Tema 5, módulo 6: infraestructura *Kerberos*

Programación Segura
grado en ingeniería informática

Curso académico 2020-2021

Objetivos del módulo

- Presentar *Kerberos* como infraestructura para autenticación y autorización.
- Preparar un servicio *Kerberos* para emplearlo en ejemplos y actividades.

Índice

1. Sistema *Kerberos*

Autenticación

- Cualquier proceso dirigido a acreditar la identidad de un sujeto con objeto de evitar suplantaciones.
  - Es necesario verificar la identidad de un sujeto para autorizar su acceso a recursos protegidos.
- Debe incluir algún mecanismo que obligue al sujeto a proporcionar una evidencia que demuestre su identidad.
- Esa evidencia puede ser de diferente naturaleza:
  - Información que solo el sujeto conoce (contraseña)
  - Información que solo el sujeto puede producir (firma digital, huella dactilar, fondo de retina)
  - Información que solo el sujeto puede poseer (tique *Kerberos*)

Infraestructura *Kerberos*

- Es un sistema de autenticación desarrollado a comienzos de la década de 1980 en el *Massachusetts Institute of Technology* (MIT).
  - Componente de la plataforma educativa *Athena*.
- Su especificación es pública.
  - Versión actual es *Kerberos 5*.
- Existen diferentes implementaciones: MIT, Heimdal, Windows 20XX

Novedades que introdujo

Solución SSO completa: el usuario solo teclea su contraseña una vez durante la sesión (*Single Sign-On*).

Protección contra robos:

- Las contraseñas nunca viajan por la red.
- Cliente y servicio protegido verifican *de facto* la identidad uno del otro dificultando el robo de información.

Separación de datos de usuario:

- La base de datos de *Kerberos* solo almacena información necesaria para autenticación.
- No está preparada para almacenar datos como nombres reales de usuario, *usernames*, identificadores de grupo, etc.
- Eso refuerza la seguridad y facilita su integración en la infraestructura de red.

Características

- Es un servicio centralizado concebido como protección contra enemigo interno.
  - Es el sistema de firma única más corrientemente empleado en sistemas tipo UNIX (LINUX y SOLARIS™).
- Implementa un protocolo altamente seguro para autenticar usuarios, máquinas (*hosts*) y servicios en una red local.
- Incorpora infraestructura criptográfica para operar de forma segura en redes locales inseguras.
  - El tráfico por la red se cifra para anular escuchas y ataques de reinyección.
- Consigue autenticación mútua de usuarios y servicios.
  - En las comunicaciones se emplean claves de sesión que solo las dos partes involucradas pueden llegar a conseguir.
- Basado en el concepto de tercero de confianza.

Tercero de confianza

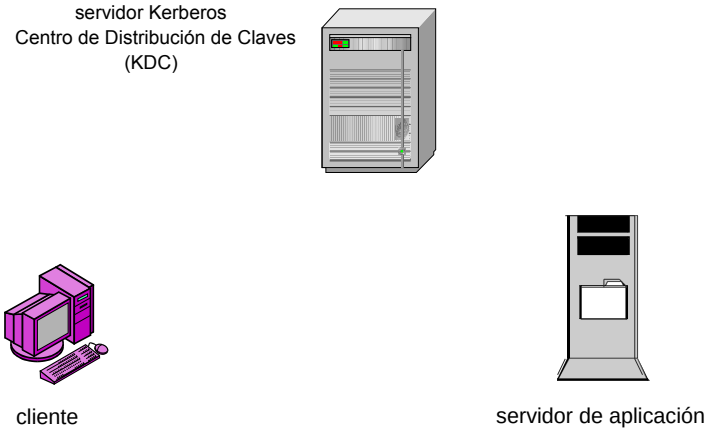
- *Kerberos* asigna una clave maestra a cada entidad a proteger en la red (cliente, máquina o aplicación de servicio a defender).
  - *Kerberos* mantiene una base de datos de claves maestras.
- Cada clave maestra es conocida únicamente por la entidad que la tiene asignada y por *Kerberos*.
  - El conocimiento de esa clave maestra sirve para probar la identidad tanto de la entidad como de *Kerberos*.
- *Kerberos* genera una clave de sesión distinta cada vez que dos entidades se ponen en comunicación.

