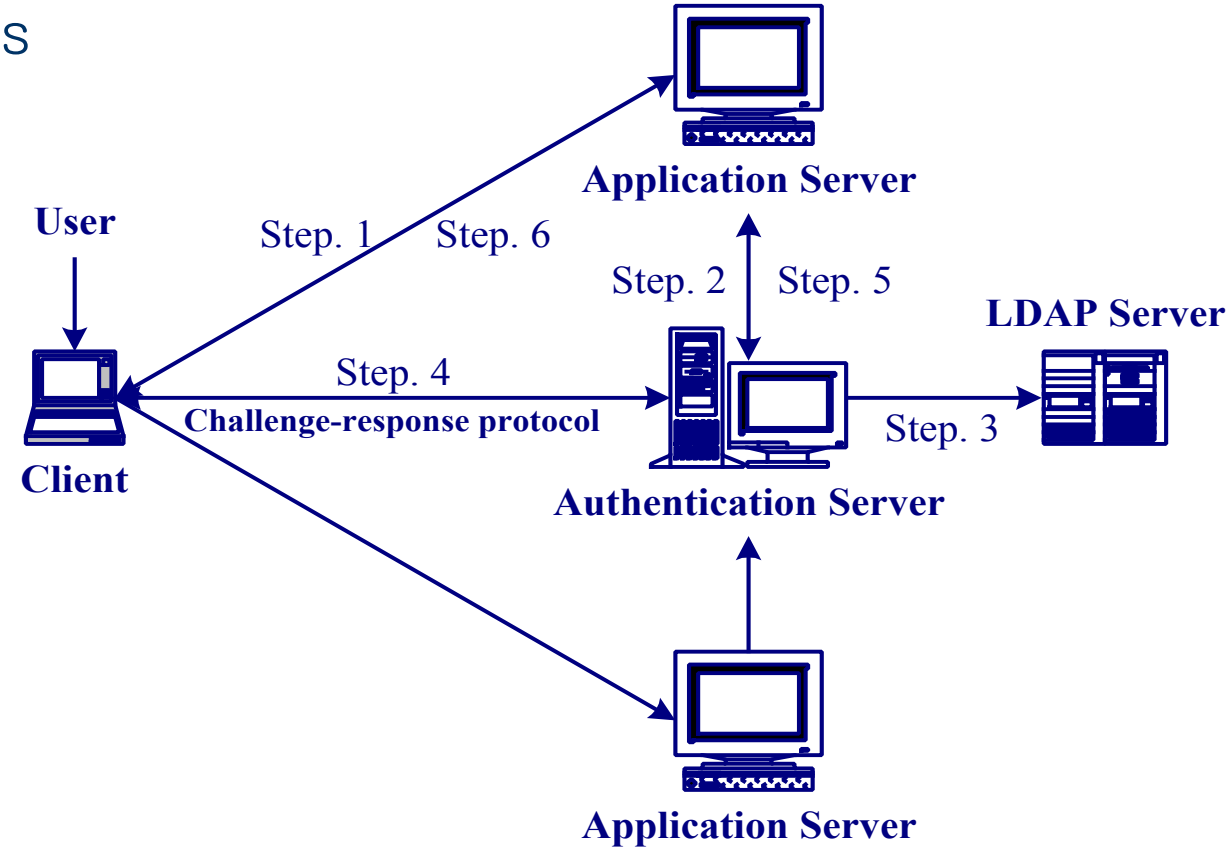

인증서 신청 - Microsoft Internet Explorer
파일(F) 편집(E) 보기(V) 즐겨찾기(A) 도구(T) 도움말(H)
뒤로 앞으로 중지 새로고침 홈 검색 즐겨찾기 목록보기 메일 크기 인쇄 편집 토론 전화걸기 Real.com Messenger
주소(D) http://hyde.yonsei.ac.kr/ca/apply.html
이동 연결
인증서 신청
인증서 신청페이지입니다. 모든 정보는 영어로 입력하여 주시기 바랍니다.
입력하신 내용(E-Mail 제외)은 네트워크상으로 전송되지 않습니다.
이름
국가
시,도
기관(회사)
전화번호
비밀번호는 개인키를 암호화하여 저장하기 위한 Pass-Phrase로 사용됩니다.(영문 20자, 한글 10자)
비밀번호
비밀번호 확인
인증서 발급 및 변경, 취소를 위한 PIN 번호를 받으시기 위해서는 E-Mail 주소를 정확히 기입하셔야 합니다.
E-Mail
신청
재입력
인증서 신청이 처리 되는대로 E-Mail을 통해 PIN 번호를 보내드리겠습니다.
인증서 발급페이지에서 PIN 번호를 이용하여 인증서를 발급받아 사용하실 수 있습니다.
완료
인터넷

