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, proyecto 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 era un miembro del personal directivo del proyecto Athena durante la fase del diseño y de instrumentación inicial de Kerberos.
- Jeffrey I. Schiller, proyecto Athena, Massachusetts Institute of Technology, Cambridge, MA 02139, jis@ATHENA.MIT.EDU

Introducción a Kerberos

Este papel da una descripción del Kerberos, un sistema de autenticación diseñado por Miller y Neuman. para los entornos de la computación de red abierta, y describe nuestra experiencia usando ella en el proyecto Athena MIT. En la sección en la motivación, explicamos porqué un nuevo modelo de autenticación es necesario para las redes abiertas, y cuáles son sus requisitos. ¿Cuál es Kerberos? seccione las listas los componentes del software Kerberos y describe cómo obran recíprocamente en proporcionar el servicio de autenticación. En el Kerberos nombra la sección, nosotros describen el esquema de asignación de nombres del Kerberos.

Cómo el Kerberos trabaja los presentes los bloques de construcción de autenticación de Kerberos - el boleto y el authenticator. Esto lleva a una discusión de los dos Protocolos de autenticación: la autenticación inicial de un usuario al Kerberos (análogo a la apertura de sesión), y el protocolo para la autenticación recíproca de un consumidor potencial y de un productor potencial de un servicio de red.

El Kerberos requiere una base de datos de información sobre sus clientes; la sección de la base de datos Kerberos describe la base de datos, su Administración, y el protocolo para su modificación. El Kerberos del exterior que mira en la sección describe el Kerberos interconecta a sus usuarios, programadores de las aplicaciones, y administradores. En la sección de la visión general, describimos cómo los ajustes del Kerberos del proyecto Athena en el resto del entorno Athena. También describimos la interacción de diversos dominios de autenticación Kerberos, o de los reinos; en nuestro caso, la relación entre el Kerberos del proyecto Athena y el Kerberos que se ejecutan en el laboratorio MIT para las Ciencias de la computación.

En los temas y problemas abiertos sección, mencionamos las cuestiones abiertas y los problemas hasta ahora sin resolver. La sección más reciente da el estado actual de Kerbero en el proyecto Athena. En el apéndice, describimos detalladamente cómo el Kerberos se aplica a un servicio de archivo de red para autenticar a los usuarios que desean acceder a los sistemas de archivos remotos.

Conceptos Kerberos

En este papel utilizamos los términos que pueden ser ambiguos, nuevo al lector, o utilizado diverso a otra parte. Debajo de los nosotros estado nuestro uso de esos términos.

¿*Usuario, cliente, servidor*? Por el usuario, significamos a un ser humano que utilice un programa o un servicio. Un cliente también utiliza algo, pero no es necesariamente una persona; puede ser un programa. Las aplicaciones de red consisten en a menudo dos porciones; un programa que se ejecuta en una máquina y pide un servicio remoto, y otro programa que los funcionamientos en la máquina remota y lleva a cabo ese servicio. Llamamos ésos el lado del cliente y el lado del servidor de la aplicación, respectivamente. A menudo, un cliente entrará en contacto un servidor en nombre de un usuario.

Cada entidad que utiliza el sistema Kerberos, sea un usuario o un servidor de red, es en un sentido un cliente, puesto que utiliza el servicio Kerberos. Para distinguir tan a los clientes Kerberos de los clientes de otro servicio, utilizamos el principal del término para indicar tal entidad. Observe que un mandante identidad única de Kerberos puede ser usuario o un servidor. (Describimos el nombramiento de los mandantes identidad única de Kerberos en una sección posterior.)

¿*Mantenga contra el servidor*? Utilizamos el servicio como una especificación abstracta de algunas acciones que se realizarán. Un proceso que realiza esas acciones se llama un servidor. En un momento dado, puede haber varios servidores (que se ejecutan generalmente en diversas máquinas) que llevan a cabo un servicio dado. Por ejemplo, en Athena hay un servidor de rlogin del BSD UNIX que se ejecuta en cada uno de nuestras máquinas del reparto del tiempo.