Esquema de tercero de confianza[1]

- Alicia (A) inicia una comunicación con Bob (B).
- Ambas partes confían en una tercera entidad S.
  - K_{AS} es la clave maestra de A que solo conocen A y S.
  - K_{BS} es la clave maestra de B que solo conocen B y S.
  - Los mensajes hacia/desde S se encriptan usando esas claves maestras.
- Protocolo para autenticación entre pares:
  1. A envía un mensaje a S solicitando comunicarse con B.
  2. S genera una clave de sesión K_{AB}.
  3. S envía un mensaje cifrado a A que incluye la clave de sesión K_{AB} y la clave de sesión cifrada con K_{BS}: $\{K_{AB}, \{K_{AB}\}_{K_{BS}}\}_{K_{AS}}$
  4. A obtiene y guarda la clave K_{AB}; presenta la parte $\{K_{AB}\}_{K_{BS}}$ a B (clave de sesión K_{AB} cifrada con clave maestra K_{BS}).
  5. B obtiene la clave K_{AB}.
- Todos los mensajes intercambiados posteriormente dentro de esa sesión por A y B se cifran usando la clave K_{AB}.

KDC (***Key Distribution Center***)

- Parte de *Kerberos* que actúa como tercero de confianza.



Componentes de *KDC*

- *KDC* consta de dos componentes lógicas separadas:
  - **Servidor de autenticación (AS)**
    - Se encarga de acreditar la identidad de los clientes cuando estos hacen *login*.
    - Expide tiques de sesión denominados TGT.
  - **Servidor emisor de tiques (TGS)**
    - Se encarga de atender las peticiones de servicios dentro de una sesión de cliente (cliente posee ya un tique de sesión).
    - Expide tiques de servicio.

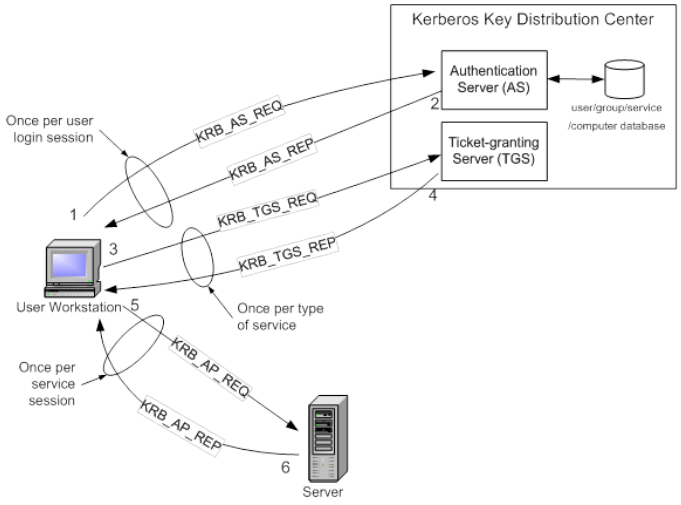
[1] Al estilo Needham-Schroeder

Tiques

- Son credenciales expedidas para que los clientes demuestren la autenticidad de su identidad ante los servicios de la red *kerberizada*.
- Van cifrados con la clave maestra del servicio al que se dirigen.
  - Cliente que presenta un tique es incapaz de conocer o cambiar su contenido.
- Entre otra información que portan:
  - Principal (nombre identificativo) del solicitante del servicio
  - Principal (nombre identificativo) del servicio solicitado
  - Momento de expedición
  - Validez temporal (por defecto, 10 horas)
  - Clave de sesión
- Su validez temporal es limitada para evitar abusos.
  - Un tique de servicio válido franquea al cliente para el que se emitió el acceso a un servicio sin que la componente *KDC* que lo expidió pueda ya impedirlo.

Esquema general de protocolo

- Un cliente registrado quiere emplear un servicio de la red *kerberizada*.



Pasos

1. Cliente se identifica positivamente mediante contraseña ante componente *AS*.
2. Componente *AS* proporciona al cliente:
   - Tique TGT (*Ticket-Granting Ticket*) que acredita al cliente ante componente *TGS*; cada tique TGT va cifrado con clave maestra de componente *TGS*; el tique TGT es válido hasta que el cliente haga *logout*.
   - Clave de sesión para diálogo cifrado entre cliente y componente *TGS*.

3. Durante la sesión, cada vez que cliente quiere hacer uso de un servicio protegido presenta el tique TGT de la sesión a componente *TGS*.

4. En cada ocasión, componente *TGS* proporciona:

   - Tique de servicio que autoriza al usuario a acceder al servicio requerido; va cifrado con clave maestra del servicio.
   - Clave de sesión para diálogo cifrado entre cliente y servicio.

5. Cliente presenta ese tique al servicio demostrando que tiene autorización para disfrutar del servicio hasta que expire.

