Modellazione di Kerberos mediante DASM

Sommario

- Descrizione Kerberos
- Modellazione
- Analisi di Correttezza

Descrizione Kerberos

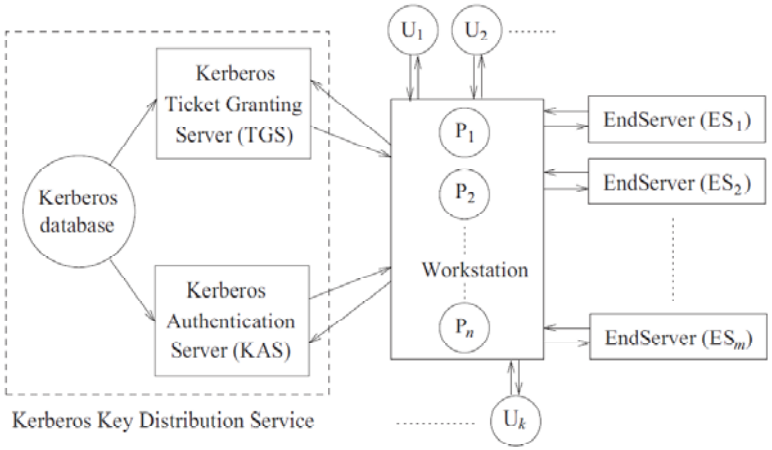
Kerberos (1)

- Kerberos è traslitterazione latina del mostro della mitologia greca Κέρβερος
  - In italiano Cerbero
  - È un mostro a tre teste posto a guardia dell'Ade
- Aspetti chiave:
  - Tre teste
  - Impedire ai non autorizzati l'accesso all'Ade

Kerberos (2)

- Il sistema Kerberos è stato sviluppato al MIT all'interno del progetto Athena
- È un sistema di autenticazione distribuita che permette a un client di dimostrare la propria identità a un server senza inviare dati attraverso la rete
- È basato su un server centralizzato, la cui funzione è quella di garantire
  - l'autenticazione dei client per i server
  - l'autenticazione dei server per i client

Architettura (1)



Architettura (2)

- Comprende tre componenti logiche principali (tre teste)
  - Workstation su cui sono eseguiti i processi P_1, P_2, …, P_n, per conto dell'insieme di utenti U_1, U_2, …, U_k
  - Kerberos Key Distribution Service (o solo Kerberos), a sua volta costituito da
    - Kerberos Authentication Server – KAS
    - Ticket Granting Server - TGS
  - Un insieme di End Server ES_1, ES_2, …, ES_m

Architettura (3)

- All'interno della componente Kerberos c'è il DB delle chiavi dei client e dei server
  - Ogni chiave è la versione crittografata secondo DES della passwd dell'utente che accede a un client
- I nomi (ID) dei client/server sono le uniche informazioni identificative che circolano in chiaro sulla rete

Funzionamento (1)

- Affinché un client possa usare un servizio deve produrre al server che lo fornisce un ticket precedentemente ottenuto da Kerberos
  - Il ticket è una stringa di bit, crittografata mediante la chiave privata del server, che contiene l'identità del client che ha fatto la richiesta
  - Il server che ha ricevuto un ticket è certo dell'identità del client

Funzionamento (2)

- Per ogni servizio richiesto è necessario un ticket
  - All'inizio della sessione il client potrebbe non conoscere l'insieme di tutti i servizi che dovrà richiedere
- Per risolvere il problema il client potrebbe richiedere all'inizio tutti i possibili ticket, uno per ogni possibile servizio
  - Soluzione inefficiente

Funzionamento (3)

- Per superare il problema il client ottiene da Kerberos al momento del login un ticket a lungo termine chiamato authentication ticket
  - È usato dal client ogni volta che deve accedere a un servizio, inviandolo al TGS di Kerberos
  - TGS riconosce l'identità grazie al ticket di autenticazione e fornisce il ticket per il servizio

Funzionamento (4)

- Oltre al ticket, il client riceve da Kerberos anche una session key con cui crittografare le comunicazioni verso il server

Il Modello

Generalità

- Il sistema è modellato mediante una DASM, costituita da quattro ASM:
  - MessagePassing
  - EncryptionDecryption
  - MultipleClients
  - MultipleEndServers

MessagePassing – Global Signature (1)

