Sistemas Operativos 2

Unidad 5: Seguridad

Sistema Kerberos

René Ornelis
Primer semestre de 2025

Contenido

Índice de figuras

Sistema Kerberos

1 Historia

Kerberos es un protocolo de autenticación diseñado para ofrecer seguridad en redes de computadoras. Su principal objetivo es permitir que los usuarios y servicios se identifiquen de manera segura en un entorno no seguro, como una red pública. Kerberos fue creado por el Instituto de Tecnología de Massachusetts (MIT) en 1983, con el objetivo de tener un sistema integrado que cumpliera con las 3 funciones de seguridad. Fue diseñado para proporcionar autenticación segura en redes informáticas, evitando el envío de contraseñas en texto plano y protegiendo contra ataques de suplantación de identidad. También se buscaba que un usuario manejara una sola contraseña para todos los servicios (*single signon*) y evitar la *pesadilla de contraseñas*.

2 Arquitectura

Los componentes de Kerberos se muestran en la Figura 1 y son:

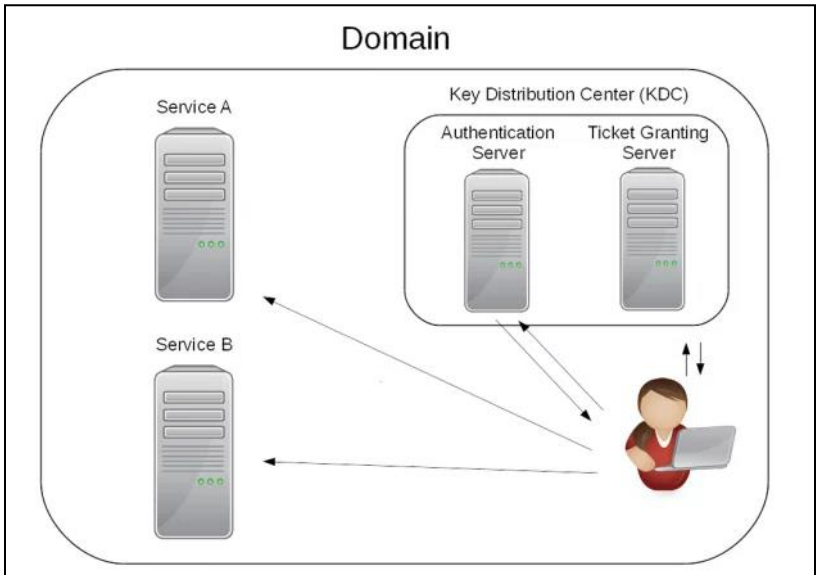

Figura 1: Arquitectura de Kerberos

- **Cliente:** el usuario o proceso que se desea autenticar y utilizar servicios.
- **Servidor de servicios:** Servidores que prestan diferentes servicios, cuyo acceso es válido por Kerberos.
- **Autenticación (AS, Authentication Service):** Es el primer punto de contacto para el cliente. Verifica la identidad del usuario y emite un ticket de autenticación (TGT) si las credenciales son correctas. Provee el medio para autenticar al usuario y proporcionar un ticket, que certifica la autenticidad del usuario
- **Tickets (TGS, Ticket Granting Service):** Basado en el ticket, determina el acceso del usuario a los diferentes servicios
- **Key Distribution Center (KDC)**: Es la union de Authentication Server (AS) y Ticket Granting Server (TGS).

Un servidor Kerberos se denomina KDC (Kerberos Distribution Center), y provee de dos servicios fundamentales:

Se tiene tres componentes básicos de seguridad:

1. **La clave de sesión:** es una clave secreta generada por Kerberos y expedida a un cliente para uso con un servidor durante una sesión; no es obligatorio utilizarla en toda la comunicación con el servidor, sólo si el servidor lo requiere (porque los datos son confidenciales) o si el servidor es un servidor de autenticación. Las claves de sesión se utilizan para minimizar el uso de las claves secretas de los diferentes agentes: éstas últimas son válidas durante mucho tiempo, por lo que es conveniente para minimizar ataques utilizarlas lo menos posible.
2. **El ticket:** es un testigo expedido a un cliente del servicio de tickets de Kerberos para solicitar los servicios de un servidor; garantiza que el cliente ha sido autenticado recientemente. A un ticket de un cliente C para acceder a un servicio S se le denomina T_{CS}. Este ticket incluye el nombre del cliente C, para evitar su posible uso por impostores, un periodo de validez y una clave de sesión asociada para uso de cliente y servidor. Kerberos siempre proporciona el ticket ya cifrado con la clave secreta del servidor al que se le entrega.
3. **El autenticador:** es un testigo construido por el cliente y enviado a un servidor para probar su identidad y la actualidad de la comunicación; sólo puede ser utilizado una vez. Un autenticador de un cliente C ante un servidor S se denota por A_{CS}. Este autenticador contiene, cifrado con la clave de la sesión, el nombre del cliente y un timestamp.

