Bài 3. Các ứng dụng xác thực

Nội dung

- Mục tiêu của các ứng dụng xác thực
- Phân loại các ứng dụng xác thực
- Mô hình Kerberos
- Mô hình X.509

3.1. Mục tiêu

- Hỗ trợ các dịch vụ xác thực và chữ ký số ở mức ứng dụng
- Cung cấp các mô hình để xây dựng các ứng dụng thực tế

3.2. Phân loại

- Phân làm 2 loại chính
  - Dựa trên mã hóa đối xứng
    - Mô hình Kerberos
    - Giao thức Needham-Schroeder
  - Dựa trên khóa công khai được chứng thực
    - Mô hình X.509

3.3. Mô hình Kerberos

- Hệ thống dịch vụ xác thực phát triển bởi MIT (Học viện công nghệ Massachusets)
- Giao thức đã được phát triển dưới nhiều phiên bản, trong đó các phiên bản từ 1 đến 3 chỉ dùng trong nội bộ MIT.

3.3. Mô hình Kerberos

- Dùng để xác thực các máy tính trước khi cho phép sử dụng dịch vụ
- Nhằm đối phó với các hiểm họa sau
  - Người dùng giả danh là người khác
  - Người dùng thay đổi địa chỉ mạng của client
  - Người dùng xem trộm thông tin trao đổi và thực hiện kiểu tấn công lặp lại

3.3. Mô hình Kerberos

- Được sử dụng mặc định trong các hệ điều hành Windows (2000, XP, 2003), Mac OS
- Một số phần mềm sử dụng Kerberos:
  - OpenSSH
  - Apache

3.3.1. Mô hình tổng quan của Kerberos

- Giao thức xây dựng trên hệ mật mã đối xứng
- Xác thực qua một bên thứ ba được tin tưởng, còn gọi là "trung tâm phân phối khóa"
  - Máy chủ xác thực (*authentication server* - AS)
  - Máy chủ cung cấp thẻ (*ticket granting server* - TGS)

3.3.1. Mô hình tổng quan của Kerberos

- Dịch vụ được cung cấp qua các server dịch vụ phân tán
  - Giải phóng chức năng xác thực khỏi các server dịch vụ và client

3.3.1. Mô hình tổng quan của Kerberos

- Giao thức xác thực đơn giản

  (1) $C \rightarrow AS : ID_C \parallel P_C \parallel ID_V$

  (2) $AS \rightarrow C :$ Thẻ

  (3) $C \rightarrow V : ID_C \parallel$ Thẻ

  Thẻ = $E_{K_V}[ID_C \parallel AD_C \parallel ID_V]$

- Hạn chế
  - Mật khẩu truyền từ C đến AS không được bảo mật
  - Nếu thẻ chỉ sử dụng được một lần thì phải cấp thẻ mới cho mỗi lần truy nhập cùng một dịch vụ
  - Nếu thẻ sử dụng được nhiều lần thì có thể bị lấy cắp để sử dụng trước khi hết hạn
  - Cần thẻ mới cho mỗi dịch vụ khác nhau

3.3.1. Mô hình tổng quan của Kerberos

- Kerberos đưa ra giao thức xác thực an toàn hơn, bằng cách sử dụng 2 loại máy chủ:
  - Máy chủ xác thực
  - Máy chủ cung cấp thẻ

