Descripción general de Kerberos: Servicio de autenticación para sistemas de red abierta

Contenido

Introducción

En un entorno de computación de red abierta, una estación de trabajo no es confiable para identificar correctamente a sus usuarios en los servicios de red. Kerberos proporciona un enfoque alternativo por el que se utiliza un servicio confiable de autenticación de terceros para verificar las identidades de los usuarios. Este documento ofrece una descripción del modelo de autenticación Kerberos según se implementó para el proyecto Athena MIT. Describe los protocolos usados por los clientes, los servidores y Kerberos para alcanzar la autenticación. También describe la administración y replicación de la base de datos requerida. Se describen las vistas de Kerberos según las ve el usuario, el programador y el administrador. Finalmente, se da el papel de Kerberos en una descripción de Athena más detallada, junto con una lista de aplicaciones que utiliza actualmente Kerberos para la autenticación de usuarios. Describimos la incorporación de la autenticación de Kerberos al Sistema de archivo de red de Sun como caso práctico para integrar Kerberos en una aplicación existente.

Autores de Kerberos

- Jennifer G. Steiner, Project Athena, Massachusetts Institute of Technology, Cambridge, MA 02139, steiner@ATHENA.MIT.EDU
- Clifford Neuman, Departamento de Informática, FR-35, Universidad de Washington, Seattle, WA 98195, bcn@CS.WASHINGTON.EDU. Clifford Neuman fue miembro del personal del Proyecto Athena durante la fase de diseño e implementación inicial de Kerberos.
- Jeffrey I. Schiller, Project Athena, Massachusetts Institute of Technology, Cambridge, MA 02139, jis@ATHENA.MIT.EDU

Introducción a Kerberos

Este artículo ofrece una visión general de Kerberos, un sistema de autenticación diseñado por Miller y Neuman. para entornos de informática de red abierta y describe nuestra experiencia en el proyecto Athena del MIT. En la sección sobre Motivación, explicamos por qué se necesita un nuevo modelo de autenticación para las redes abiertas y cuáles son sus requisitos. ¿Qué es Kerberos? muestra los componentes del software Kerberos y describe cómo interactúan al proporcionar el servicio de autenticación. En la sección Nombres Kerberos, describimos el esquema de nombres Kerberos.

Cómo funciona Kerberos presenta los bloques de creación de la autenticación Kerberos - el ticket y el autenticador. Esto lleva a una discusión de los dos protocolos de autenticación: la autenticación inicial de un usuario a Kerberos (análoga a la conexión) y el protocolo para la autenticación mutua de un consumidor potencial y un productor potencial de un servicio de red.

Kerberos requiere una base de datos de información sobre sus clientes; la sección Base de datos Kerberos describe la base de datos, su administración y el protocolo para su modificación. La sección Kerberos desde el exterior describe la interfaz Kerberos para sus usuarios, programadores de aplicaciones y administradores. En la sección La imagen más grande, describimos cómo encaja el Proyecto Athena Kerberos en el resto del entorno Athena. También describimos la interacción de diferentes dominios o rangos de autenticación Kerberos; en nuestro caso, la relación entre el Proyecto Athena Kerberos y Kerberos que se ejecuta en el Laboratorio de Informática del MIT.

En la sección Temas y Problemas Abiertos, mencionamos temas y problemas abiertos aún sin resolver. La última sección muestra el estado actual de Kerberos en el Proyecto Athena. En el Apéndice, describimos en detalle cómo se aplica Kerberos a un servicio de archivos de red para autenticar a los usuarios que desean obtener acceso a sistemas de archivos remotos.

Conceptos Kerberos

A lo largo de este trabajo utilizamos términos que pueden ser ambiguos, nuevos para el lector o usados de manera diferente en otros lugares. A continuación, indicamos el uso de esos términos.

Usuario, Cliente, Servidor: por usuario, nos referimos a un ser humano que utiliza un programa o servicio. Un cliente también utiliza algo, pero no es necesariamente una persona; puede ser un programa. A menudo, las aplicaciones de red constan de dos partes: un programa que se ejecuta en un equipo y solicita un servicio remoto, y otro que se ejecuta en el equipo remoto y realiza ese servicio. Los llamamos el lado cliente y el lado servidor de la aplicación, respectivamente. A menudo, un cliente se pondrá en contacto con un servidor en nombre de un usuario.

Cada entidad que utiliza el sistema Kerberos, ya sea un usuario o un servidor de red, es en cierto sentido un cliente, ya que utiliza el servicio Kerberos. Para distinguir los clientes Kerberos de los clientes de otros servicios, utilizamos el término principal para indicar tal entidad. Tenga en cuenta que una entidad de seguridad Kerberos puede ser un usuario o un servidor. (Describimos el nombre de los principales Kerberos en una sección posterior.)

Servicio frente a Servidor: utilizamos el servicio como especificación abstracta de algunas acciones que se deben realizar. Un proceso que realiza esas acciones se denomina servidor. En un momento dado, puede haber varios servidores (normalmente ejecutándose en máquinas diferentes) que realizan un servicio determinado. Por ejemplo, en Athena hay un servidor de inicio de sesión BSD UNIX ejecutándose en cada una de nuestras máquinas de intercambio de horas.

Key (Clave privada), Private Key (Clave privada), Password (Contraseña): Kerberos utiliza cifrado de clave privada. A cada entidad principal Kerberos se le asigna un número grande, su clave privada, conocida sólo por ese principal y Kerberos. En el caso de un usuario, la clave privada es el resultado de una función unidireccional aplicada a la contraseña del usuario. Utilizamos la clave como método abreviado para la clave privada.

Credenciales: desafortunadamente, esta palabra tiene un significado especial tanto para el sistema de archivos de red Sun como para el sistema Kerberos. Expresamos explícitamente si nos referimos a credenciales NFS o a credenciales Kerberos, de lo contrario el término se utiliza en el sentido normal del idioma inglés.

Maestro y esclavo: es posible ejecutar el software de autenticación Kerberos en más de una máquina. Sin embargo, siempre hay sólo una copia definitiva de la base de datos Kerberos. La máquina que aloja esta base de datos se denomina máquina maestra, o simplemente el maestro. Otras máquinas pueden poseer copias de sólo lectura de la base de datos Kerberos, que se denominan esclavos.

La motivación detrás de Kerberos

En un entorno informático personal no conectado a la red, los recursos y la información se pueden proteger protegiendo físicamente el ordenador personal. En un entorno informático de uso compartido de horas, el sistema operativo protege a los usuarios unos de otros y controla los

recursos. Para determinar lo que cada usuario puede leer o modificar, es necesario que el sistema de intercambio de horas identifique a cada usuario. Esto se logra cuando el usuario inicia sesión.