3 Funcionamiento

Tal como se ilustra en la Figura 2, el proceso de autenticación y autorización para que un cliente pueda acceder a un servicio es:

1) Un usuario ingresa su nombre de usuario y password en el cliente
2) El cliente genera una clave hash a partir del password y la usará como la clave secreta del cliente: K_c
3) El cliente envía un mensaje en texto plano al AS solicitando servicio en nombre del usuario: ***m0: usuario, timestamp***.
4) El AS comprueba si el cliente está en su base de datos. Si es así, se genera una clave de sesión para usar con el TGS: ***Kc-tgs.*** Luego el AS envía dos mensajes al cliente:
   a) Mensaje A: Client/TGS session key cifrada usando la clave secreta del usuario: $\{Kc\text{-}tgs\}K_c$
   b) Mensaje B: Ticket-Granting Ticket (que incluye el ID de cliente, la dirección de red del cliente, el período de validez y el Client/TGS session key) cifrado usando la clave secreta del TGS. $\{Kc\text{-}tgs\ ,\ usuario,\ ip,\ vigencia\}K_{tgs}$

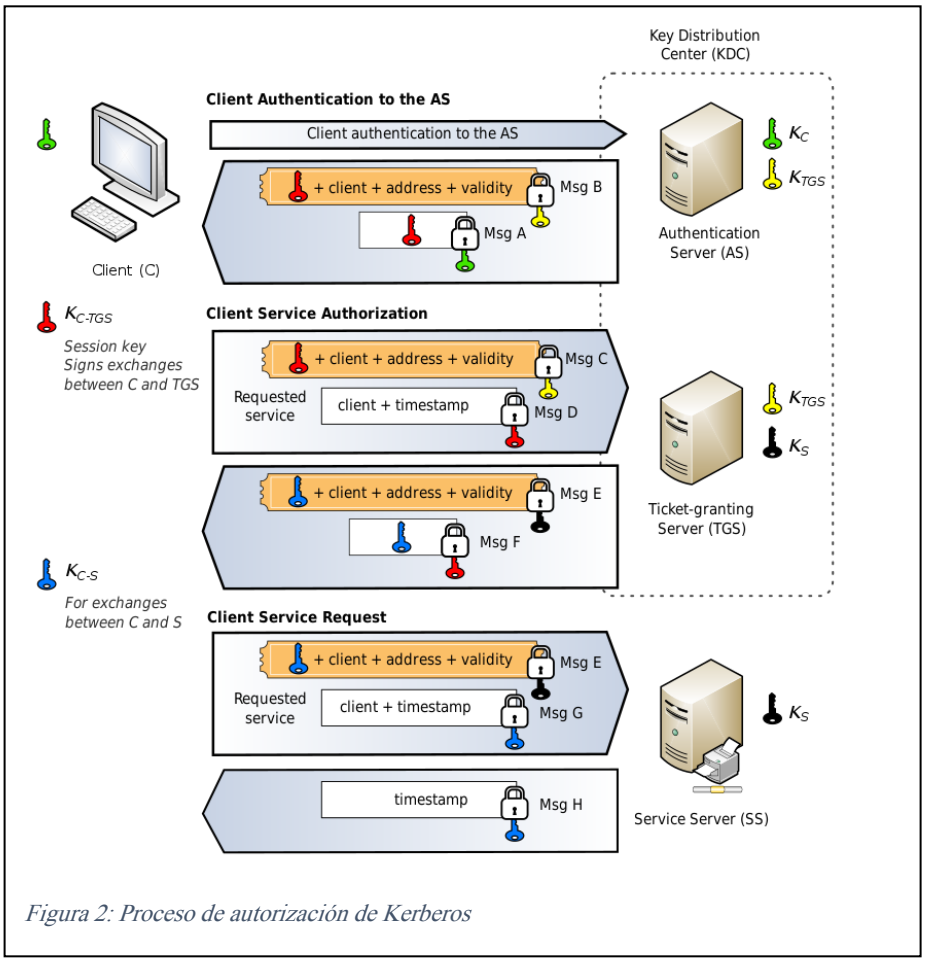

Figura 2: Proceso de autorización de Kerberos

5) Una vez que el cliente ha recibido los mensajes, descifra el mensaje A para obtener el client/TGS session key ***Kc-tgs***. Esta session key se usa para las posteriores comunicaciones con el TGS. (El cliente no puede descifrar el mensaje B pues para cifrar éste se ha usado la clave del TGS). En este momento el cliente ya se puede autenticar contra el TGS. Entonces el cliente envía los siguientes mensajes al TGS:
   a) Mensaje C: Compuesto del Ticket-Granting Ticket del mensaje B y el ID del servicio solicitado: ***servicio,{Kc-tgs , usuario, ip, vigencia}K_{tgs}***
   b) Mensaje D: Autenticador (compuesto por el ID de cliente y una marca de tiempo), cifrado usando el client/TGS session key. ***{usuario, timestamp}Kc-tgs***
6) Cuando recibe los mensajes anteriores, el TGS descifra el mensaje D (autenticador) usando el client/TGS session key y envía los siguientes mensajes al cliente:
   a) Mensaje E: Client-to-server ticket (que incluye Client/Server session key , el ID de cliente, la dirección de red del cliente y el período de validez) cifrado usando la clave secreta del servicio ***{Kc-s, usuario, ip, vigencia}K_s***.
   b) Mensaje F: Client/server session key cifrada usando el client/TGS session key. ***{Kc-s}$K_{c\text{-}tgs}$***
7) Cuando el cliente recibe los mensajes E y F, ya tiene suficiente información para autenticarse contra el SS. El cliente se conecta al SS y envía los siguientes mensajes:
   a) Mensaje E del paso anterior: ***{Kc-s, usuario, ip, vigencia}K_s***
   b) Mensaje G: un nuevo Autenticador que incluye el ID de cliente, una marca de tiempo y que está cifrado usando el client/server session key: ***{usuario, timestamp}Kc-s***
8) El SS descifra el ticket usando su propia clave secreta y envía el siguiente mensaje al cliente para confirmar su identidad:
   a) Mensaje H: la marca de tiempo encontrada en el último Autenticador recibido del cliente más uno, cifrado el client/server session key ***{usuario, timestamp+1}Kc-s***.

b) El cliente descifra la confirmación usando el client/server session key y chequea si la marca de tiempo está correctamente actualizada. Si esto es así, el cliente confiará en el servidor y podrá comenzar a usar el servicio que este ofrece.