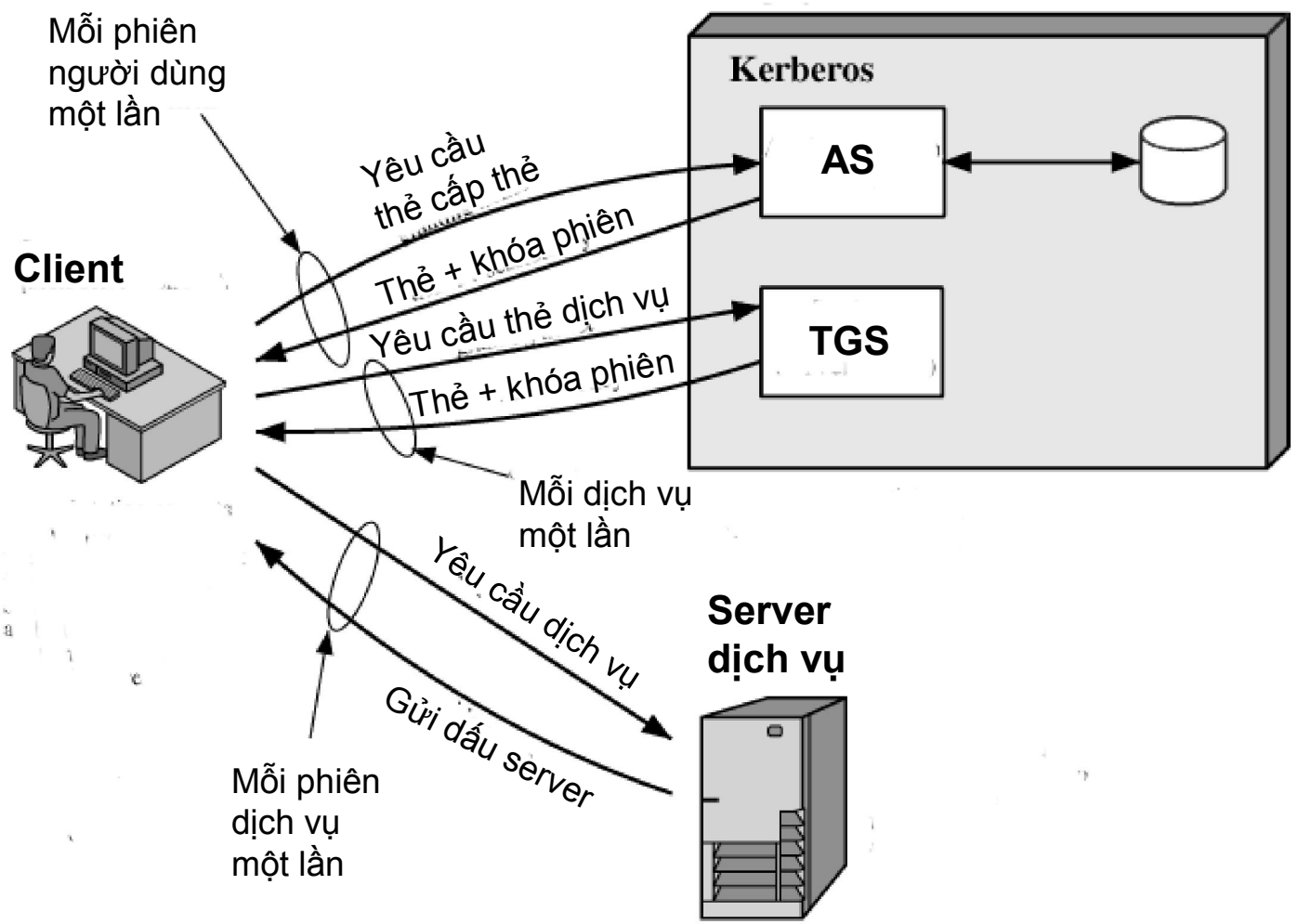
3.3.1. Mô hình tổng quan của Kerberos

- Máy chủ xác thực:
  - Lưu danh sách và khóa bí mật của người dùng
  - Xác thực người dùng trước khi cho phép sử dụng máy chủ cấp thẻ

3.3.1. Mô hình tổng quan của Kerberos

- Máy chủ cung cấp thẻ
  - Cung cấp cho người sử dụng các thẻ dịch vụ

3.3.1. Mô hình tổng quan của Kerberos



Giao thức xác thực trong Kerberos 4

(a) Trao đổi với dịch vụ xác thực : để có thẻ xác thực

(1) $C \rightarrow AS : \ ID_C \parallel ID_{tgs} \parallel TS_1$

(2) $AS \rightarrow C : \ E_{K_C}[K_{C,tgs} \parallel ID_{tgs} \parallel TS_2 \parallel Hạn_2 \parallel Thẻ_{tgs}]$

$Thẻ_{tgs} = E_{K_{tgs}}[K_{C,tgs} \parallel ID_C \parallel AD_C \parallel ID_{tgs} \parallel TS_2 \parallel Hạn_2]$

(b) Trao đổi với dịch vụ cấp thẻ : để có thẻ dịch vụ

(3) $C \rightarrow TGS : \ ID_V \parallel Thẻ_{tgs} \parallel Dấu_C$

(4) $TGS \rightarrow C : \ E_{K_{C,tgs}}[K_{C,V} \parallel ID_V \parallel TS_4 \parallel Thẻ_V]$