En una red de usuarios que requieren servicios desde muchos ordenadores independientes, hay tres enfoques que se pueden adoptar para controlar el acceso: No se puede hacer nada, ya que se confía en el equipo en el que el usuario ha iniciado sesión para evitar el acceso no autorizado; se puede requerir que el host demuestre su identidad, pero confiar en la palabra del host sobre quién es el usuario; o se puede exigir al usuario que demuestre su identidad para cada servicio requerido.

En un entorno cerrado donde todas las máquinas están bajo estricto control, se puede utilizar el primer enfoque. Cuando la organización controla todos los hosts que se comunican a través de la red, este es un enfoque razonable.

En un entorno más abierto, uno podría confiar selectivamente sólo en aquellos hosts bajo control organizacional. En este caso, se debe exigir a cada host que demuestre su identidad. Los programas rlogin y rsh utilizan este enfoque. En esos protocolos, la autenticación se realiza comprobando la dirección de Internet desde la que se ha establecido una conexión.

En el entorno de Athena, debemos ser capaces de cumplir las solicitudes de los anfitriones que no están bajo control organizacional. Los usuarios tienen el control total de sus estaciones de trabajo: pueden reiniciarlas, ponerlas en funcionamiento de forma independiente o incluso arrancarlas de sus propias cintas. Como tal, debe adoptarse el tercer enfoque; el usuario debe probar su identidad para cada servicio deseado. El servidor también debe probar su identidad. No es suficiente con asegurar físicamente el host que ejecuta un servidor de red; es posible que alguien de cualquier parte de la red se esté disfrazando de servidor dado.

Nuestro entorno impone varios requisitos a un mecanismo de identificación. Primero, debe ser seguro. Evitar debe ser lo suficientemente difícil como para que un atacante potencial no encuentre el mecanismo de autenticación como el eslabón débil. Alguien que observa la red no debe poder obtener la información necesaria para hacerse pasar por otro usuario. En segundo lugar, debe ser fiable. El acceso a muchos servicios dependerá del servicio de autenticación. Si no es fiable, el sistema de servicios en su conjunto no lo será. En tercer lugar, debe ser transparente. Lo ideal es que el usuario no tenga en cuenta la autenticación que se está produciendo. Por último, debería ser escalable. Muchos sistemas pueden comunicarse con los hosts Athena. No todos ellos respaldarán nuestro mecanismo, pero el software no debería quebrar si lo hicieran.

Kerberos es el resultado de nuestro trabajo para satisfacer los requisitos anteriores. Cuando un usuario accede a una estación de trabajo inicia sesión. Según el usuario, esta identificación inicial es suficiente para demostrar su identidad a todos los servidores de red requeridos durante la sesión de inicio de sesión. La seguridad de Kerberos depende de la seguridad de varios servidores de autenticación, pero no del sistema desde el que los usuarios inician sesión, ni de la seguridad de los servidores finales que se utilizarán. El servidor de autenticación proporciona a un usuario autenticado correctamente una manera de probar su identidad a servidores dispersos por la red.

La autenticación es un pilar fundamental para un entorno de red seguro. Si, por ejemplo, un servidor conoce con certeza la identidad de un cliente, puede decidir si proporciona el servicio, si se le deben otorgar privilegios especiales al usuario, quién debe recibir la factura del servicio, etc. En otras palabras, los esquemas de autorización y contabilización pueden construirse sobre la autenticación que proporciona Kerberos, lo que da como resultado una seguridad equivalente al

equipo personal solitario o al sistema de tiempo compartido.

¿Qué es Kerberos?

Kerberos es un servicio de autenticación de terceros de confianza basado en el modelo presentado por Needham y Schroeder. Se confía en que cada uno de sus clientes considere que el juicio de Kerberos sobre la identidad de cada uno de sus clientes es preciso. Las marcas de tiempo (números grandes que representan la fecha y hora actuales) se han agregado al modelo original para ayudar en la detección de la repetición. La repetición se produce cuando se roba un mensaje de la red y se envía más tarde. Para obtener una descripción más completa de la repetición y otros problemas de autenticación, vea Voydock y Kent.

¿Qué hace Kerberos?

Kerberos mantiene una base de datos de sus clientes y sus claves privadas. La clave privada es un gran número conocido sólo por Kerberos y el cliente al que pertenece. En el caso de que el cliente sea un usuario, se trata de una contraseña cifrada. Los servicios de red que requieren autenticación se registran con Kerberos, al igual que los clientes que desean utilizar esos servicios. Las claves privadas se negocian en el momento del registro.

Dado que Kerberos conoce estas claves privadas, puede crear mensajes que convenzan a un cliente de que otro es realmente quien afirma ser. Kerberos también genera claves privadas temporales, llamadas claves de sesión, que se entregan a dos clientes y a nadie más. Se puede utilizar una clave de sesión para cifrar mensajes entre dos partes.

Kerberos proporciona tres niveles distintos de protección. El programador de aplicaciones determina cuál es apropiado, según los requisitos de la aplicación. Por ejemplo, algunas aplicaciones sólo requieren que se establezca la autenticidad al inicio de una conexión de red, y pueden suponer que otros mensajes de una dirección de red dada se originan de la parte autenticada. Nuestro sistema de archivos de red autenticado utiliza este nivel de seguridad.

Otras aplicaciones requieren la autenticación de cada mensaje, pero no importa si el contenido del mensaje se revela o no. Para estos, Kerberos proporciona mensajes seguros. Sin embargo, los mensajes privados proporcionan un mayor nivel de seguridad, donde cada mensaje no solo se autentica, sino que también se cifra. Los mensajes privados son utilizados, por ejemplo, por el propio servidor Kerberos para enviar contraseñas a través de la red.

Componentes del software Kerberos

La implementación de Athena consta de varios módulos:

- biblioteca de aplicaciones Kerberos
- biblioteca de encripción
- biblioteca de base de datos
- programas de administración de bases de datos
- servidor de administración
- servidor de autenticación
- software de propagación de la base de datos
- programas de usuario
- aplicaciones

La biblioteca de aplicaciones Kerberos proporciona una interfaz para clientes de aplicaciones y servidores de aplicaciones. Contiene, entre otras cosas, rutinas para crear o leer solicitudes de autenticación y las rutinas para crear mensajes seguros o privados.

El cifrado en Kerberos se basa en DES, el estándar de cifrado de datos. La biblioteca de cifrado implementa esas rutinas. Se proporcionan varios métodos de cifrado, con ventajas y desventajas entre velocidad y seguridad. También se proporciona una extensión al modo de encadenamiento de bloques de cifrado DES (CBC), denominado modo CBC de propagación. En CBC, un error se propaga solamente a través del bloque actual del cifrado, mientras que en PCBC, el error se propaga a través del mensaje. Esto hace que todo el mensaje sea inútil si se produce un error, en lugar de sólo una parte de él. La biblioteca de cifrado es un módulo independiente y se puede reemplazar con otras implementaciones DES o con una biblioteca de cifrado diferente.

