PROTOCOLO KERBEROS
(Administración Avanzada de Sistemas Operativos. Grado en Ingeniería Informática. Facultad de Informática. Universidad de Murcia. Curso 2011/12).

PARTICIPANTES:

- Un servidor Kerberos, o *KDC* ("Key Distribution Center"): máquina Linux, controlador de dominio de Active Directory, Novell KDC, etc **(1)**. Debe ser una máquina especialmente protegida (poner especial cuidado con el cortafuegos), ya que el resto de participantes del protocolo confían "ciegamente" en ella.

- Uno o varios *clientes Kerberos*: equipos Linux o Windows, desde los que el usuario se autentica, ya sea con el ***login*** de PAM (desde el terminal gráfico, desde cualquiera de los terminales de texto, con "su", etc), o con un programa especial de Kerberos que en Linux se denomina ***kinit***.

- Uno o varios *servicios kerberizados*: PAM, SSH, FTP, HTTP, NFSv4, etc. En general, cualquier servicio servidor que requiera, en algún momento, la autenticación de un usuario desde un cliente es susceptible de ser "kerberizado".

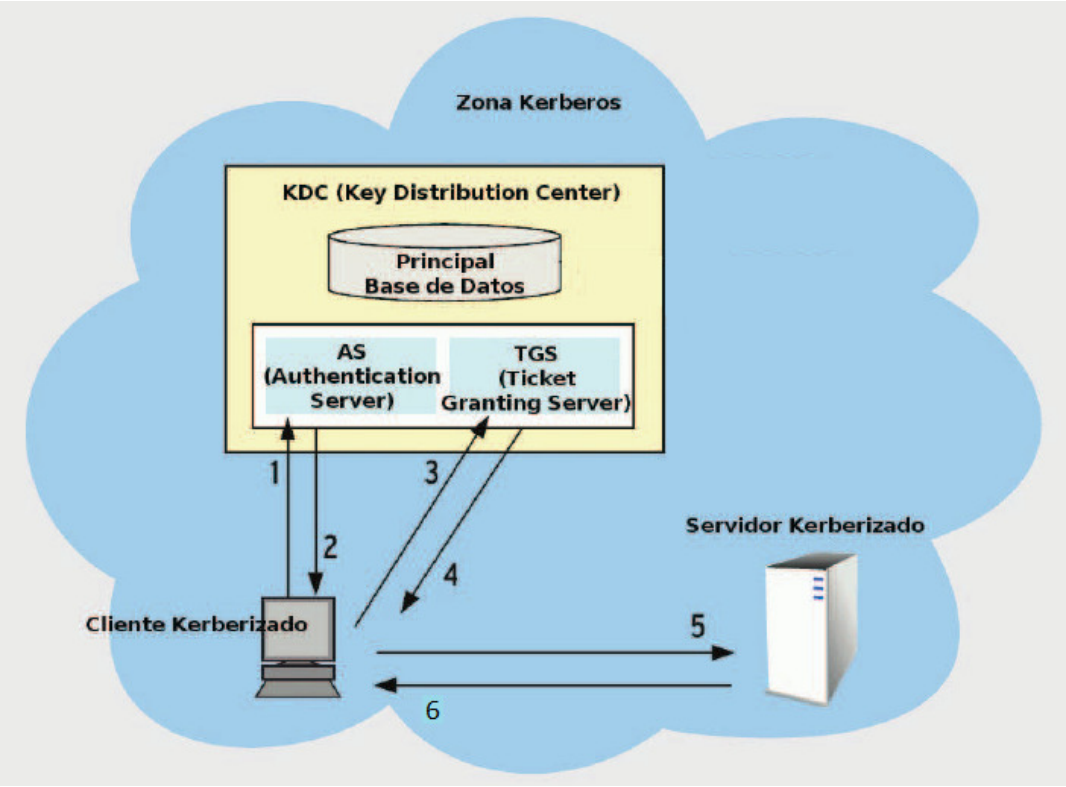

Fig. 1: Secuencia de mensajes de la autenticación Kerberos

(1)

En las prácticas el KDC será una máquina Linux, pero también sería posible que clientes Kerberos se autenticaran contra un KDC que no fuese Linux.