$Thẻ_V = E_{K_V}[K_{C,V} \parallel ID_C \parallel AD_C \parallel ID_V \parallel TS_4 \parallel Hạn_4]$

$Dấu_C = E_{K_{C,tgs}}[ID_C \parallel AD_C \parallel TS_3]$

(c) Trao đổi xác thực client/server : để có dịch vụ

(5) $C \rightarrow V : \ Thẻ_V \parallel Dấu_C$

(6) $V \rightarrow C : \ E_{K_{C,V}}[TS_5 + 1]$

$Dấu_C = E_{K_{C,V}}[ID_C \parallel AD_C \parallel TS_5]$

Ký hiệu

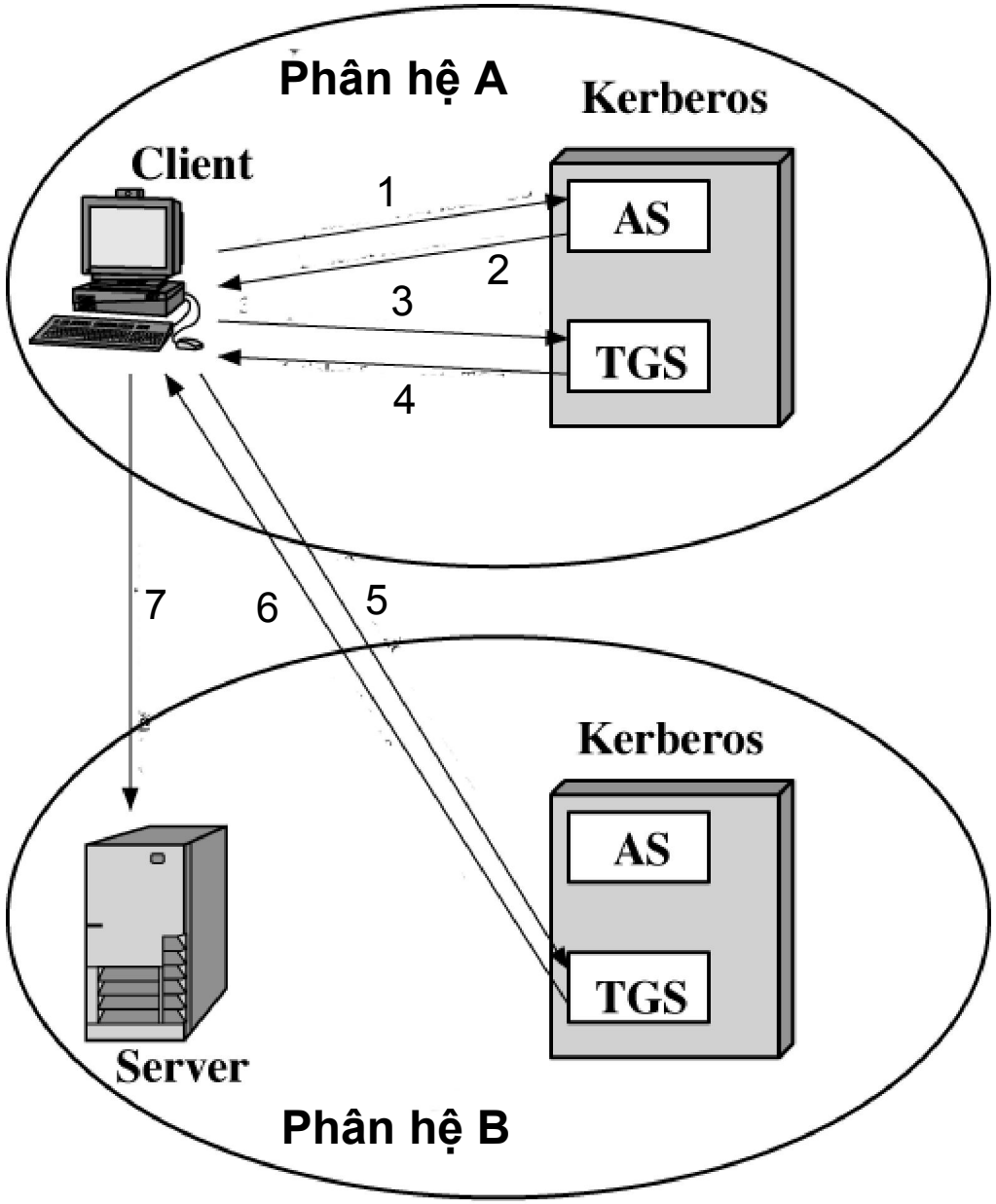
- C : Client
- AS : Server xác thực
- V : Server dịch vụ
- ID_C : Danh tính người dùng trên C
- ID_V : Danh tính của V
- P_C : Mật khẩu của người dùng trên C
- AD_C : Địa chỉ mạng của C
- K_V : Khóa bí mật chia sẻ bởi AS và V
- $\|$: Phép ghép
- TGS : Server cấp thẻ
- TS : Nhãn thời gian