Contents

- Introduction
- Previous Works
- Design and Implementation of Public-Key Infrastructure
- ✓ Design and Implementation of Single Sign-On
- Conclusion

The Proposed Single Sign-On

- Components
  - Authentication server
  - Application servers
  - Client modules



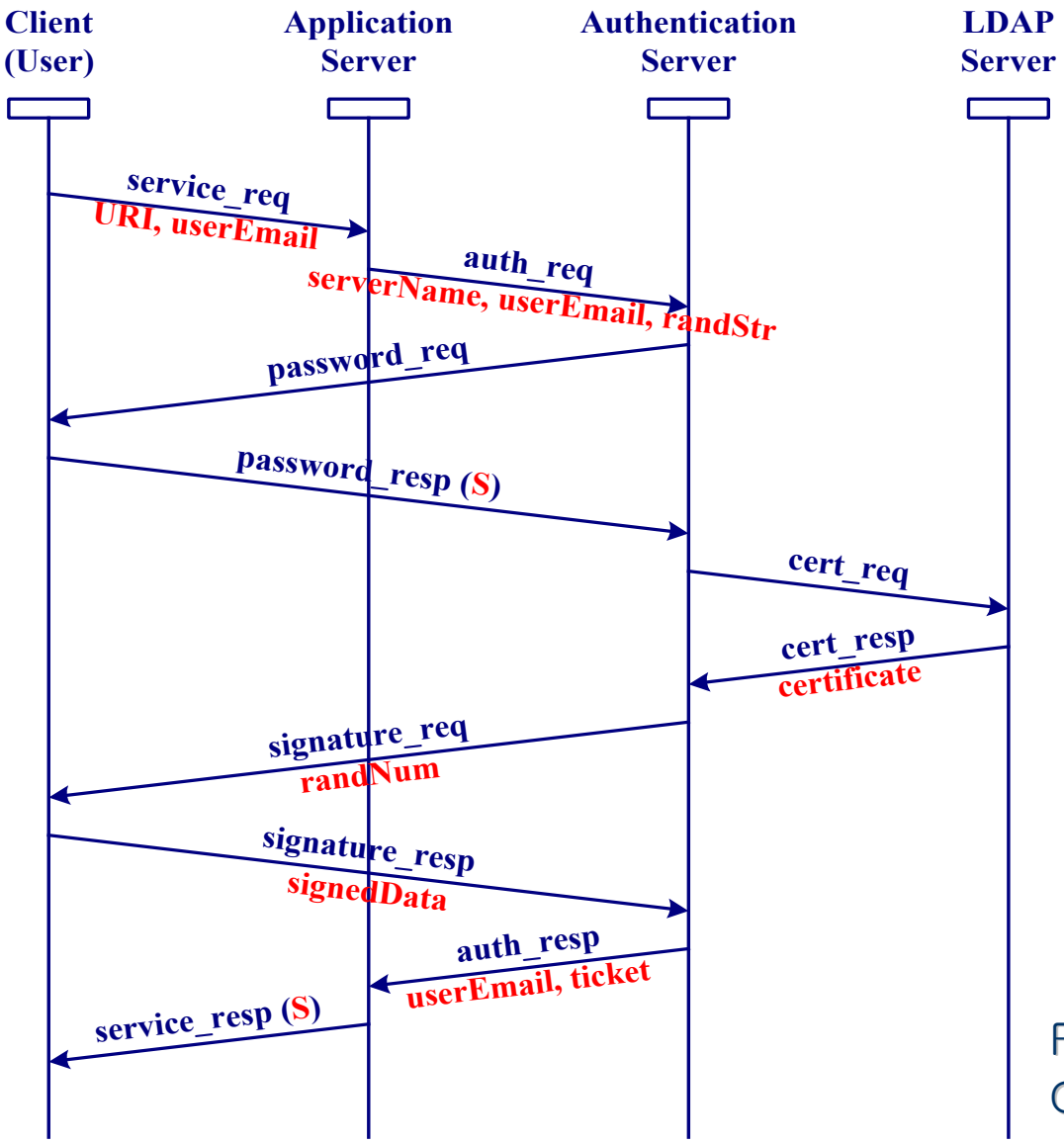
Requirements Capture

Message Sequence Chart

- Guide for designing the system
- Visualization of the system runs

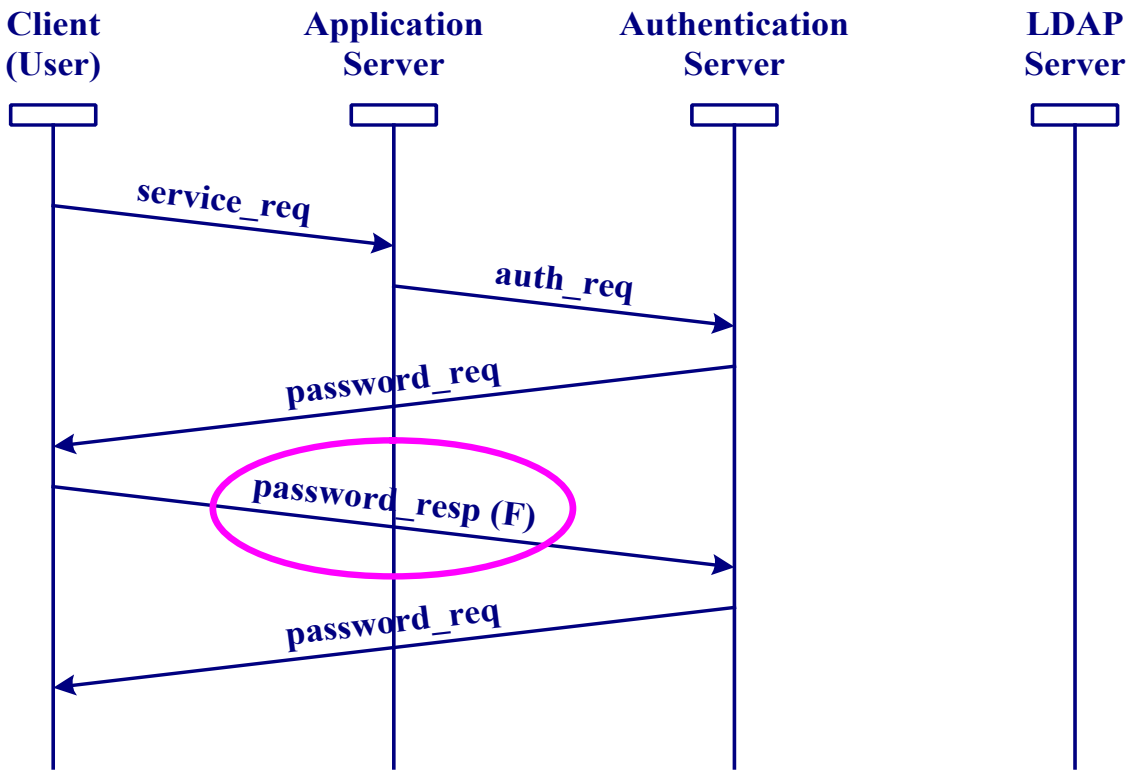
Messages	Subjects	Parameters
service_req	Client (User)	URI, userEmail
service_resp	↔ Application Server	flag
auth_req	Application Server	serverName, userEmail, randStr
auth_resp	↔ Authentication Server	userEmail, ticket
password_req	Authentication Server	–
password_resp	↔ Client (User)	flag
cert_req	Authentication Server	–
cert_resp	↔ LDAP Server	certificate
signature_req	Authentication Server	randNum
signature_resp	↔ Client (User)	signedData