¿*Clave, clave privada, contraseña*? Encripción de claves privadas de las aplicaciones del Kerberos. Cada mandante identidad única de Kerberos se asigna un número grande, su clave privada, sabida solamente a ése principal y al Kerberos. En el caso de un usuario, la clave privada es el resultado de una función unidireccional aplicada a la contraseña de usuario. Utilizamos la clave como taquigrafía para la clave privada.

¿*Credenciales*? Desafortunadamente, esta palabra tiene un significado especial para el Sistema de archivo de red de Sun y el sistema Kerberos. Estado explícitamente si significamos los credenciales NFS o los credenciales de Kerberos, si no el término se utiliza en el sentido de idioma inglés normal.

¿*Master y esclavo*? Es posible funcionar con el software de la autenticación de Kerberos en más de una máquina. Sin embargo, hay siempre solamente una copia definitiva de la base de datos Kerberos. La máquina que contiene esta base de datos se llama la máquina principal, o apenas el master. Otras máquinas pueden poseer las copias de sólo lectura de la base de datos Kerberos, y éstos se llaman los esclavos.

La motivación detrás de Kerberos

En un entorno NON-conectado de la computación personal, los recursos y la información pueden ser protegidos físicamente asegurando la computadora personal. En un entorno de computación del reparto del tiempo, el sistema operativo protege a los usuarios a partir del uno otro y controla los recursos. Para determinar cuál puede cada usuario leer o modificarse, es necesario que el sistema de tiempo compartido identifique a cada usuario. Esto es realizado cuando el usuario abre una sesión.

En una red de usuarios que requiere los servicios de muchos ordenadores separados, hay tres acercamientos uno puede llevar el control de acceso: Uno no puede hacer nada, confiando en la máquina a la cual abren una sesión al usuario para prevenir el acceso no autorizado; uno puede requerir al host probar su identidad, pero confía en la palabra del host en cuanto a quién es el usuario; o uno puede requerir al usuario probar su identidad para cada servicio solicitado.

En un entorno cerrado donde están todas las máquinas bajo control estricto, uno puede utilizar el primer acercamiento. Cuando las organizaciones controlas todos los host que comunican sobre la red, esto son un acercamiento razonable.

En más el entorno abierto, uno pudo confiar en selectivamente solamente esos host bajo control organizativo. En este caso, cada host se debe requerir probar su identidad. Los programas de rlogin y rsh utilizan este acercamiento. En esos protocolos, la autenticación es hecha marcando a la dirección de Internet de quien se ha establecido una conexión.

En el entorno Athena, debemos poder honrar las peticiones de los host que no están bajo control organizativo. Los usuarios tienen control total de sus puestos de trabajo: pueden reiniciarlos, traerlos encima de independiente, o aún iniciar sus las propio las cintas. Como tal, el tercer acercamiento debe ser tomado; el usuario debe probar su identidad para cada servicio deseado. El servidor debe también probar su identidad. No es suficiente asegurar físicamente el host que funciona con a un servidor de red; alguien a otra parte en la red puede disfrazarse como el servidor determinado.

Nuestro entorno pone los varios requerimientos en un mecanismo de identificación. Primero, debe ser seguro. Evitarlo debe ser bastante difícil que un atacante potencial no encuentra el mecanismo de autenticación para ser el punto débil. Alguien que mira la red no debe poder obtener la información necesaria personificar a otro usuario. En segundo lugar, debe ser confiable. El acceso a muchos servicios dependerá del servicio de autenticación. Si no es confiable, el sistema de servicios en conjunto no estará. Tercero, debe ser transparente. Idealmente, el usuario no debe ser consciente de la autenticación que ocurre. Finalmente, debe ser scalable. Muchos sistemas pueden comunicar con los host de Athena. No todos los éstos soportarán nuestro mecanismo, pero el software no debe romperse si lo hicieron.