DESCRIPCIÓN DEL PROTOCOLO:

- Fase 2 (empezamos por el final):

    - Contexto:

      El cliente ya se ha autenticado (ha introducido su password, y lo ha hecho una única vez) contra una base de datos de usuarios "Principal", que reside en el KDC (y que no tiene nada que ver con los usuarios del /etc/passwd).

      Como resultado de la anterior autenticación (que veremos más adelante), el cliente cuenta, durante un período de tiempo virtualmente ilimitado (o hasta que apague su equipo), con:

      a) Una clave secreta, *Kcli*, que es compartida por el cliente y por el KDC.

      b) Un registro en memoria, denominado TGT ("***Ticket Granting Ticket***"), que está cifrado con una clave secreta, *Ktgt*, que no conoce el cliente, pero sí el KDC. Este TGT es distinto para cada posible cliente, y almacena:

         ➔ Los datos de autenticación del usuario **(2)**.
         ➔ Un período de validez (normalmente, del orden de una semana).
         ➔ La clave *Kcli* anteriormente aludida.

      Además, el servidor del servicio kerberizado también tiene (por cada uno de sus servicios kerberizados, si hubiere varios):

      a) Una clave secreta, *Ksk*, que es compartida por el servicio kerberizado y por el KDC.

      En esta ocasión, *Ksk* es permanente, y se guarda en el fichero del servidor /etc/krb5.keytab. Este fichero está, por supuesto, encriptado (de hecho, no estaría de más aplicarle un "chmod 0700 /etc/krb5.keytab"), y es generado sólo una vez desde la herramienta administrativa de Kerberos (***kadmin*** o ***kadmin.local***), mediante el comando *ktadd*.

(2)

Entre ellos figura el nombre de usuario, pero no su contraseña.

- Protocolo de la fase 2 (en la ilustración, pasos 3, 4, 5 y 6):

  a) El usuario, desde el host cliente, solicita al KDC una acreditación para presentársela al servicio kerberizado (paso 3). Esta solicitud contiene los tres siguientes mensajes.

     ➔ Mensaje A1. Un código que indica el servicio kerberizado ante el que el usuario se quiere autenticar, sin cifrar.
     ➔ Mensaje A2. El TGT, cifrado con la clave *Ktgt*, que el host cliente no conoce, pero sí el KDC.
     ➔ Mensaje A3. Los datos del usuario y una marca de tiempo asociada al instante actual, cifrados ambos con la clave *Kcli*, que conocen tanto cliente como KDC.

  b) El KDC recibe la solicitud anterior. Su proceso expendedor de tickets para servicios kerberizados, denominado TGS ("Ticket Granting Server"), empieza a trabajar partiendo de esta solicitud. En concreto, el TGS descifra el TGT que venía en A2 (mediante *Ktgt*) y comprueba si los datos de usuario que vienen en él coinciden con los datos de usuario que ha aportado el cliente en el mensaje A3 (este mensaje lo puede descifrar el KDC gracias a que en uno de los campos del TGT venía *Kcli*). También se lee la marca de tiempo de A3, para comprobar que entre el host cliente y el KDC no existe un desfase temporal superior a 5'. Si la comprobación anterior de los datos de usuario que vienen en A2 y A3 es satisfactoria, y si la sincronización entre cliente y KDC también lo es, el proceso TGS del KDC otorgará al usuario un "salvoconducto" para autenticarse frente al servicio kerberizado, bajo la forma de los dos siguientes mensajes (paso 4):

     ➔ Mensaje B1. Un ticket de servicio para el servicio kerberizado contra el que se quiere autenticar el usuario. Este ticket ha sido cifrado con *Ksk* (la clave compartida entre KDC y el servicio kerberizado), y contiene:

        1. Los datos del usuario.
        2. La IP del host cliente.
        3. El período de validez del ticket (normalmente, del orden de 1 día).
        4. Una nueva clave de cifrado, *Kcli2*, que se utilizará en el paso siguiente, para cifrar las comunicaciones entre el host cliente y el servidor del servicio kerberizado.