Scenario 1



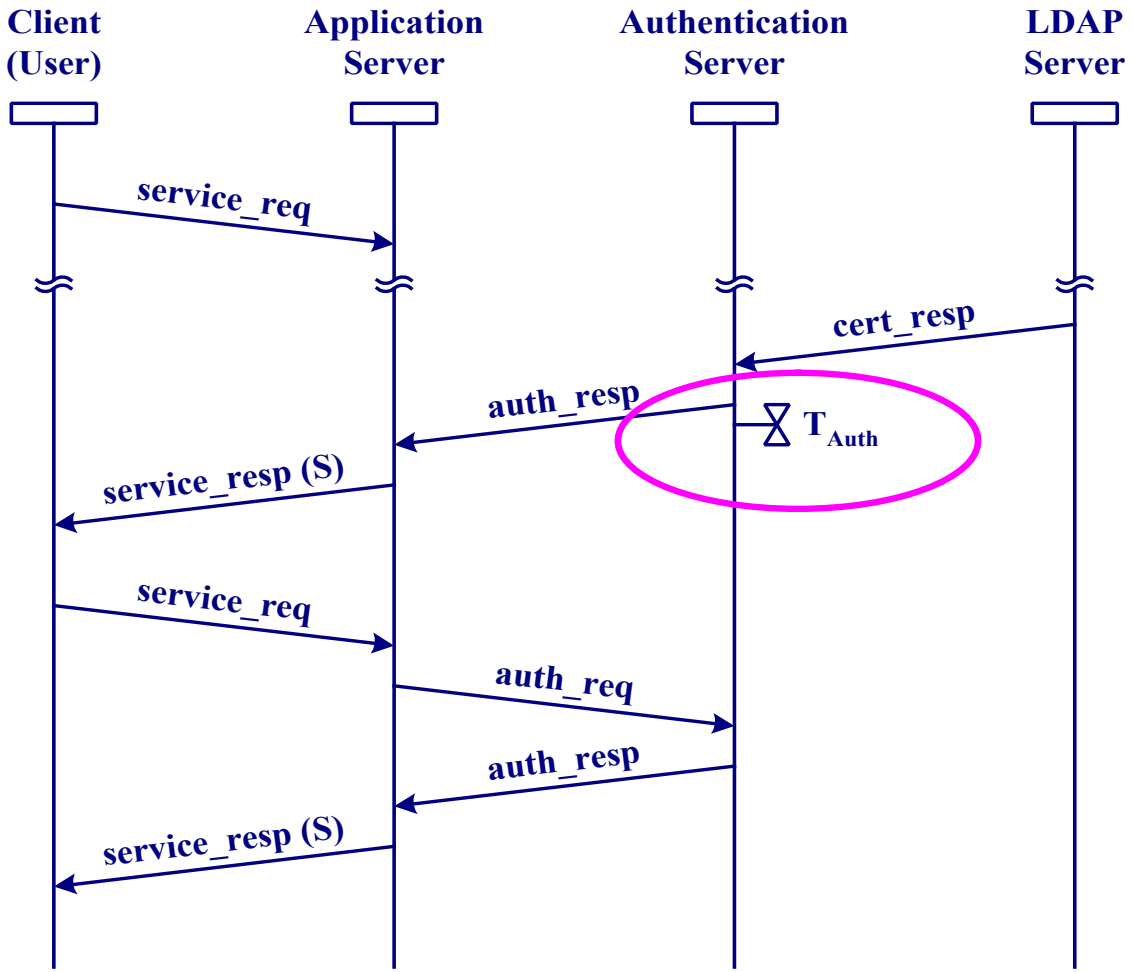
First login:
General case

Scenario 2



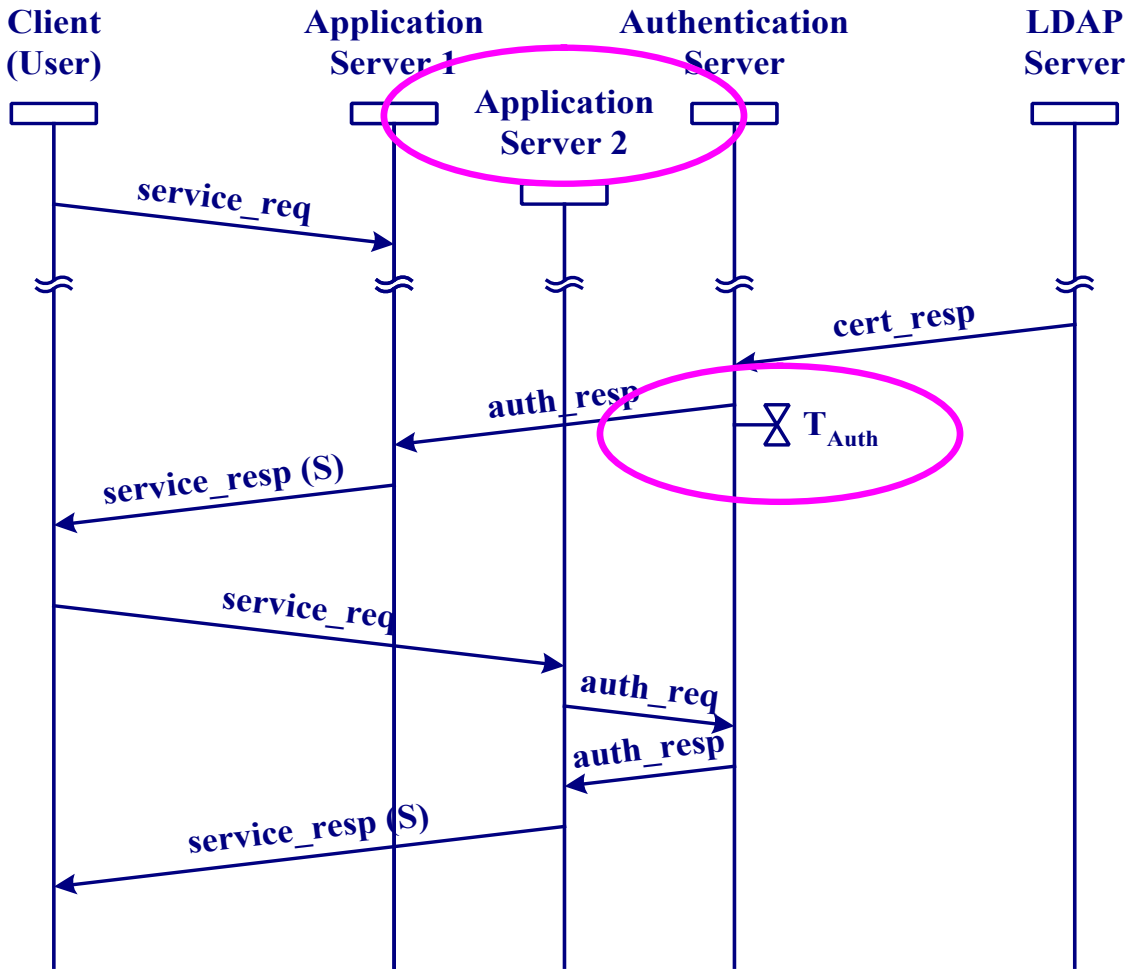
In case a user enters invalid password

Scenario 3