El Kerberos es el resultado de nuestro trabajo para satisfacer los requisitos antedichos. Cuando un usuario se acerca hasta un puesto de trabajo inician sesión. Por lo que el usuario puede decir, esta identificación inicial es suficiente probar su identidad a todos los servidores de red requerida para la duración de la sesión de conexión al sistema. La Seguridad del Kerberos confía en las seguridades de varios servidores de autenticación, pero no en el sistema del cual los usuarios inician sesión, ni en la Seguridad de los servidores extremos que serán utilizados. El servidor de autenticación proporciona correctamente a un usuario autenticado con una manera de probar su identidad a los servidores dispersados a través de la red.

La autenticación es un bloque de construcción fundamental para un entorno interconectado en red seguro. Si, por ejemplo, un servidor conoce con certeza la identidad de un cliente, puede decidir a si proporcionar el servicio, si el usuario debe ser dado los privilegios especiales, que deben recibir la cuenta para el servicio, y así sucesivamente. Es decir los esquemas de la autorización y de las estadísticas se pueden construir encima de la autenticación que el Kerberos proporciona, dando por resultado la seguridad equivalente a la computadora personal solitaria o al sistema de tiempo compartido.

¿Qué es Kerberos?

El Kerberos es un servicio confiable de autenticación de terceros basado en el modelo presentado por Needham y el Schroeder. Se confía en el sentido ese cada uno del juicio de su Kerberos de las creencias de los clientes en cuanto a la identidad de cada uno de sus otros clientes de ser exacto. Los grupos fecha/hora (números grandes que representan la fecha y hora actual) se han agregado al modelo original para ayudar en la detección de respuesta. La respuesta ocurre cuando un mensaje se roba de la red y se vuelve a enviar más adelante. Para una más descripción completa de la respuesta, y otras aplicaciones la autenticación, vea Voydock y Kent.

¿Qué hace Kerberos?

El Kerberos guarda una base de datos de sus clientes y de sus claves privadas. La clave privada es un número grande sabido solamente al Kerberos y al cliente que pertenece a. En caso de que el cliente sea usuario, es una contraseña encriptada. Los servicios de red que requieren la autenticación se registran con el Kerberos, al igual que los clientes que desean utilizar esos servicios. Las claves privadas se negocian en el registro.

Porque el Kerberos conoce estas claves privadas, puede crear los mensajes que convencen a un cliente de que otros son realmente quién demanda ser. El Kerberos también genera las claves privadas temporales, llamadas las claves de la sesión, que se dan a dos clientes y nadie más. Una clave de la sesión se puede utilizar para cifrar los mensajes entre dos partidos.

El Kerberos proporciona tres niveles claros de protección. El programador de la aplicación determina que es apropiado, según los requisitos de la aplicación. Por ejemplo, algunas aplicaciones requieren solamente que la autenticidad esté establecida en el lanzamiento de una conexión de red, y pueden asumir que los otros mensajes de una determinada dirección de red originan del partido autenticado. Nuestro Network File System autenticado utiliza este nivel de seguridad.

Otras aplicaciones requieren la autenticación de cada mensaje, pero no cuidan si el contenido del mensaje esté divulgado o no. Para éstos, el Kerberos proporciona los mensajes seguros. Con todo un de alto nivel de la Seguridad es proporcionada por los mensajes privados, donde cada mensaje no sólo se autentica, pero también cifrada. Los mensajes privados son utilizados, por ejemplo, por el servidor de Kerberos sí mismo para enviar las contraseñas sobre la red.

Componentes del software Kerberos

La instrumentación Athena comprende varios módulos:

- Biblioteca de las aplicaciones Kerberos
- biblioteca de encripción
- biblioteca de la base de datos
- programas de la Administración de base de datos
- servidor de la administración
- servidor de autenticación
- software de difusión DB
- programas del usuario
- aplicaciones

La biblioteca de las aplicaciones Kerberos proporciona una interfaz para los clientes de la aplicación y los servidores de aplicaciones. Contiene, entre otros, las rutinas para los pedidos de autenticación que crean o de lecturas, y las rutinas para crear la caja fuerte o los mensajes privados.