3.3.2. Phân hệ Kerberos

- Mô hình Kerberos có thể được cài đặt ở nhiều vùng riêng biệt có liên hệ với nhau. Mỗi vùng được gọi là một phân hệ
- Một phân hệ Kerberos bao gồm
  - Một server Kerberos chứa trong CSDL danh tính và mật khẩu băm của các thành viên
  - Một số người dùng đăng ký làm thành viên
  - Một số server dịch vụ, mỗi server có một khóa bí mật riêng chỉ chia sẻ với server Kerberos

3.3.2. Phân hệ Kerberos

- Hai phân hệ có thể tương tác với nhau nếu 2 server chia sẻ 1 khóa bí mật và đăng ký với nhau
  - Điều kiện là phải tin tưởng lẫn nhau



1. Yêu cầu thẻ cho TGS cục bộ
2. Thẻ cho TGS cục bộ
3. Yêu cầu thẻ cho TGS ở xa
4. Thẻ cho TGS ở xa
5. Yêu cầu thẻ cho server ở xa
6. Thẻ cho server ở xa
7. Yêu cầu dịch vụ ở xa

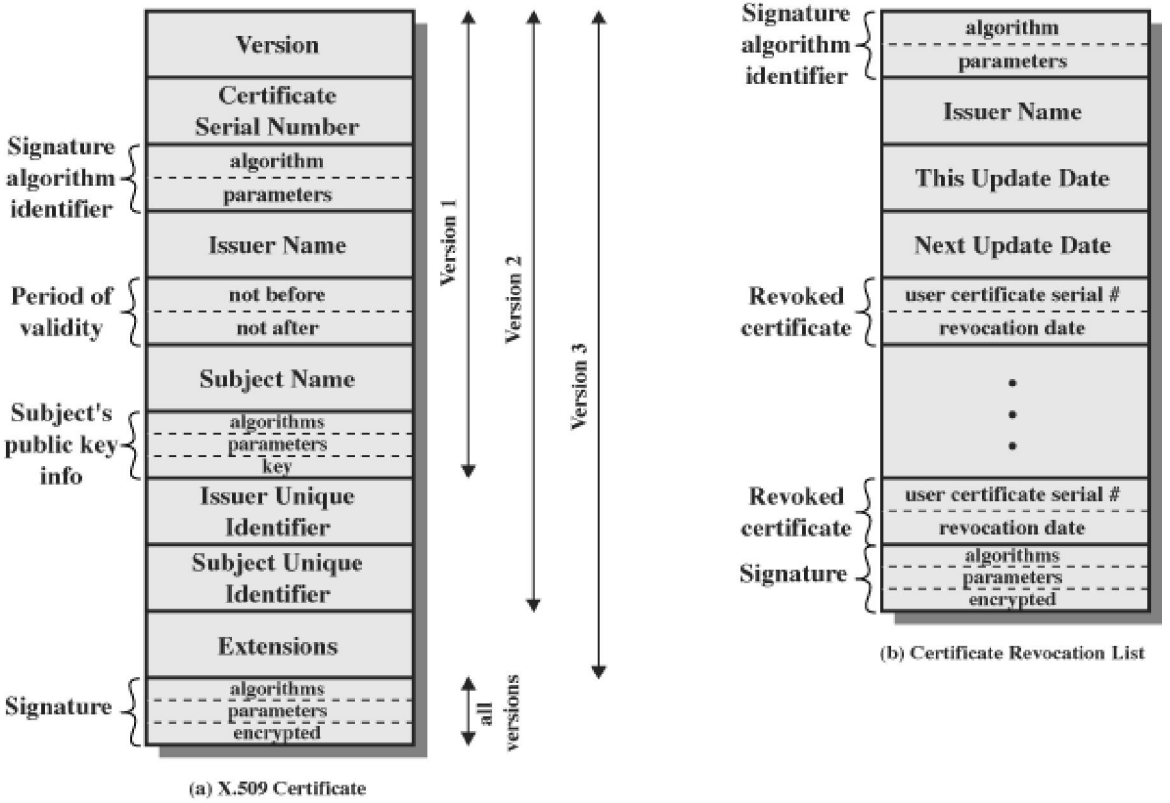
3.4. Mô hình X.509

- Nằm trong loạt khuyến nghị X.500 của ITU-T nhằm chuẩn hóa dịch vụ thư mục khóa công khai
- Công bố lần đầu tiên vào năm 1988
- Sử dụng mật mã khóa công khai và chữ ký số
  - Không chuẩn hóa giải thuật nhưng khuyến nghị RSA

3.4. Mô hình X.509

- Định ra một cơ cấu cho dịch vụ xác thực
  - Danh bạ chứa các chứng thực khóa công khai