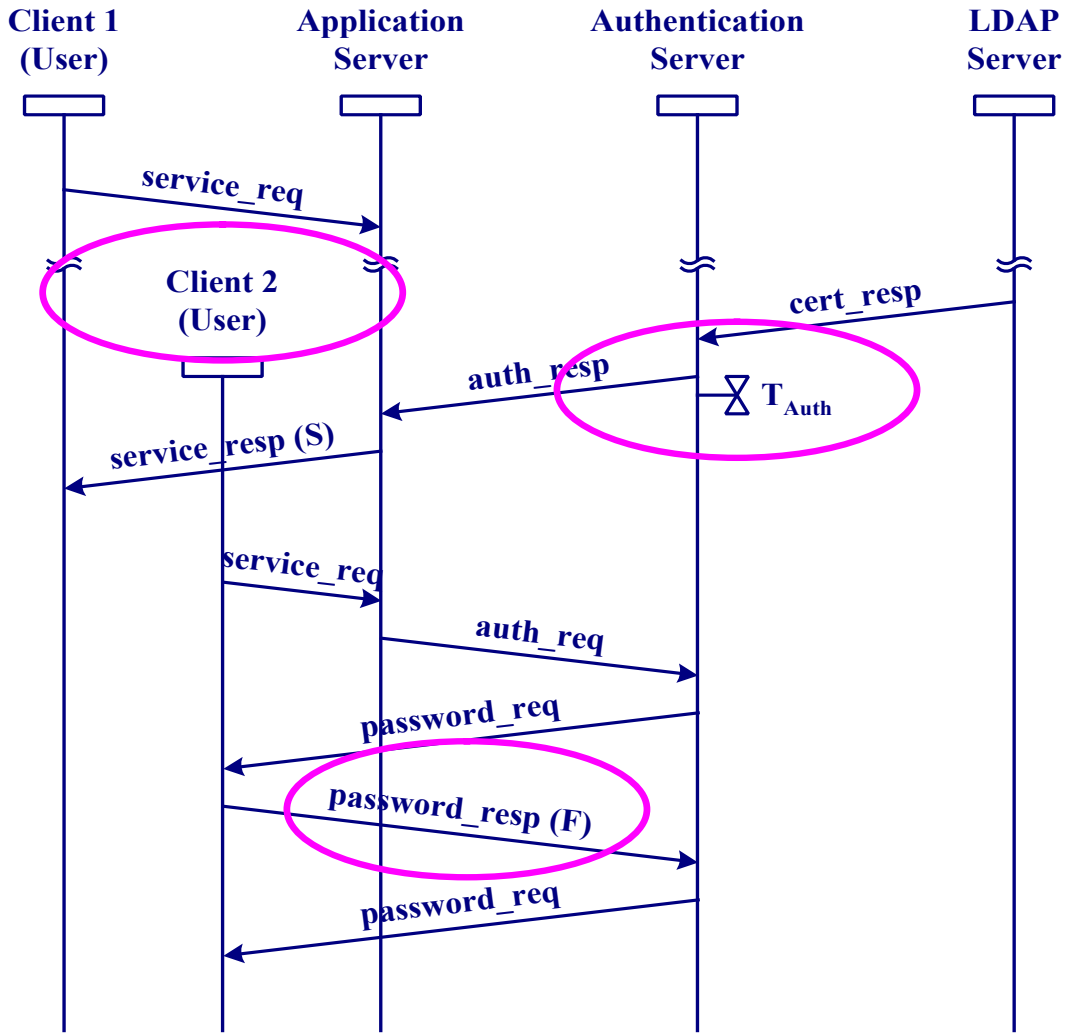
In case a user accesses the same application server from the same client within timer expiration

Scenario 4



In case a user accesses another application server from