El cifrado en el Kerberos se basa en el DES, la Data Encryption Standard. La biblioteca de encripción implementa esas rutinas. Varios métodos de encripción se proporcionan, los equilibrios entre la velocidad y la Seguridad. Una extensión al modo del DES Cypher Block Chaining (CBC), llamado el modo CBC de propagación, también se proporciona. En el CBC, un error se propaga solamente a través del bloque actual de la cifra, mientras que en el PCBC, el error se propaga en el mensaje. Esto hace el mensaje completo inútil si ocurre un error, bastante que apenas una porción de él. La biblioteca de encripción es un módulo independiente, y se puede substituir por otras implementaciones de DES o una diversa biblioteca de encripción.

Otro módulo reemplazable es el Sistema de administración de la base de datos. La instrumentación actual de Athena de la biblioteca de la base de datos utiliza el ndbm, aunque Ingres fuera utilizado originalmente. Otras bibliotecas de la administración de base de datos se podían utilizar también.

Las necesidades de la base de datos Kerberos son directas; un expediente se lleva a cabo para cada principal, conteniendo el nombre, la clave privada, y la fecha de vencimiento del principal, junto con una cierta información administrativa. (La fecha de vencimiento es la fecha después de lo cual una entrada es no más válida. Se fija generalmente a algunos años en el futuro en el registro.)

La otra información del usuario, tal como Nombre real, número de teléfono, y así sucesivamente, es guardada por otro servidor, el nameserver de Hesiod. Esta manera, información vulnerable, a saber las contraseñas, se puede manejar por el Kerberos, usando bastante las medidas de gran seguridad; mientras que la información no sensible guardada por Hesiod se trata de diferentemente; puede, por ejemplo, ser enviada unencrypted sobre la red.

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

El servidor de la administración (o el servidor KDBM) proporciona una interfaz de la red de lectura/grabación a la base de datos. El lado del cliente del programa se puede ejecutar en cualquier máquina en la red. El lado del servidor, sin embargo, debe ejecutarse en la máquina que contiene la base de datos Kerberos para realizar los cambios a la base de datos.

El servidor de autenticación (o el servidor de Kerberos), por otra parte, realiza las operaciones de sólo lectura en la base de datos Kerberos, a saber, la autenticación de principales, y la generación de claves de la sesión. Puesto que este servidor no modifica la base de datos Kerberos, puede ejecutarse en una máquina que contiene una copia de sólo lectura de la base de datos principal de Kerberos.

El software de difusión de la base de datos maneja la replicación de la base de datos Kerberos. Es posible tener copias de la base de datos en varias diversas máquinas, con una copia del servidor de autenticación que se ejecuta en cada máquina. Cada uno de estas máquinas auxiliares recibe una actualización de la base de datos Kerberos de la máquina principal en los intervalos determinados.

Finalmente, hay programas del usuario final para abrir una sesión al Kerberos, cambiar una contraseña de Kerberos, y visualizar o destruirlos los boletos del Kerberos (los boletos se explican después).

Nombres de Kerberos

La parte de que autentica una entidad la está nombrando. El proceso de la autenticación es la verificación que el cliente es el que está nombrado en una petición. ¿Qué un nombre consiste en? En el Kerberos, nombran a los usuarios y los servidores. Por lo que al servidor de autenticación, son equivalente. Un nombre consiste en un nombre principal, un caso, y un reino, expresado como name.instance@realm.

El nombre principal es el nombre del usuario o del servicio. El caso se utiliza para distinguir entre las variaciones en el nombre principal. Para los usuarios, un caso puede exigir los privilegios especiales, tales como los casos de la “raíz” o “admin”. Para los servicios en el entorno Athena, el caso es generalmente el nombre de la máquina en la cual el servidor se ejecuta. Por ejemplo, el servicio de rlogin tiene diversos casos en diversos host: rlogin.priam es el servidor de rlogin en el host denominado priam. Un boleto del Kerberos es solamente bueno para un solo servidor designado. Como tal, un boleto distinto se requiere acceder a diversos casos del mismo servicio. El reino es el nombre de una entidad administrativa que mantenga los datos de autenticación. Por ejemplo, diversas instituciones pueden cada uno tener su propia máquina del Kerberos, conteniendo una diferente base de datos. Tienen diversos terrenos de Kerberos. (Los reinos se discuten más lejos en Interactionwith el otro Kerberi.)