Otro módulo reemplazable es el sistema de administración de bases de datos. La implementación actual de Athena de la biblioteca de base de datos utiliza ndbm, aunque Ingres se utilizó originalmente. También podrían utilizarse otras bibliotecas de gestión de bases de datos.

Las necesidades de la base de datos Kerberos son sencillas; se guarda un registro para cada entidad principal, que contiene el nombre, la clave privada y la fecha de vencimiento del principal, junto con alguna información administrativa. (La fecha de vencimiento es la fecha a partir de la cual una entrada ya no es válida. Suele fijarse en unos años para el futuro en el momento de la inscripción).

Otro servidor, el servidor de nombres Hesiod, guarda otra información de usuario, como el nombre real, el número de teléfono, etc. De esta manera, Kerberos puede manejar información confidencial, es decir, contraseñas, utilizando medidas de seguridad bastante altas; mientras que la información no sensible que lleva Hesiod se trata de manera diferente; puede, por ejemplo, enviarse sin cifrar a través de la red.

Los servidores Kerberos utilizan la biblioteca de bases de datos, al igual que las herramientas para administrar la base de datos.

El servidor de administración (o servidor KDBM) proporciona una interfaz de red de lectura y escritura a la base de datos. El lado cliente del programa puede ejecutarse en cualquier equipo de la red. Sin embargo, el lado del servidor debe ejecutarse en el equipo que contiene la base de datos Kerberos para realizar cambios en la base de datos.

El servidor de autenticación (o servidor Kerberos), por otra parte, realiza operaciones de sólo lectura en la base de datos Kerberos, a saber, la autenticación de los principales y la generación de claves de sesión. Dado que este servidor no modifica la base de datos Kerberos, puede ejecutarse en un equipo que contenga una copia de sólo lectura de la base de datos Kerberos maestra.

El software de propagación de base de datos administra la replicación de la base de datos Kerberos. Es posible tener copias de la base de datos en varias máquinas diferentes, con una copia del servidor de autenticación ejecutándose en cada máquina. Cada una de estas máquinas esclavas recibe una actualización de la base de datos Kerberos del equipo maestro a intervalos determinados.

Por último, hay programas de usuario final para iniciar sesión en Kerberos, cambiar una contraseña Kerberos y mostrar o destruir entradas Kerberos (las notificaciones se explican más adelante).

Nombres de Kerberos

Parte de la autenticación de una entidad es nombrarla. El proceso de autenticación es la verificación de que el cliente es el designado en una solicitud. ¿En qué consiste un nombre? En Kerberos, se nombran tanto usuarios como servidores. En lo que respecta al servidor de autenticación, son equivalentes. Un nombre consta de un nombre principal, una instancia y un rango, expresados como name.instance@realm.

El nombre principal es el nombre del usuario o el servicio. La instancia se utiliza para distinguir entre las variaciones del nombre principal. Para los usuarios, una instancia puede conllevar privilegios especiales, como las instancias "root" o "admin". Para los servicios en el entorno Athena, la instancia es generalmente el nombre de la máquina en la que se ejecuta el servidor. Por ejemplo, el servicio rlogin tiene diferentes instancias en diferentes hosts: rlogin.priam es el servidor rlogin en el host denominado priam. Un ticket Kerberos sólo es bueno para un único servidor con nombre. Como tal, se necesita un billete separado para acceder a diferentes instancias del mismo servicio. El rango es el nombre de una entidad administrativa que mantiene los datos de autenticación. Por ejemplo, diferentes instituciones pueden tener cada una su propio equipo Kerberos, que contiene una base de datos diferente. Tienen diferentes rangos Kerberos. (Los rangos se discuten más a fondo en Interacción con otro Kerberi.)

Cómo funciona Kerberos

Esta sección describe los protocolos de autenticación Kerberos. Como se mencionó anteriormente, el modelo de autenticación Kerberos se basa en el protocolo de distribución de claves Needham y Schroeder. Cuando un usuario solicita un servicio, se debe establecer su identidad. Para ello, se presenta un billete al servidor, junto con la prueba de que el billete se entregó originalmente al usuario, no se robó. Hay tres fases para la autenticación a través de Kerberos. En la primera fase, el usuario obtiene las credenciales que se utilizarán para solicitar acceso a otros servicios. En la segunda fase, el usuario solicita autenticación para un servicio específico. En la fase final, el usuario presenta esas credenciales al servidor final.

Credenciales de Kerberos

Hay dos tipos de credenciales utilizadas en el modelo de autenticación Kerberos: entradas y autenticadores. Ambos se basan en el cifrado de clave privada, pero se cifran utilizando claves diferentes. Un ticket se utiliza para pasar de forma segura la identidad de la persona a la que se emitió el ticket entre el servidor de autenticación y el servidor final. El billete también pasa información que puede utilizarse para asegurarse de que la persona que lo utiliza sea la misma persona a la que se le expidió. El autenticador contiene la información adicional que, cuando se compara con la del billete, demuestra que el cliente que presenta el billete es el mismo al que se emitió el billete.

Un ticket es bueno para un único servidor y un único cliente. Contiene el nombre del servidor, el nombre del cliente, la dirección de Internet del cliente, una marca de tiempo, una duración y una clave de sesión aleatoria. Esta información se cifra utilizando la clave del servidor para el que se utilizará el ticket. Una vez emitido el ticket, el cliente designado puede utilizarlo varias veces para obtener acceso al servidor designado, hasta que caduque el ticket. Tenga en cuenta que, dado que el ticket está cifrado en la clave del servidor, es seguro permitir que el usuario pase el ticket al servidor sin tener que preocuparse de que el usuario modifique el ticket.

A diferencia del ticket, el autenticador sólo se puede utilizar una vez. Se debe generar uno nuevo

cada vez que un cliente desea utilizar un servicio. Esto no presenta un problema porque el cliente puede construir el autenticador en sí. Un autenticador contiene el nombre del cliente, la dirección IP de la estación de trabajo y la hora actual de la estación de trabajo. El autenticador se cifra en la clave de sesión que forma parte del ticket.

Obtener la notificación Kerberos inicial

Cuando el usuario se acerca a una estación de trabajo, sólo un fragmento de información puede probar su identidad: la contraseña del usuario. El intercambio inicial con el servidor de autenticación está diseñado para minimizar la posibilidad de que la contraseña se vea comprometida, al mismo tiempo que no se permite que un usuario se autentique correctamente sin conocer la contraseña. El proceso de inicio de sesión parece que el usuario es el mismo que iniciar sesión en un sistema de intercambio de horas. Sin embargo, tras bambalinas, es bastante diferente.

Se le solicita al usuario su nombre de usuario. Una vez que se ha introducido, se envía una solicitud al servidor de autenticación que contiene el nombre del usuario y el nombre de un servicio especial conocido como servicio de concesión de notificaciones.