9) El servidor provee del servicio al cliente.

4 Ventajas:

1. **Seguridad:** Utiliza técnicas criptográficas sólidas para proteger la autenticación y la comunicación entre los componentes. Cubre las tres funciones de seguridad y la contraseña nunca viaja por la red en el proceso de autenticación.
2. **Autenticación mutua**: Tanto los usuarios como los servicios se autentican entre sí, reduciendo el riesgo de ataques de suplantación.
3. **Menor exposición de contraseñas**: Los usuarios no envían contraseñas a cada servicio; en su lugar, utilizan tickets, lo que minimiza el riesgo de intercepción.
4. **Centralización:** Proporciona un único punto de autenticación para acceder a múltiples servicios en la red.
5. **Escalabilidad:** Puede manejar grandes cantidades de usuarios y servicios en una red distribuida, lo que lo hace adecuado para entornos empresariales grandes.
6. **Integración**: Es compatible con muchos sistemas operativos y aplicaciones, facilitando su implementación en diversas infraestructuras.
7. **Renovación de tickets:** Permite la renovación de tickets para evitar que los usuarios tengan que autenticarse repetidamente.

5 Desventajas:

1) **Complejidad:** Implementar y mantener un sistema Kerberos puede ser complejo debido a la necesidad de configuración y gestión de claves.
2) **Dependencia de la infraestructura:** Requiere servidores dedicados y una infraestructura de red robusta para funcionar correctamente.
3) **Punto único de fallo:** Si el servidor de autenticación (AS) o el servidor de tickets (TGS) fallan, los usuarios pueden perder el acceso a los recursos protegidos.
4) **Requisitos de tiempo sincronizado:** Los servidores en el sistema Kerberos deben tener sus relojes sincronizados para evitar problemas de autenticación.
    a) 5 minutos de tolerancia: una estación que tenga una diferencia mayor a 5 minutos (configurable) se le negarán todos los requerimientos.
    b) Uso obligatorio de ntp: Aunque no forma parte de la especificación de Kerberos, es un estándar *de facto* utilizar ntp (network time protocol) para sincronizar la hora de todas las computadoras.
5) **Dependencia de la red**: Si la red falla o hay problemas de conectividad, puede afectar la capacidad de autenticación.
6) **Administración:** No hay estándar de los protocolos de administración
7) **Uso intensivo de criptografía**: 5 llaves involucradas en el proceso de autenticación y autorización.

En resumen, Kerberos es un protocolo de autenticación sólido que proporciona seguridad en redes informáticas, pero su implementación puede ser compleja y requiere una infraestructura adecuada para funcionar de manera efectiva.

6 Referencias

Documentación oficial del proyecto: https://web.mit.edu/Kerberos/krb5-latest/doc/index.html

The Kerberos Network Authentication Service (V5): https://www.rfc-editor.org/rfc/rfc4120

Public Key Cryptography for Initial Authentication in Kerberos (PKINIT): https://www.rfc-editor.org/rfc/rfc4556

Kerberos (protocol): https://en.wikipedia.org/wiki/Kerberos_(protocol)

What ist Kerberos authentication and how does it work?: https://www.ionos.com/digitalguide/server/security/kerberos/