Cómo funciona Kerberos

Esta sección describe los protocolos de autenticación de Kerberos. Como se mencionó anteriormente, modelo de autenticación Kerberos se basa en protocolo de la distribución de claves de Needham y del Schroeder. Cuando las peticiones del usuario un servicio, su identidad deben ser establecidas. Para hacer esto, un boleto se presenta al servidor, junto con la prueba que el boleto fue publicado al usuario, no robado originalmente. Hay tres fases a la autenticación con el Kerberos. En la primera fase, el usuario obtiene las credenciales que se utilizarán para pedir el acceso a los otros servicios. En la segunda fase, la autenticación de las peticiones del usuario para un servicio específico. En la fase final, el usuario presenta el servidor de esas credenciales al final.

Credenciales de Kerberos

Hay dos tipos de credencial usados en modelo de autenticación Kerberos: boletos y authenticators. Ambos se basan en la encripción de claves privadas, pero se cifran usando diversas claves. Un boleto se utiliza para pasar con seguridad la identidad de la persona a quien el boleto fue publicado entre el servidor de autenticación y el servidor extremo. Un boleto también pasa la información que se puede utilizar para aseegurarse que la persona que usa el boleto es la misma persona a quien fue publicado. El authenticator contiene la información adicional que, cuando está comparada contra eso en el boleto prueba que el cliente que presenta el boleto es el mismo al cual el boleto fue publicado.

Un boleto es bueno para un servidor único y un solo cliente. Contiene el nombre del servidor, el nombre del cliente, la dirección de Internet del cliente, un grupo fecha/hora, un curso de la vida, y una clave de sesión aleatoria. Esta información se cifra usando la clave del servidor para el cual el boleto será utilizado. Una vez que se ha publicado el boleto, puede ser utilizado las épocas múltiples por el cliente Nombrado de acceder al servidor designado, hasta que expire el boleto. Observe que porque el boleto se cifra en la clave del servidor, es seguro permitir que el usuario pase el boleto encendido al servidor sin tener que preocuparse del usuario que modifica el boleto.

A diferencia del boleto, el authenticator se puede utilizar solamente una vez. Un nuevo debe ser generado cada vez que un cliente quiere utilizar un servicio. Esto no presenta un problema

porque el cliente puede construir el authenticator sí mismo. Un authenticator contiene el nombre del cliente, de la dirección IP del puesto de trabajo, y del tiempo actual de estación de trabajo. El authenticator se cifra en la clave de la sesión que es parte del boleto.

Consiga el ticket inicial de Kerberos

Cuando el usuario se acerca hasta un puesto de trabajo, sólo la una pieza de la información puede probar su identidad: la contraseña de usuario. El intercambio inicial con el servidor de autenticación se diseña para minimizar la ocasión que la contraseña será comprometida, mientras que al mismo tiempo no permite que un usuario autentique correctamente la/sí mismo sin el conocimiento de esa contraseña. El proceso de inicio de sesión aparece al usuario ser lo mismo que abriendo una sesión a un sistema de tiempo compartido. Detrás de las escenas, aunque, es muy diferente.

Indican al usuario para su nombre de usuario. Una vez que se ha ingresado, una petición se envía al servidor de autenticación que contiene el nombre de usuario y el nombre de un servicio especial conocido como el servicio de distribución de tickets.

Los controles del servidor de autenticación que saben sobre el cliente. Si es así genera una clave de sesión aleatoria que sea utilizada más adelante entre el cliente y el servidor de distribución de tickets. Entonces crea un boleto para el servidor de distribución de tickets que contiene el nombre del cliente, el nombre del servidor de distribución de tickets, la hora actual, un curso de la vida para el boleto, la dirección IP del cliente, y la clave de sesión aleatoria apenas creada. Esto se cifra todo en una clave sabida solamente al servidor de distribución de tickets y al servidor de autenticación.

El servidor de autenticación entonces envía el boleto, junto con una copia de la clave de sesión aleatoria y de una cierta información adicional, de nuevo al cliente. Esta respuesta se cifra en la clave privada del cliente, sabida solamente al Kerberos y al cliente, que se deriva de la contraseña de usuario.