  - Mỗi chứng thực bao gồm khóa công khai của người dùng ký bởi một bên chuyên trách chứng thực đáng tin
- Đưa ra các giao thức xác thực

3.4.1. Khuôn dạng chứng thực X.509



(a) X.509 Certificate

(b) Certificate Revocation List

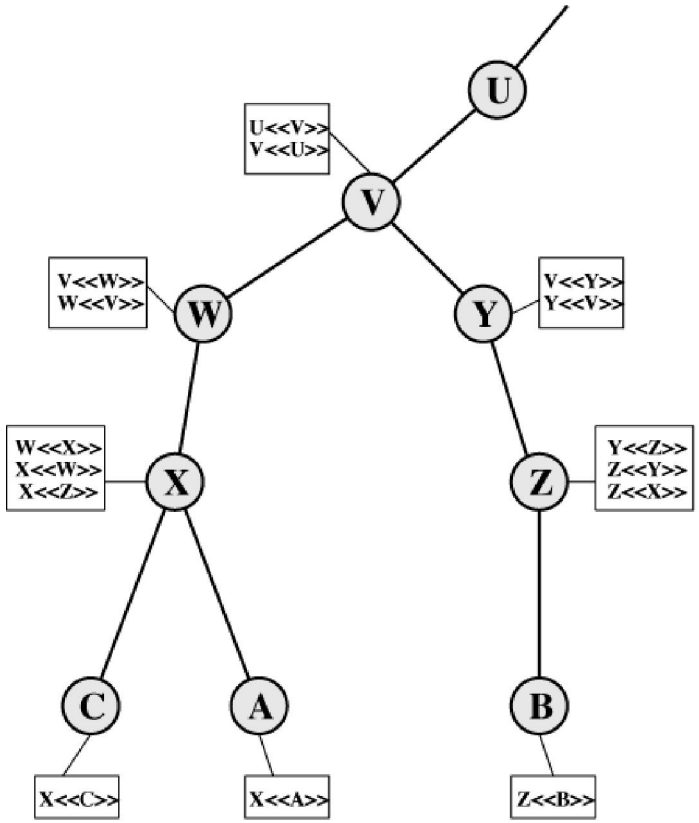
3.4.2. Đặc điểm X.509

- Xác minh chứng thực bằng khóa công khai của CA
- Chỉ CA mới có thể thay đổi chứng thực
  - Chứng thực có thể đặt trong một thư mục công khai

3.4.2. Đặc điểm X.509

- Sử dụng cấu trúc phân cấp CA
  - Người dùng được chứng thực bởi CA đã đăng ký
  - Mỗi CA có hai loại chứng thực
    - Chứng thực thuận : Chứng thực CA hiện tại bởi CA cấp trên
    - Chứng thực nghịch : Chứng thực CA cấp trên bởi CA hiện tại
- Cấu trúc phân cấp CA cho phép người dùng xác minh chứng thực bởi bất kỳ CA nào

3.4.2. Đặc điểm X.509



3.4.3. Thu hồi chứng thực

- Mỗi chứng thực có một thời hạn hợp lệ
- Có thể cần thu hồi chứng thực trước khi hết hạn
  - Khóa riêng của người dùng bị tiết lộ
  - Người dùng không còn được CA chứng thực
  - Chứng thực của CA bị xâm phạm
- Mỗi CA phải duy trì danh sách các chứng thực bị thu hồi (CRL)
- Khi nhận được chứng thực, người dùng phải kiểm tra xem nó có trong CRL không