Terminología específica

- **Realm**
  - Dominio administrativo dentro del cual un servidor *KDC* tiene autoridad para autenticar a un usuario o servicio.
  - Suele coincidir con el nombre del dominio DNS escrito en mayúsculas.
  - En nuestro caso:

    `LABOPROGSEGURA.UNAVARRA.ES`

- **Principal:**
  - Toda entidad (usuario/máquina/servicio) registrada en la base de datos de un reino.
  - Cada principal tiene una clave maestra propia.
  - Ejemplos de principales:

```
K/M@LABOPROGSEGURA.UNAVARRA.ES
root/admin@LABOPROGSEGURA.UNAVARRA.ES
kadmin/eim-alu-XXXXX.lab.unavarra.es@LABOPROGSEGURA.UNAVARRA.ES
krbtgt/LABOPROGSEGURA.UNAVARRA.ES@LABOPROGSEGURA.UNAVARRA.ES
mikaldaz@LABOPROGSEGURA.UNAVARRA.ES
```

Documentación sobre *Kerberos*

Oficial: http://web.mit.edu/kerberos/krb5-latest/doc/index.html

Experto: http://techpubs.spinlocksolutions.com/dklar/kerberos.html

2. Instalación

2.1. Instalación en servidor

0: Instalar servicio `ntp` en servidor y clientes

- Una parte de la seguridad que proporciona la infraestructura de *Kerberos* se basa en los cuños de tiempo (*timestamp*) con los que se marcan los tiques.
- En consecuencia resulta esencial que todas las máquinas estén razonablemente sincronizadas.

- Se puede emplear el protocolo *ntp* para asegurar que cada máquina esté ajustada a *UTC* (*Coordinated Universal Time*) con una resolución de décimas de milisegundo.
- Orden:

```
lunik:> sudo apt-get install ntp
```

1: Instalar paquetes para servidor KDC y servidor administrativo

- Orden:

```
lunik:> sudo apt-get install krb5-kdc krb5-admin-server
```

- Durante la instalación solicita el nombre del nuevo reino:

```
LABOPROGSEGURA.UNAVARRA.ES
```

- También solicita los nombres de las máquinas que van a operar como servidores.
  - Servidor administrativo (*Kerberos Master Server*): para tareas administrativas con la base de datos de principales; permite acceso remoto.
  - Servidor KDC: atiende peticiones de autenticación y emite tiques.
  - Se sugiere emplear *localhost* para ambos.

2: Crear un nuevo reino (*realm*)

- Lanza el *script* para creación de reinos:

```
lunik:> sudo krb5_newrealm
```

- Puede tardar unos minutos en generar material criptográfico para cifrado de la base de datos de principales de ese reino.
- Solicita la contraseña para principal K/M.
  - Principal K/M es la identidad del administrador de la base de datos de principales.
  - Su contraseña protege el acceso a la base de datos de principales y encripta su contenido.
  - Se almacena en fichero `/etc/krb5kdc/stash`

3: Modificar fichero `/etc/krb5.conf`

- Establece parámetros de configuración del servicio *KDC* instalado en la máquina y de los diferentes reinos que ese servicio puede administrar.
- Empleando un editor de texto:
  - En sección `[realms]` eliminar todos los reinos excepto el creado para la asignatura.

- En sección `[domain_realms]` añadir líneas

```
.lab.unavarra.es = LABOPROGSEGURA.UNAVARRA.ES
 lab.unavarra.es = LABOPROGSEGURA.UNAVARRA.ES
```

  y eliminar el resto.

4: Consultar fichero `/etc/krb5kdc/kdc.conf`

- Proporciona información de configuración de los diferentes reinos albergados en esta máquina.
- Para el reino `LABOPROGSEGURA.UNAVARRA.ES` muestra:
  - Ubicación de la base de datos de principales
  - Ubicación del fichero de administradores (*acl*)
  - Ubicación del fichero con la clave maestra que encripta la base de datos
  - Duración por defecto de un tique
  - *Suites* de cifrado disponibles

5: Definir políticas para contraseñas

- Arranca la herramienta local de administración:

```
lunik:> sudo kadmin.local
```

- Observa que por el momento no se solicita contraseña ninguna para ejercer de administrador.
- Tecleando `?` se obtiene un listado completo de órdenes de administración.
- Define cuatro políticas con condiciones a cumplir por las contraseñas:

```
kadmin.local: addpol -minlength 12 -minclasses 4 admin
kadmin.local: addpol -minlength 10 -minclasses 3 host
kadmin.local: addpol -minlength 10 -minclasses 3 service
kadmin.local: addpol -minlength  8 -minclasses 2 user
kadmin.local: quit
```

6: Crear administrador del reino (*administration principal*)

- Arranca la herramienta local de administración:

```
lunik:> sudo kadmin.local
```

- Introduce:

```
kadmin.local: addprinc -policy admin root/admin
kadmin.local: quit
```

- Solicita nombre y contraseña para el administrador principal.

- Nombre debe terminar en */admin* para distinguirlo de principales corrientes.
- Exige que la contraseña elegida cumpla con la política indicada.