Una vez que la respuesta ha sido recibida por el cliente, piden el usuario su contraseña. La contraseña se convierte a una clave DES y se utiliza para desencriptar la respuesta del servidor de autenticación. El boleto y la clave de la sesión, junto con algo de la otra información, se salvan para uso futuro, y la contraseña de usuario y la clave DES se borran de la memoria.

El intercambio se ha completado una vez, el puesto de trabajo posee la información que puede utilizar para probar la identidad de su usuario para el curso de la vida del ticket distribuido. Mientras el software en el puesto de trabajo no hubiera sido tratado de forzar previamente con, ninguna información existe que permitirá que algún otro personifique usuario más allá la vida del boleto.

Pida un servicio Kerberos

Por el momento, déjenos fingen que el usuario tiene ya un boleto para el servidor deseado. Para acceder al servidor, la aplicación construye un authenticator que contiene el nombre y la dirección IP del cliente, y la hora actual. El authenticator entonces se cifra en la clave de la sesión que fue recibida con el boleto para el servidor. El cliente entonces envía el authenticator junto con el boleto al servidor de una forma definido por la aplicación individual.

Una vez que el authenticator y el boleto han sido recibidos por el servidor, el servidor desencripta el boleto, utiliza la clave de la sesión incluida en el boleto para desencriptar el authenticator,

compara la información en el boleto con ésa en el authenticator, la dirección IP de los cuales la petición fue recibida, y la actualidad. Si todo hace juego, permite la petición de proceder.

Se asume que los relojes están sincronizados dentro de varios minutos. Si el tiempo en la petición es demasiado lejano en el futuro o el pasado, el servidor trata la petición como tentativa de jugar de nuevo un pedido anterior. El servidor también se permite no perder de vista todos las peticiones del pasado con los grupos fecha/hora que son todavía válidos. Para foil más lejos los ataques con paquetes copiados, una petición recibida con el mismo boleto y el grupo fecha/hora que uno recibido ya puede ser desechado.

Finalmente, si el cliente especifica que quisiera que el servidor probara su identidad también, el servidor agrega uno al grupo fecha/hora el cliente enviado en el authenticator, cifra el resultado en la clave de la sesión, y envía el resultado de nuevo al cliente.

En el final de este intercambio, el servidor está seguro que, según el Kerberos, el cliente es quién lo dice es. Si ocurre la autenticación recíproca, convencen el cliente también de que el servidor es auténtico. Por otra parte, la parte del cliente y servidor una clave que nadie más conozca, y puede asumir con seguridad que razonablemente un mensaje reciente cifrado en esa clave originó con el otro partido.

Consiga los ticketes del servidor Kerberos

Recuerde que un boleto es solamente bueno para un servidor único. Como tal, es necesario obtener un boleto distinto para cada servicio que el cliente quiere utilizar. Los boletos para los servidores individuales se pueden obtener del servicio de distribución de tickets. Puesto que el servicio de distribución de tickets es sí mismo un servicio, hace uso del protocolo de acceso del servicio descrito en la sección anterior.

Cuando un programa requiere un boleto que no se ha pedido ya, envía una petición al servidor de distribución de tickets. La petición contiene el nombre del servidor para el cual se pide un boleto, junto con el ticket distribuido y de un authenticator construidos según lo descrito en la sección anterior.

El servidor de distribución de tickets entonces marca el authenticator y el ticket distribuido como se describe anteriormente. Si es válido, el servidor de distribución de tickets genera una nueva clave de sesión aleatoria que se utilizará entre el cliente y el nuevo servidor. Entonces construye un boleto para el nuevo servidor que contiene el nombre del cliente, Nombre del servidor, la hora actual, la dirección IP del cliente y la nueva clave de la sesión que acaba de generar. El curso de la vida del nuevo boleto es el mínimo del tiempo de vida restante para el ticket distribuido y del valor por defecto para el servicio.

El servidor de distribución de tickets entonces envía el boleto, junto con la clave de la sesión y la otra información, de nuevo al cliente. Esta vez, sin embargo, la contestación se cifra en la clave de la sesión que era parte del ticket distribuido. Esta manera, allí no es ninguna necesidad del usuario de ingresar su contraseña otra vez.