El servidor de autenticación verifica que conoce el cliente. Si es así, genera una clave de sesión aleatoria que se utilizará más adelante entre el cliente y el servidor que concede notificaciones. A continuación, crea un ticket para el servidor de otorgamiento de notificaciones que contiene el nombre del cliente, el nombre del servidor de otorgamiento de notificaciones, la hora actual, una vida útil para el ticket, la dirección IP del cliente y la clave de sesión aleatoria recién creada. Todo esto está cifrado en una clave conocida sólo por el servidor que concede las notificaciones y el servidor de autenticación.

Luego, el servidor de autenticación envía el ticket, junto con una copia de la clave de sesión aleatoria y alguna información adicional, de vuelta al cliente. Esta respuesta está cifrada en la clave privada del cliente, conocida sólo por Kerberos y el cliente, que deriva de la contraseña del usuario.

Una vez que el cliente ha recibido la respuesta, se le solicita al usuario su contraseña. La contraseña se convierte en una clave DES y se utiliza para descifrar la respuesta del servidor de autenticación. El ticket y la clave de sesión, junto con otra información, se almacenan para su uso futuro, y la contraseña del usuario y la clave DES se borran de la memoria.

Una vez que se ha completado el intercambio, la estación de trabajo posee la información que puede utilizar para probar la identidad de su usuario durante el período de vigencia del billete que concede el billete. Mientras el software en la estación de trabajo no haya sido manipulado previamente, no existe información que permita que otra persona personifique al usuario más allá de la vida útil del billete.

Solicitar un servicio Kerberos

Por el momento, finjamos que el usuario ya tiene un ticket para el servidor deseado. Para obtener acceso al servidor, la aplicación genera un autenticador que contiene el nombre del cliente y la dirección IP, y la hora actual. El autenticador se cifra en la clave de sesión que se recibió con el ticket para el servidor. A continuación, el cliente envía el autenticador junto con el ticket al servidor de una manera definida por la aplicación individual.

Una vez que el servidor ha recibido el autenticador y el ticket, el servidor descifra el ticket, utiliza

la clave de sesión incluida en el ticket para descifrar el autenticador, compara la información del ticket con la del autenticador, la dirección IP desde la que se recibió la solicitud y la hora actual. Si todo coincide, permite que la solicitud continúe.

Se supone que los relojes se sincronizan en varios minutos. Si la hora de la solicitud es demasiado lejana en el futuro o en el pasado, el servidor trata la solicitud como un intento de reproducir una solicitud anterior. El servidor también puede realizar un seguimiento de todas las solicitudes anteriores con marcas de tiempo que aún son válidas. Con el fin de seguir frustrando los ataques de repetición, una solicitud recibida con el mismo ticket y el mismo sello de fecha y hora que el recibido ya puede ser descartada.

Por último, si el cliente especifica que desea que el servidor también demuestre su identidad, el servidor agrega uno a la marca de tiempo que el cliente envió en el autenticador, cifra el resultado en la clave de sesión y envía el resultado de vuelta al cliente.

Al final de este intercambio, el servidor está seguro de que, según Kerberos, el cliente es quien dice ser. Si se produce la autenticación mutua, el cliente también está convencido de que el servidor es auténtico. Además, el cliente y el servidor comparten una clave que nadie más conoce y pueden asumir con seguridad que un mensaje relativamente reciente cifrado en esa clave se originó con la otra parte.

Obtener notificaciones del servidor Kerberos

Recuerde que un ticket solo es bueno para un único servidor. Como tal, es necesario obtener un ticket separado para cada servicio que el cliente desea utilizar. Las entradas para servidores individuales se pueden obtener del servicio de otorgamiento de entradas. Dado que el servicio de otorgamiento de notificaciones es en sí mismo un servicio, hace uso del protocolo de acceso al servicio descrito en la sección anterior.

Cuando un programa requiere un ticket que aún no se ha solicitado, envía una solicitud al servidor que concede el ticket. La solicitud contiene el nombre del servidor para el que se solicita un ticket, junto con el ticket de otorgamiento de boletos y un autenticador construido como se describe en la sección anterior.

El servidor de otorgamiento de notificaciones verifica luego el ticket de autenticador y de otorgamiento de notificaciones como se describe anteriormente. Si es válido, el servidor que concede notificaciones genera una nueva clave de sesión aleatoria que se utilizará entre el cliente y el nuevo servidor. A continuación, crea un ticket para el nuevo servidor que contiene el nombre del cliente, el nombre del servidor, la hora actual, la dirección IP del cliente y la nueva clave de sesión que acaba de generar. La duración del nuevo billete es el mínimo de vida restante para el billete que concede el billete y el valor predeterminado para el servicio.

El servidor de otorgamiento de notificaciones envía entonces el ticket, junto con la clave de sesión y otra información, de vuelta al cliente. Esta vez, sin embargo, la respuesta está cifrada en la clave de sesión que fue parte del ticket de otorgamiento de boletos. De esta manera, no es necesario que el usuario introduzca su contraseña de nuevo.

La base de datos Kerberos

Hasta ahora, hemos hablado de operaciones que requieren acceso de sólo lectura a la base de datos Kerberos. Estas operaciones las realiza el servicio de autenticación, que puede ejecutarse

tanto en máquinas maestra como esclavas.

En esta sección, analizamos las operaciones que requieren acceso de escritura a la base de datos. Estas operaciones las realiza el servicio de administración, denominado Servicio de administración de bases de datos Kerberos (KDBM). La implementación actual estipula que sólo se pueden realizar cambios en la base de datos principal Kerberos; las copias esclavas son de sólo lectura. Por lo tanto, el servidor KDBM sólo puede ejecutarse en el equipo Kerberos maestro.

Tenga en cuenta que, aunque la autenticación todavía puede producirse (en esclavos), las solicitudes de administración no se pueden atender si la máquina maestra está inactiva. Según nuestra experiencia, esto no ha supuesto un problema, ya que las solicitudes de administración son poco frecuentes.

El KDBM gestiona las solicitudes de los usuarios para cambiar sus contraseñas. El lado cliente de este programa, que envía solicitudes al KDBM a través de la red, es el programa kpasswd. El KDBM también acepta solicitudes de los administradores Kerberos, que pueden agregar principios a la base de datos, así como cambiar las contraseñas de los principales existentes. El lado cliente del programa de administración, que también envía solicitudes al KDBM a través de la red, es el programa kadmin.

El servidor KDBM

El servidor KDBM acepta solicitudes para agregar principales a la base de datos o para cambiar las contraseñas de los principales existentes. Este servicio es único en el sentido de que el servicio de concesión de entradas no emitirá entradas para él. En su lugar, se debe utilizar el propio servicio de autenticación (el mismo servicio que se utiliza para obtener un ticket que concede el ticket). El propósito de esto es exigir al usuario que introduzca una contraseña. Si esto no fuera así, si un usuario dejara su estación de trabajo desatendida, un transeúnte podría subir y cambiarles su contraseña, algo que debería evitarse. De la misma manera, si un administrador deja su estación de trabajo sin vigilancia, un transeúnte podría cambiar cualquier contraseña en el sistema.