7: Incluir nuevo administrador en fichero de control de acceso `/etc/krb5kdc/kadm5.acl`

- Empleando un editor de texto se añade la línea:

```
root/admin@LABOPROGSEGURA.UNAVARRA.ES  *
```

- Habilita al principal root/admin para realizar cualquier operación sobre ese reino.
- Permisos para acciones concretas:
  - *INQUIRE*
  - *ADD*
  - *MODIFY*
  - *DELETE*

8: Relanzar servicio de administración

- Para que las modificaciones realizadas en el fichero `/etc/krb5kdc/kadm5.acl` tengan efecto es necesario relanzar el servicio de administración.
- Orden:

```
lunik:> sudo /etc/init.d/krb5-admin-server restart
```

9: Activar demonios de servicio

- Órdenes:

```
lunik:> sudo krb5kdc
lunik:> sudo kadmind
```

  asociados respectivamente al servicio de distribución de claves (KDC) y al servicio administrativo.
- Si el instalador no lo ha hecho ya, se pueden añadir las correspondientes entradas a los ficheros `/etc/init.d` y `/etc/rcX.d` .
- Mediante las órdenes

```
lunik:> sudo /etc/init.d/krb5-kdc status
lunik:> sudo /etc/init.d/krb5-admin-server status
```

  se puede consultar en cualquier momento el estado del servicio correspondiente.

Herramientas administrativas

- `kadmin` y `kadmin.local` son la misma herramienta administrativa.
- `kadmin` está preparada para administración remota (por medio de *ssh*) por parte de un principal con permisos de administración.
- `kadmin.local` debe ejecutarse localmente en la misma máquina que el servidor administrativo.
  - Debe ejecutarse con permisos de administrador de sistema (permisos de *root*).
  - Se emplea mientras no existe un administrador principal.
  - Abre directamente la base de datos de *Kerberos* sin requerir los permisos de acceso extra que tiene el administrador principal.
- Ahora ya existe un administrador principal en el servicio *Kerberos* instalado por lo que puede emplearse `kadmin`.

Prueba: listado de principales

- Arranca la herramienta de administración:

```
lunik:> sudo kadmin
```

- Introduce:

```
kadmin: listprincs
kadmin: quit
```

10: Añadir nuevos principales

- Arranca la herramienta de administración:

```
lunik:> sudo kadmin
```

- Añade principal con perfil de usuario:

```
kadmin: addprinc -policy user mikaldaz
kadmin: quit
```

- Otras órdenes internas:

  getprinc: consulta información de un principal

  delprinc: elimina un principal

  listpols: lista políticas de contraseñas definidas

  getpol: consulta detalles sobre una política concreta

  delpol: elimina una política de contraseñas

  ?: listado completo de órdenes

Desinstalación

- Suele ser corriente que *Kerberos* opere deficientemente cuando su instalación no ha sido correctamente completada.
- En esos casos una buena solución es volver a instalar habiendo eliminado previamente todo elemento de la instalación defectuosa.
- Para volver a instalar *Kerberos* desde cero es necesario purgar la instalación defectuosa y además eliminar un directorio de datos de aplicación:

```
lunik:> sudo apt-get purge krb5-kdc krb5-admin-server
        krb5-config krb5-locales krb5-user
lunik:> sudo rm -r /var/lib/krb5kdc
```

2.2. Instalación en cliente

1: Instalación de paquetes de usuario

- Orden con `sudo`:

```
lunik:> sudo apt-get install krb5-user libpam-krb5 libpam-ccreds
```

2: Configurar cliente

- Orden con `sudo`:

```
lunik:> sudo dpkg-reconfigure krb5-config
```

- Solicita nombre del reino y nombre de los servidores.

3: Probar configuración

- Orden:

```
lunik:> kinit kuser@LABOPROGSEGURA.UNAVARRA.ES
```

4: Examinar tique

- Orden:

```
lunik:> klist
```

5: Cambiar contraseña de usuario

- Orden:

```
lunik:> kpasswd
```

2.3. Ejemplo desde Java

Ejemplo: *JaasAcn.java*

- Sin *SecurityManager*: ejecutar con las opciones de *VM*

```
-Djava.security.auth.login.config=./etc/login.conf
-Djava.security.krb5.realm=LABOPROGSEGURA.UNAVARRA.ES
-Djava.security.krb5.kdc=IP servidor kdc
```

- Con *SecurityManager*: ejecutar con las opciones de *VM*

```
-Djava.security.manager
-Djava.security.policy=jaasacn.policy
-Djava.security.auth.login.config=./etc/login.conf
-Djava.security.krb5.realm=LABOPROGSEGURA.UNAVARRA.ES
-Djava.security.krb5.kdc=IP servidor kdc
```