La base de datos Kerberos

Hasta esta punta, hemos discutido las operaciones que requerían acceso de sólo lectura a la base de datos Kerberos. Estas operaciones son realizadas por el servicio de autenticación, que puede hacer funcionar en ambos las máquinas del master y del esclavo.

En esta sección, discutimos las operaciones que requieren el acceso de escritura a la base de datos. Estas operaciones son realizadas por el servicio de la administración, llamado el Kerberos Database Management Service (KDBM). La implementación actual estipula que los cambios se pueden realizar solamente a la base de datos principal de Kerberos; las copias auxiliares son solo lecturas. Por lo tanto, el servidor KDBM puede ejecutarse solamente en la máquina de Kerberos principal.

Observe que, mientras que la autenticación puede todavía ocurrir (en los esclavos), las peticiones de administración no pueden ser mantenidas si la máquina principal está abajo. En nuestra experiencia, esto no ha presentado un problema, pues las peticiones de administración son infrecuentes.

El KDBM maneja las peticiones de los usuarios de cambiar sus contraseñas. El lado del cliente de este programa, que envía las peticiones al KDBM sobre la red, es el programa kpasswd. El KDBM también valida las peticiones de los administradores de Kerberos, que pueden agregar los principales a la base de datos, así como cambia las contraseñas para los mandantes existentes. El lado del cliente del programa de la administración, que también envía las peticiones al KDBM sobre la red, es el programa Kadmin.

El servidor KDBM

El servidor KDBM valida las peticiones de agregar los principales a la base de datos o de cambiar las contraseñas para los mandantes existentes. Este servicio es único en que el servicio de distribución de tickets no publicará los boletos para él. En lugar, el servicio de autenticación sí mismo debe ser utilizado (el mismo servicio que se utiliza para conseguir un ticket distribuido). El propósito de esto es requerir al usuario ingresar una contraseña. Si esto no estaba así pues, después si un usuario salió de su puesto de trabajo desatendido, un transeúnte podría recorrer para arriba y cambiar su contraseña para ellos, algo que debe ser prevenido. Asimismo, si un administrador salió de su puesto de trabajo sin vigilar, un transeúnte podría cambiar cualquier contraseña en el sistema.

Cuando el servidor KDBM recibe una petición, la autoriza comparando el nombre principal autenticado del solicitante del cambio al nombre principal de la blanco de la petición. Si son lo mismo, se permite la petición. Si no son lo mismo, el servidor KDBM consulta un Access Control List (salvado en un archivo en el sistema Kerberos principal). Si el nombre principal del solicitante se encuentra en este archivo, se permite la petición, si no se niega.

Por el convenio, los nombres con un caso NULO (el caso predeterminado) no aparecen en el archivo del Access Control List; en lugar, se utiliza una instancia admin. Por lo tanto, para que un usuario haga administrador del Kerberos a la instancia admin para ese nombre de usuario se debe crear, y agregar al Access Control List. Este convenio permite que un administrador utilice una diversa contraseña para la administración de Kerberos entonces que él utilizaría para el login normal.

Todas las peticiones al programa KDBM, están permitidas o negadas, se registran.

Programas kadmin y kpasswd

Los administradores del Kerberos utilizan el programa Kadmin para agregar los principales a la base de datos, o cambian las contraseñas de los mandantes existentes. Requieren a un administrador ingresar la contraseña para su nombre de la instancia admin cuando invocan el

programa Kadmin. Esta contraseña se utiliza para traer un boleto para el servidor KDBM.

Los usuarios pueden cambiar sus contraseñas de Kerberos usando el programa kpasswd. Los requieren ingresar su contraseña anterior cuando invocan el programa. Esta contraseña se utiliza para traer un boleto para el servidor KDBM.