Cuando el servidor KDBM recibe una solicitud, la autoriza comparando el nombre principal autenticado del solicitante del cambio con el nombre principal del destino de la solicitud. Si son iguales, se permite la solicitud. Si no son iguales, el servidor KDBM consulta una lista de control de acceso (almacenada en un archivo del sistema Kerberos maestro). Si el nombre principal del solicitante se encuentra en este archivo, se permite la solicitud; de lo contrario, se deniega.

Por convención, los nombres con una instancia NULL (la instancia predeterminada) no aparecen en el archivo de lista de control de acceso; en su lugar, se utiliza una instancia administrativa. Por lo tanto, para que un usuario se convierta en administrador de Kerberos, se debe crear una instancia administrativa para ese nombre de usuario y agregarla a la lista de control de acceso. Esta convención permite que un administrador utilice una contraseña diferente para la administración de Kerberos y que la utilice para el inicio de sesión normal.

Se registran todas las solicitudes al programa KDBM, estén o no autorizadas.

Programas kadmin y kpasswd

Los administradores de Kerberos utilizan el programa kadmin para agregar principios a la base de datos o cambiar las contraseñas de los principales existentes. Se requiere que un administrador

introduzca la contraseña para su nombre de instancia administrativa cuando invoque el programa kadmin. Esta contraseña se utiliza para obtener un ticket para el servidor KDBM.

Los usuarios pueden cambiar sus contraseñas Kerberos utilizando el programa kpasswd. Se les exige que introduzcan la contraseña antigua cuando invoquen el programa. Esta contraseña se utiliza para obtener un ticket para el servidor KDBM.

Réplica de base de datos Kerberos

Cada rango Kerberos tiene un equipo Kerberos maestro, que aloja la copia maestra de la base de datos de autenticación. Es posible (aunque no necesario) disponer de copias adicionales de sólo lectura de la base de datos sobre máquinas esclavas en otros lugares del sistema. Las ventajas de tener varias copias de la base de datos son las que se suelen mencionar para su replicación: mayor disponibilidad y mejor rendimiento. Si la máquina maestra está inactiva, la autenticación puede lograrse en una de las máquinas esclavas. La capacidad de realizar la autenticación en cualquiera de las varias máquinas reduce la probabilidad de un cuello de botella en la máquina maestra.

El mantenimiento de varias copias de la base de datos plantea el problema de la coherencia de los datos. Hemos encontrado que métodos muy simples bastan para hacer frente a la incoherencia. La base de datos maestra se vierte cada hora. La base de datos se envía, en su totalidad, a las máquinas esclavas, que luego actualizan sus propias bases de datos. Un programa en el host maestro, llamado kprop, envía la actualización a un programa de peer, llamado kpropd, que se ejecuta en cada una de las máquinas esclavas. Primero, kprop envía una suma de comprobación de la nueva base de datos que está a punto de enviar. La suma de comprobación está cifrada en la clave de base de datos principal Kerberos, que poseen las máquinas Kerberos principal y esclava. Los datos se transfieren a través de la red al kpropd en la máquina esclava. El servidor de propagación esclavo calcula una suma de comprobación de los datos que ha recibido y, si coincide con la suma de comprobación enviada por el maestro, la nueva información se utiliza para actualizar la base de datos esclava.

Todas las contraseñas de la base de datos Kerberos están cifradas en la clave de la base de datos maestra. Por lo tanto, la información que pasa de maestro a esclavo a través de la red no es útil para un explorador. Sin embargo, es esencial que los esclavos acepten sólo la información del anfitrión principal y que se detecte la manipulación de los datos, por lo que se trata de la suma de comprobación.

Kerberos desde una perspectiva exterior

En esta sección se describe Kerberos desde el punto de vista práctico, primero como lo ve el usuario, después desde el punto de vista del programador de aplicaciones y, por último, a través de las tareas del administrador Kerberos.

Vista del usuario de Kerberos

Si todo va bien, el usuario apenas notará que Kerberos está presente. En nuestra implementación de UNIX, el ticket de otorgamiento de notificaciones se obtiene de Kerberos como parte del proceso de inicio de sesión. El cambio de la contraseña Kerberos de un usuario forma parte del programa passwd. Y los tickets Kerberos se destruyen automáticamente cuando un usuario cierra la sesión.

Si la sesión de inicio de sesión del usuario dura más de la duración del ticket de concesión de notificaciones (actualmente, 8 horas), el usuario notará la presencia de Kerberos porque la próxima vez que se ejecute una aplicación autenticada por Kerberos, se producirá un error. El ticket Kerberos para él habrá caducado. En ese momento, el usuario puede ejecutar el programa kinit para obtener un nuevo ticket para el servidor que concede el ticket. Al igual que al iniciar sesión, se debe proporcionar una contraseña para obtenerla. Un usuario que ejecuta el comando klist por curiosidad puede sorprenderse de todas las notificaciones que se han obtenido silenciosamente en su nombre para los servicios que requieren autenticación Kerberos.

Kerberos desde el punto de vista del programador

Un programador que escribe una aplicación Kerberos a menudo agregará autenticación a una aplicación de red ya existente formada por un cliente y un servidor. Llamamos a este proceso "Kerberizing" (Kerberizing) un programa. La identificación de claves generalmente implica realizar una llamada a la biblioteca Kerberos para realizar la autenticación en la solicitud inicial de servicio. También puede implicar llamadas a la biblioteca DES para cifrar mensajes y datos que se envían posteriormente entre el cliente de aplicación y el servidor de aplicaciones.

Las funciones de biblioteca más utilizadas son krb_mk_req en el lado del cliente y krb_rd_req en el lado del servidor. La rutina krb_mk_req toma como parámetros el nombre, la instancia y el rango del servidor de destino, que se solicitarán, y posiblemente una suma de verificación de los datos que se enviarán. El cliente entonces envía el mensaje devuelto por la llamada krb_mk_req a través de la red al lado del servidor de la aplicación. Cuando el servidor recibe este mensaje, realiza una llamada a la rutina de biblioteca krb_rd_req. La rutina devuelve un juicio acerca de la autenticidad de la supuesta identidad del remitente.

Si la aplicación requiere que los mensajes enviados entre el cliente y el servidor sean secretos, se pueden realizar llamadas de biblioteca a krb_mk_priv (krb_rd_priv) para cifrar (descifrar) mensajes en la clave de sesión que comparten ambos lados ahora.