the same client within timer expiration

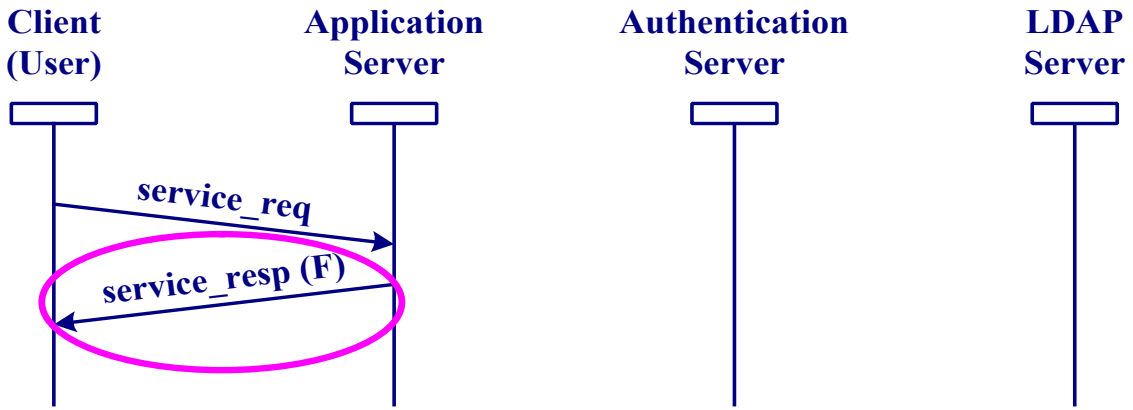
Scenario 5



In case an opponent accesses the application server from the another client within timer expiration