- È il modello che formalizza il formato dei messaggi scambiati tra gli agenti
  - In prima istanza consideriamo un modello semplificato, contenente un solo Client (C) e un solo EndServer (ES)
- La DASM relativa comprende una ASM per ciascuno dei 4 agenti del modello
  - Il KAS; il Client C; il TGS; l'EndServer ES
- Quindi, l'universo AGENT={KAS, C, TGS, ES}

MessagePassing – Global Signature (2)

- L'Universo MESSAGE contiene messaggi di due tipi, composti (concatenazione di dati) o crittografati (secondo una key), indicati rispettivamente con {X, X'} e $\{X, X'\}_{key}$
  - Per conoscere il tipo di un messaggio si usa la funzione type: MESSAGGE → {cleartext, encrypted}

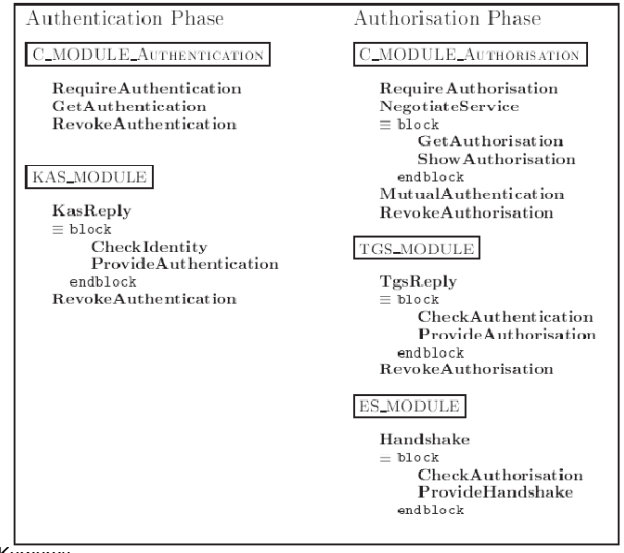
MessagePassing – Global Signature(3)

- Le chiavi, i ticket e le autenticazioni sono modellati da oggetti atomici
  - appartenenti rispettivamente agli universi KEY, TICKET e AUTH
- La funzione K: AGENT → KEY restituisce la chiave privata dell'agente, conosciuta solo dall'agente stesso e registrata nel db delle chiavi

MessagePassing – Global Signature (4)

- Le chiavi, i ticket e le autenticazioni sono spediti all'interno dei messaggi, da cui vengono estratti mediante le funzioni:
  - ExtractKey: MESSAGE → KEY
  - ExtractTicket: MESSAGE → TICKET
  - ExtractAuth: MESSAGE → AUTH
- Altre funzioni
  - sender, receiver: MESSAGE→ AGENT
  - mode: AGENT → {ReadyToSend, ReadyToReceive, ReadyToStart}

MessagePassing – Specifica



EncryptionDecryption – Global Signature

- Gli oggetti astratti del modello precedente sono raffinati dalle procedure di crittografia-decrittografia usate per costruire le credenziali del client
- Le funzioni sono:
  - encrypt: DATA X KEY → CRYPTDATA
  - decrypt: CRYPTDATA X KEY → DATA

  tali che

  decrypt(encrypt(t,k), k')=t se k=k',
  =undef altrimenti

Correttezza (1)

- Condizioni globali: ad ogni livello di specifica devono valere le seguenti
  - G1: defined(K(C)) & defined(K(TGS)) & defined(K(ES))
  - G2: NOT authenticated(C) ➔ NOT authorised (C)
- Condizioni Iniziali
  - Specifica degli stati

Correttezza (2)

- Condizioni di lavoro:
  - W1: monotonia del clock
  - W2: correttezza della password
  - W3: limite superiore al tempo di reazione del server
  - W4: validità delle credenziali del client

Correttezza (3)

- Esecuzione regolare: È regolare ogni esecuzione tale che
  - Le condizioni globali G1 e G2 sono soddisfatte
  - Le condizioni iniziali soddisfano la specifica degli stati
  - Le condizioni di lavoro W1, W2, W3, W4 sono soddisfatte

Ulteriori Raffinamenti

- La gestione di più client è ottenuta da un raffinamento che produce il modello MultipleClients
- La gestione di più end server è ottenuta da un raffinamento che produce il modello MultipleEndServers
- La simulazione di possibili minacce al sistema è ottenuta realizzando modelli capaci di effettuare operazioni di spionaggio

Bibliografia

- G. Bella, E. Riccobene, “Formal Analysis of the Kerberos Authentication System”, *Journal of Universal Computer Science*, vol. 3, no. 12 (1997), pp.1337-1381