La tarea del administrador de Kerberos

El trabajo del administrador Kerberos comienza con la ejecución de un programa para inicializar la base de datos. Se debe ejecutar otro programa para registrar los principales esenciales en la base de datos, como el nombre del administrador Kerberos con una instancia administrativa. Se deben iniciar el servidor de autenticación Kerberos y el servidor de administración. Si hay bases de datos esclavas, el administrador debe organizar que los programas para propagar actualizaciones de bases de datos de maestros a esclavos se inicien periódicamente.

Después de que se hayan realizado estos pasos iniciales, el administrador manipula la base de datos a través de la red, utilizando el programa kadmin. A través de ese programa, se pueden agregar nuevos principios y cambiar las contraseñas.

En particular, cuando se agrega una nueva aplicación Kerberos al sistema, el administrador Kerberos debe realizar algunos pasos para que funcione. El servidor debe estar registrado en la base de datos y se le debe asignar una clave privada (normalmente es una clave aleatoria generada automáticamente). A continuación, algunos datos (incluida la clave del servidor) deben extraerse de la base de datos e instalarse en un archivo del equipo del servidor. El archivo predeterminado es /etc/srvtab. La rutina de la biblioteca krb_rd_req a la que llama el servidor (consulte la sección anterior) utiliza la información de ese archivo para descifrar los mensajes enviados cifrados en la clave privada del servidor. El archivo /etc/srvtab autentica el servidor cuando una contraseña tecleada en un terminal autentica al usuario.

El administrador de Kerberos también debe asegurarse de que las máquinas Kerberos estén físicamente seguras, y también sería prudente mantener copias de seguridad de la base de datos principal.

Descripción detallada de Kerberos

En esta sección, describimos cómo Kerberos encaja en el entorno Athena, incluido su uso por otros servicios de red y aplicaciones, y cómo interactúa con rangos Kerberos remotos. Para obtener una descripción más completa del entorno Athena, consulte G.W. Treese.

Utilización de Kerberos de otros servicios de red

Se han modificado varias aplicaciones de red para utilizar Kerberos. Los comandos rlogin y rsh primero intentan autenticarse usando Kerberos. Un usuario con entradas Kerberos válidas puede iniciar sesión en otra máquina Athena sin tener que configurar archivos .rhosts. Si la autenticación Kerberos falla, los programas recurren a sus métodos de autorización habituales, en este caso, los archivos .rhosts.

Hemos modificado el Protocolo de oficina de correos para utilizar Kerberos para autenticar a los usuarios que desean recuperar su correo electrónico de la "oficina de correos". Recientemente se ha desarrollado en Athena un programa de entrega de mensajes, llamado Zephyr, que también utiliza Kerberos para la autenticación.

El programa para registrar nuevos usuarios, denominado registro, utiliza tanto el Sistema de administración de servicios (SMS) como Kerberos. Desde SMS, determina si la información introducida por el futuro usuario de Athena, como el nombre y el número de identificación del MIT, es válida. Luego verifica con Kerberos para ver si el nombre de usuario solicitado es único. Si todo va bien, se realiza una nueva entrada en la base de datos Kerberos, que contiene el nombre de usuario y la contraseña.

Para obtener información detallada sobre el uso de Kerberos para proteger el sistema de archivos de red de Sun, consulte el apéndice.

Interacción con otro Kerberi

Se espera que diferentes organizaciones administrativas deseen utilizar Kerberos para la autenticación de usuarios. También se espera que, en muchos casos, los usuarios de una organización deseen utilizar los servicios en otra. Kerberos admite varios dominios administrativos. La especificación de nombres en Kerberos incluye un campo denominado rango. Este campo contiene el nombre del dominio administrativo en el que se autenticará al usuario.

Los servicios generalmente se registran en un solo rango y sólo aceptarán las credenciales emitidas por un servidor de autenticación para ese rango. Por lo general, un usuario se registra en un solo rango (el rango local), pero es posible para él obtener credenciales emitidas por otro rango (el rango remoto), sobre la fuerza de la autenticación proporcionada por el rango local. Las credenciales válidas en un rango remoto indican el rango en el que se autenticó originalmente al usuario. Los servicios en el ámbito remoto pueden elegir si se deben cumplir esas credenciales, dependiendo del grado de seguridad requerido y del nivel de confianza en el rango que autenticó inicialmente al usuario.

Para realizar la autenticación de rango cruzado, es necesario que los administradores de cada

par de rangos seleccionen una clave para ser compartida entre sus rangos. Un usuario en el rango local puede entonces solicitar un ticket de otorgamiento de entradas del servidor de autenticación local para el servidor de otorgamiento de entradas en el rango remoto. Cuando se utiliza ese ticket, el servidor de otorgamiento de ticket remoto reconoce que la solicitud no es de su propio rango, y utiliza la clave previamente intercambiada para descifrar el ticket de otorgamiento de ticket. Luego emite un ticket como lo haría normalmente, excepto que el campo de rango para el cliente contiene el nombre del rango en el que el cliente fue autenticado originalmente.

Este enfoque podría ampliarse para que uno pueda autenticarse a través de una serie de rangos hasta alcanzar el rango con el servicio deseado. Para ello, sin embargo, sería necesario registrar el trayecto completo que se tomó, y no sólo el nombre del rango inicial en el que se autenticó al usuario. En tal situación, todo lo que sabe el servidor es que A dice que B dice que C dice que el usuario es así. Esta declaración sólo puede ser de confianza si todos los usuarios de la ruta también son de confianza.

Problemas de Kerberos y problemas abiertos

Hay varios problemas y problemas abiertos asociados con el mecanismo de autenticación Kerberos. Entre los problemas se encuentran cómo decidir la duración correcta de un ticket, cómo permitir proxies y cómo garantizar la integridad de la estación de trabajo.

El problema de la vida útil de los billetes es una cuestión de elegir el equilibrio adecuado entre seguridad y conveniencia. Si la vida de un billete es larga, entonces si un billete y su clave de sesión asociada son robados o extraviados, pueden ser usados por un período de tiempo más largo. Dicha información puede ser robada si un usuario se olvida de cerrar sesión en una estación de trabajo pública. De manera alternativa, si un usuario se ha autenticado en un sistema que permite varios usuarios, otro usuario con acceso a root podría encontrar la información necesaria para usar las notificaciones robadas. Sin embargo, el problema de dar una duración corta a un billete es que cuando caduque, el usuario tendrá que obtener uno nuevo que requiera que el usuario introduzca la contraseña de nuevo.

Un problema abierto es el problema del proxy. ¿Cómo puede un usuario autenticado permitir que un servidor adquiera otros servicios de red en su nombre? Un ejemplo donde esto sería importante es el uso de un servicio que obtendrá acceso a archivos protegidos directamente desde un servidor de archivos. Otro ejemplo de este problema es lo que llamamos reenvío de autenticación. Si un usuario inicia sesión en una estación de trabajo e inicia sesión en un host remoto, sería bueno que el usuario tuviera acceso a los mismos servicios disponibles localmente mientras ejecuta un programa en el host remoto. Lo que hace esto difícil es que el usuario podría no confiar en el host remoto, por lo que el reenvío de autenticación no es deseable en todos los casos. Actualmente no tenemos una solución a este problema.