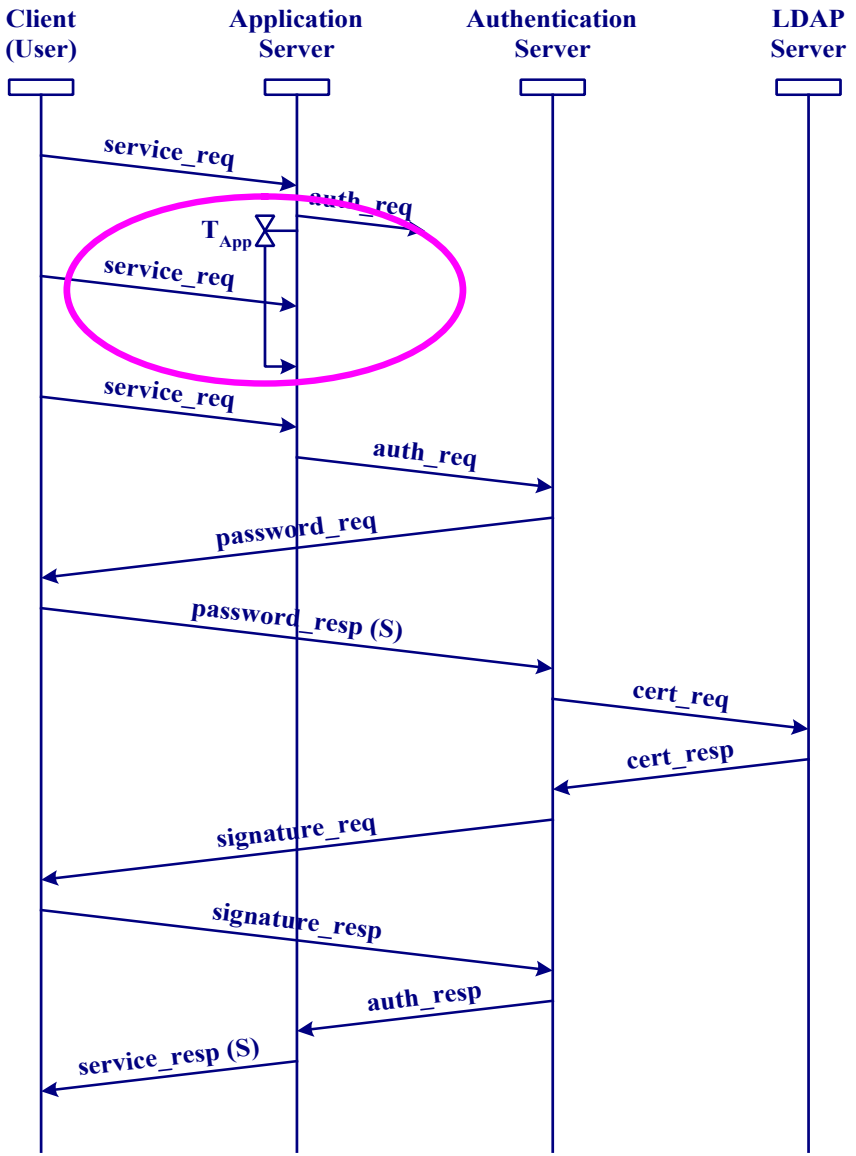
Scenario 6

- randStr & ticket
  - Application server can be accessed through two URIs
  - To prevent an opponent from directly accessing an application server without authentication protocol
  - Can prevent server-spoofing attack
  - ticket: randStr encrypted by authentication server



In case an opponent directly accesses an application server without authentication process

Scenario 7



In case a user sends multiple service_req messages

Authentication Server

- Functions
  - Challenge-response protocol
  - Generates the ticket
    - DES-CBC with all 0's IV
    - To transmit as CGI parameter
      - Base-64 encoding algorithm
      - Additional encoding: + → -, / → @, = → :
  - Control of the timer(T_{Auth})
    - Implemented by checking the log-in time in the database
    - 1 hour
- Implementation
  - Apache web server
  - CGI programming
  - Database management system
    - MySQL database

Application Server

- Functions
  - Verifies the ticket
    - DES-CBC with all 0's IV
    - Base-64 decoding algorithm
  - Control of the timer(T_{App})
    - Maximum Segment Lifetime(MSL)
    - 30 seconds
- Implementation
  - Apache web server
  - CGI programming
  - Redirection
    - Location header of HTTP

Client Module

- Functions
  - Decrypts the private-key
    - Password-based decryption
  - Challenge-response protocol
    - RSA signature algorithm with MD5 hash function
    - Size of modulus: 1024bit
- Implementation
  - Visual Basic Script language
  - Microsoft ActiveX control programming
  - Socket programming

Conclusion

- Advantages
  - Has the mechanism that a user directly connects the application server he wants to access
  - Independent of lower layer protocol implementations
  - Easy to implement and use in the Intranet
- Future Research
  - Development of SSO for other application protocols
    - Current: HTTP
    - Future: Telnet, customized application protocols
  - Extension to multiple-CA environment
  - Use of smart card