     ➔ Mensaje B2. La nueva clave de cifrado *Kcli2* (ver B1), cifrada con *Kcli* (con ello se pretende que el cliente pueda conocer *Kcli2*).

c) El cliente recibe B1 y B2. Con este último obtiene *Kcli2*. Por último, envía al servidor del servicio kerberizado los dos mensajes siguientes (paso 5):

   ➔ Mensaje C1. El ticket de servicio, es decir, el B1 que le llegó del KDC, sin cambio alguno.
   ➔ Mensaje C2. Los datos de usuario y una marca de tiempo, cifrados con *Kcli2*.

d) El servicio kerberizado recibe C1 (el ticket de servicio) y C2 (los datos de usuario). C1 lo desencripta con *Ksk*, y obtiene así, entre otras cosas, los datos de usuario y la otra clave, *Kcli2*, que necesitará para descifrar C2. A continuación, ya con *Kcli2*, el servicio kerberizado descifra C2, con lo que obtiene los datos de usuario, que deberán coincidir con los del ticket de servicio (C1). También en este caso se comprueba la marca de tiempo que viene en C2, para asegurar que quien solicita la autenticación la ha pedido en los últimos 5'. Una vez ejecutados estos pasos, el servicio kerberizado sabrá que el cliente que se dirige a él es, en efecto, quien dice ser (otro asunto serán los permisos de utilización del servicio por parte del cliente, pero la autenticación, que es la parte de la que se responsabiliza Kerberos, ya se ha completado).

   Por último, el servicio kerberizado envía un acuse de recibo al cliente, bajo la forma de un mensaje cifrado con *Kcli2* que contiene la marca de tiempo que venía en C2, pero incrementada en 1. Esto le sirve al cliente para estar también él seguro de que el servicio kerberizado que se dispone a utilizar es, efectivamente, el que él esperaba (paso 6).

- Fase 1:

  - Protocolo de la fase 1 (en la ilustración, pasos 1 y 2):

    a) El usuario introduce su contraseña Kerberos (en el caso del servicio kerberizado PAM, mediante el login; en otros casos, normalmente mediante ***Kinit***), y a continuación, el cliente "fabrica" una clave secreta, *Kinic*, que es el valor devuelto por una función hash unidireccional **(3)** conocida por los clientes y por el KDC, aplicada a la contraseña Kerberos del usuario. Con esta clave secreta (*Kinic*), el cliente cifra y envía al KDC un mensaje con una marca de tiempo, y envía también un segundo mensaje, esta vez como texto en claro, con el nombre de usuario (paso 1).

(3)

El hecho de que la función hash sea unidireccional significa que, una vez aplicada, no existe modo alguno de obtener una función inversa que nos devuelva el contenido original.

b) El proceso AS ("Authentication Server") del KDC recibe el mensaje con el nombre de usuario y busca en la base de datos su contraseña Kerberos. Seguidamente, obtiene *Kinic* aplicando la misma función hash que ha utilizado el cliente, y, ya con *Kinic*, descifra el mensaje de la marca de tiempo. Si comprueba que dicha marca de tiempo no tiene un desfase superior a 5' con el instante actual, el KDC construye un TGT (con los campos ya conocidos, es decir, los datos del usuario autenticado, un período de validez y la clave *Kcli*) cifrado con *Ktgt*, y crea también otro mensaje, cifrado con *Kinic*, cuyo contenido es la clave *Kcli*, de modo que a partir de ahora entre cliente y KDC sólo se utilizará *Kcli*, en vez de *Kinic* (paso 2).

CONSIDERACIONES FINALES:

Nótese que, con este modo de proceder, se ha logrado un nivel de seguridad bastante alto, por lo siguiente:

1. Se ha evitado la necesidad de transmitir el password a través de la red.

2. Se ha empleado como clave inicial (*Kinic*) un secreto compartido que está en función del instante en que el usuario introduce en el host cliente su contraseña, y que por tanto no es reutilizable en sucesivas autenticaciones. El resto de comunicaciones se basa en cifrados simétricos cuyas claves compartidas están suficientemente protegidas.

3. Se ha implementado un sistema de autenticación "Single Sign-On", que le permite al usuario teclear su contraseña una única vez (al abrir un terminal, si PAM está kerberizado, o, en todo caso, al ejecutar ***Knit***), evitando tener que volver a introducirla cada vez que deba autenticarse ante un servicio kerberizado. En definitiva, los pasos 1 y 2 de la figura 1 se ejecutan una sola vez, y una vez que el host cliente tiene en memoria el TGT del usuario que se ha autenticado en él **(4)**, ya no se le exige volver a introducir el password cada vez que éste quiera utilizar un servicio (pasos del 3 al 6).

(4)

O los TGT de los usuarios, si se han autenticado en ese mismo host varios usuarios.