Otro problema, que es importante en el entorno de Athena, es cómo garantizar la integridad del software que se ejecuta en una estación de trabajo. Esto no es tanto un problema en estaciones de trabajo privadas, ya que el usuario que lo va a utilizar tiene control sobre él. Sin embargo, en estaciones de trabajo públicas, alguien podría haber venido y modificado el programa de inicio de sesión para guardar la contraseña del usuario. La única solución disponible actualmente en nuestro entorno es dificultar la modificación del software que se ejecuta en las estaciones de trabajo públicas. Una solución mejor requeriría que la clave del usuario nunca deje un sistema en el que el usuario sepa que se puede confiar. Una manera de hacerlo sería si el usuario tuviera una tarjeta inteligente capaz de hacer los cifrados requeridos en el protocolo de autenticación.

Estado de Kerberos

Una versión prototipo de Kerberos entró en producción en septiembre de 1986. Desde enero de 1987, Kerberos ha sido el único medio del Proyecto Athena para autenticar sus 5,000 usuarios, 650 estaciones de trabajo y 65 servidores. Además, Kerberos se está utilizando ahora en lugar de archivos .rhosts para controlar el acceso en varios de los sistemas de intercambio de horas de Athena.

Reconocimientos de Kerberos

Kerberos fue inicialmente diseñado por Steve Miller y Clifford Neuman con sugerencias de Jeff Schiller y Jerry Saltzer. Desde entonces, muchas otras personas han participado en el proyecto. Entre ellos están Jim Aspnes, Bob Baldwin, John Barba, Richard Basch, Jim Bloom, Bill Bryant, Mark Colan, Rob French, Dan Geer, John Kohl, John Kubiatowicz, Bob Mckie, Brian Murphy, John Ostlund Ken Raeburn, Chris Reed, Jon Rochlis, Mike Shanzer, Bill Sommerfeld, Win Treese y Stan Zanarotti.

Agradecemos a Dan Geer, Kathy Lieben, Josh Lubarr, Ken Raeburn, Jerry Saltzer, Ed Steiner, Robbert van Renesse y Win Treese cuyas sugerencias mejoraron mucho los borradores anteriores de este documento.

Jedlinsky, J.T. Kohl y W.E. Sommerfeld, "The Zephyr Notification System", en las actas de conferencia de Usenix (Invierno, 1988).

M.A. Rosenstein, D.E. Geer y P.J. Levine, en las actas de conferencia de Usenix (Invierno, 1988).

R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh y B. Lyon, "Design and Implementation of the Sun Network Filesystem" (Diseño e implementación del sistema de archivos de red Sun), en las actas de la conferencia de Usenix (verano, 1985).

Apéndice: Aplicación Kerberos al Network File System (NFS) de SUN

Un componente clave del sistema de estación de trabajo del Proyecto Athena es la interposición de la red entre la estación de trabajo del usuario y su almacenamiento privado de archivos (directorio personal). Todo el almacenamiento privado reside en un conjunto de ordenadores (actualmente VAX 11/750) dedicados a este fin. Esto nos permite ofrecer servicios en estaciones de trabajo UNIX disponibles públicamente. Cuando un usuario inicia sesión en una de estas estaciones de trabajo disponibles públicamente, en lugar de validar su nombre y contraseña en un archivo de contraseña residente localmente, utilizamos Kerberos para determinar su autenticidad. El programa de inicio de sesión solicita un nombre de usuario (como en cualquier sistema UNIX). Este nombre de usuario se utiliza para obtener un ticket de otorgamiento de notificaciones Kerberos. El programa de inicio de sesión utiliza la contraseña para generar una clave DES para descifrar el ticket. Si el descifrado es exitoso, el directorio principal del usuario se encuentra consultando el servicio de nombres Hesiod y se monta a través de NFS. A continuación, el programa de inicio de sesión pasa el control al shell del usuario, que puede ejecutar los archivos de personalización tradicionales por usuario porque el directorio principal ahora está "conectado" a la estación de trabajo. El servicio Hesiod también se utiliza para construir una entrada en el archivo de contraseña local. (Esto es para beneficio de los programas que buscan información en /etc/passwd.)

A partir de varias opciones para la entrega del servicio de archivos remoto, elegimos Sun Network File System. Sin embargo, este sistema no se ajusta a nuestras necesidades de una manera crucial. NFS asume que todas las estaciones de trabajo se dividen en dos categorías (tal y como se ven desde el punto de vista de un servidor de archivos): fiable y no fiable. Los sistemas no confiables no pueden acceder a ningún archivo en absoluto, los de confianza sí. Los sistemas de confianza son totalmente fiables. Se supone que un sistema de confianza se gestiona mediante una gestión sencilla. Específicamente, desde una estación de trabajo de confianza es posible enmascarar como cualquier usuario válido del sistema de servicio de archivos y así obtener acceso a casi todos los archivos del sistema. (Sólo los archivos propiedad de "root" están exentos.)

En nuestro entorno, la gestión de una estación de trabajo (en el sentido tradicional de la administración del sistema UNIX) está en manos del usuario que la utiliza actualmente. No hacemos ningún secreto de la contraseña raíz en nuestras estaciones de trabajo, ya que nos damos cuenta de que un usuario realmente poco amigable puede entrar por el hecho mismo de que está sentado en la misma ubicación física que la máquina y tiene acceso a todas las funciones de la consola. Por lo tanto, no podemos confiar realmente en nuestras estaciones de trabajo en la interpretación NFS de la confianza. Para permitir los controles de acceso adecuados en nuestro entorno, tuvimos que hacer algunas modificaciones en el software NFS base e integrar Kerberos en el esquema.

NFS no modificado de Kerberos

En la implementación de NFS con la que empezamos (desde la Universidad de Wisconsin), la autenticación se proporcionó en la forma de un fragmento de datos incluidos en cada solicitud NFS (llamado "credencial" en terminología NFS). Esta credencial contiene información sobre el identificador de usuario único (UID) del solicitante y una lista de los identificadores de grupo (GID) de la pertenencia al solicitante. Luego, el servidor NFS utiliza esta información para la verificación de acceso. La diferencia entre una estación de trabajo confiable y una no confiable es si sus credenciales son aceptadas o no por el servidor NFS.

NFS modificado por Kerberos

En nuestro entorno, los servidores NFS deben aceptar las credenciales de una estación de trabajo si y sólo si las credenciales indican el UID del usuario de la estación de trabajo, y ningún otro.

Una solución obvia sería cambiar la naturaleza de las credenciales de simples indicadores de UID y GID a datos autenticados Kerberos completos. Sin embargo, se pagaría una pena significativa por rendimiento si se adoptara esta solución. Las credenciales se intercambian en cada operación NFS, incluidas todas las actividades de lectura y escritura del disco. La inclusión de una autenticación Kerberos en cada transacción de disco añadiría un número considerable de cifrados completos (realizados en software) por transacción y, según nuestros cálculos de sobres, habría proporcionado un rendimiento inaceptable. (También habría requerido colocar las rutinas de la biblioteca Kerberos en el espacio de dirección del núcleo.)

Necesitábamos un enfoque híbrido, que se describe a continuación. La idea básica es tener las credenciales del mapa del servidor NFS recibidas de estaciones de trabajo cliente, a una credencial válida (y posiblemente diferente) en el sistema del servidor. Este mapeo se realiza en el kernel del servidor en cada transacción NFS y se configura en tiempo de "montaje" mediante un proceso de nivel de usuario que se involucra en la autenticación moderada Kerberos antes de

establecer un mapeo de credenciales del núcleo válido.

Para implementar esto, agregamos una nueva llamada del sistema al núcleo (necesaria sólo en sistemas de servidor, no en sistemas de cliente) que proporciona el control de la función de mapeo que asigna las credenciales entrantes de estaciones de trabajo cliente a credenciales válidas para el uso en el servidor (si las hubiera). La función de mapeo básica mapea el tuple:

```
<CLIENT-IP-ADDRESS, UID-ON-CLIENT>
```

a una credencial NFS válida en el sistema del servidor. El CLIENT-IP-ADDRESS se extrae del paquete de solicitud NFS suministrado por el sistema cliente. Nota: se descarta toda la información de la credencial generada por el cliente excepto UID-ON-CLIENT.

Si no existe ninguna asignación, el servidor reacciona de una de dos maneras, dependiendo de que esté configurado. En nuestra configuración amigable, establecemos de forma predeterminada las solicitudes no asignadas en las credenciales para el usuario "nadie" que no tiene acceso privilegiado y tiene un UID único. Los servidores poco descriptivos devuelven un error de acceso NFS cuando no se puede encontrar una asignación válida para una credencial NFS entrante.

Nuestra nueva llamada del sistema se utiliza para agregar y eliminar entradas del mapa residente en el núcleo. También proporciona la capacidad de vaciar todas las entradas que se asignan a un UID específico en el sistema de servidor, o vaciar todas las entradas de un CLIENT-IP-ADDRESS dado.

Modificamos el demonio de montaje (que gestiona las solicitudes de montaje NFS en sistemas de servidor) para aceptar un nuevo tipo de transacción, la solicitud de asignación de autenticación Kerberos. Básicamente, como parte del proceso de montaje, el sistema cliente proporciona un autenticador Kerberos junto con una indicación de su UID-ON-CLIENT (cifrado en el autenticador Kerberos) en la estación de trabajo. El demonio de montaje del servidor convierte el nombre principal Kerberos en un nombre de usuario local. A continuación, se busca este nombre de usuario en un archivo especial para generar la lista de UID y GID del usuario. Para mayor eficiencia, este archivo es un archivo de base de datos ndbm con el nombre de usuario como clave. A partir de esta información, se construye una credencial NFS y se entrega al núcleo como la asignación válida del tuplo <CLIENT-IP-ADDRESS, CLIENT-UID> para esta solicitud.

En el momento de desmontaje, se envía una solicitud al demonio de montaje para quitar el mapeo previamente agregado del kernel. También es posible enviar una solicitud a la hora de cierre de sesión para invalidar toda la asignación para el usuario actual en el servidor en cuestión, limpiando así cualquier asignación restante que exista (aunque no debería) antes de que la estación de trabajo esté disponible para el siguiente usuario.

Consecuencias en la seguridad de Kerberos de NFS modificado

Esta implementación no es completamente segura. Para empezar, los datos de los usuarios se siguen enviando a través de la red en un formulario no cifrado y, por lo tanto, interceptable. La autenticación de bajo nivel por transacción se basa en un par <CLIENT-IP-ADDRESS, CLIENT-UID> proporcionado sin cifrar en el paquete de solicitud. Esta información podría falsificarse y, por lo tanto, poner en peligro la seguridad. Sin embargo, se debe tener en cuenta que sólo mientras un usuario utiliza activamente sus archivos (es decir, mientras está conectado), hay asignaciones válidas en su lugar y, por lo tanto, esta forma de ataque se limita a cuando el usuario en cuestión está conectado. Cuando un usuario no ha iniciado sesión, ninguna cantidad de falsificación de direcciones IP permitirá el acceso no autorizado a sus archivos.

Referencias de Kerberos

1. S.P. Miller, B.C. Neuman, J.I. Schiller y J.H. Saltzer, sección E.2.1: Sistema de autenticación y autorización Kerberos, M.I.T. Project Athena, Cambridge, Massachusetts (21 de diciembre de 1987).
2. E. Balkovich, S.R. Lerman y R.P. Parmelee, "Computing in Higher Education: The Athena Experience", Communications of the ACM, Vol. 28(11), pp. 1214-1224, ACM (noviembre, 1985).
3. R.M. Needham y M.D. Schroeder, "Using Encryption for Authentication in Large Networks of Computers", Comunicaciones de ACM, Vol. 21(12), pp. 993-999 (diciembre de 1978).
4. V.L. Voydock y S.T. Kent, "Security Mecms in High-Level Network Protocols", Computing Surveys, vol. 15(2), ACM (junio de 1983).
5. Oficina Nacional de Normas, "Data Encryption Standard", publicación 46 de las Normas Federales de Procesamiento de Información, Oficina de Impresión del Gobierno, Washington, DC (1977).
6. SP Dyer, "Hesiod", en las actas de conferencia de Usenix (Invierno, 1988).
7. W.J. Bryant, Tutorial del Programador Kerberos, Proyecto Athena del MIT (En preparación).
8. W.J. Bryant, Manual del Administrador de Kerberos, Proyecto Athena del MIT (En preparación).
9. G.W. Treese, "Berkeley Unix en 1000 estaciones de trabajo: Athena cambia a 4.3BSD", en las actas de conferencia de Usenix (Invierno, 1988).
10. C.A. DellaFera, M.W. Eichin, R.S. French, D.C. Jedlinsky, J.T. Kohl y W.E. Sommerfeld, "The Zephyr Notification System", en las actas de conferencia de Usenix (Invierno, 1988).
11. M.A. Rosenstein, D.E. Geer y P.J. Levine, en las actas de conferencia de Usenix (Invierno, 1988).
12. R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh y B. Lyon, "Design and Implementation of the Sun Network Filesystem" (Diseño e implementación del sistema de archivos de red Sun), en las actas de la conferencia de Usenix (verano, 1985).

Información Relacionada

- Página de soporte de Kerberos
- Soporte Técnico y Documentación - Cisco Systems