Réplica de base de datos Kerberos

Cada terreno de Kerberos tiene una máquina de Kerberos principal, que contiene la copia original de la base de datos de autenticación. Es posible (aunque sea no necesario) tener adicional, las copias de sólo lectura de la base de datos en las máquinas auxiliares a otra parte en el sistema. Las ventajas del tener copias múltiples de la base de datos son ésas citada generalmente para la replicación: más alta disponibilidad y mejor rendimiento. Si la máquina principal está abajo, la autenticación se puede todavía alcanzar en una de las máquinas auxiliares. La capacidad de realizar la autenticación en de varias máquinas reduce la probabilidad de un embotellamiento en la máquina principal.

La custodia de las copias múltiples de la base de datos introduce el problema de la consistencia de los datos. Hemos encontrado que mismo los métodos simples son suficientes para ocuparse de la inconsistencia. La base de datos maestra se vacia cada hora. La base de datos se envía, en su totalidad, a las máquinas auxiliares, que entonces ponen al día sus propias bases de datos. Un programa sobre el host maestro, llamado kprop, envía la actualización a un programa del par, llamado kpropd, ejecutándose en cada uno de las máquinas auxiliares. El primer kprop envía una suma de comprobación de la nueva base de datos que es alrededor enviar. La suma de comprobación se cifra en la clave de la base de datos maestra del Kerberos, que ambas las máquinas del Kerberos del master y del esclavo poseen. Los datos entonces se transfieren sobre la red al kpropd en la máquina auxiliar. El servidor auxiliar de la propagación calcula una suma de comprobación de los datos que ha recibido, y si hace juego la suma de comprobación enviada por el master, la nueva información se utiliza para poner al día la base de datos del esclavo.

Todas las contraseñas en la base de datos Kerberos se cifran en la clave de la base de datos maestra por lo tanto, la información pasajera del master para esclavizar sobre la red no son útiles a un eavesdropper. Sin embargo, es esencial que solamente la información del host maestro sea validada por los esclavos, y que el tratar de forzar de los datos esté detectado, así la suma de comprobación.

Kerberos desde una perspectiva exterior

Esta sección describe el Kerberos desde el punto de vista práctico, primero según lo considerado por el usuario, después del punto de vista del programador de la aplicación, y finalmente, con las tareas del administrador de Kerberos.

Vista del usuario de Kerberos

Si va todo bien, el usuario notará apenas que el Kerberos está presente. En nuestra implementación de UNIX, el ticket distribuido se obtiene del Kerberos como parte del proceso de ingreso. El cambio de la contraseña de Kerberos de un usuario es parte del programa del passwd. Y los boletos del Kerberos se destruyen automáticamente cuando un usuario termina la sesión.

Si la sesión de conexión al sistema del usuario dura más de largo que el curso de la vida del ticket distribuido (actualmente 8 horas), el usuario notará la presencia del Kerberos porque la próxima

vez que se ejecuta una aplicación Kerberos-autenticada, fallará. El boleto del Kerberos para él habrá expirado. En ese momento, el usuario puede funcionar con el programa del kinit para obtener un nuevo boleto para el servidor de distribución de tickets. Como al abrir una sesión, una contraseña se debe proporcionar para conseguirlo. Un usuario que ejecuta el comando klist fuera de la curiosidad puede ser sorprendido en todos los boletos cuáles se han obtenido silenciosamente en nombre su para los servicios cuáles requieren la autenticación de Kerberos.

Kerberos desde el punto de vista del programador

Un programador que escribe una aplicación Kerberos agregará a menudo la autenticación ya a una aplicación de la red existente que consiste en un lado del cliente y servidor. Llamamos este “Kerberizing de proceso” un programa. Kerberizing implica generalmente el hacer de una llamada a la biblioteca Kerberos para realizar la autenticación en la Solicitud inicial para el servicio. Puede también implicar las llamadas a la biblioteca DES para cifrar los mensajes y los datos que se envían posteriormente entre el cliente de la aplicación y el servidor de aplicaciones.

Las funciones de biblioteca más de uso general son krb_mk_req en el lado del cliente, y krb_rd_req en el lado del servidor. La rutina del krb_mk_req toma como parámetros el nombre, el caso, y el reino del servidor de destino, que será pedido, y una suma de comprobación de los datos que se enviarán posiblemente. El cliente entonces envía el mensaje vuelto por la llamada del krb_mk_req sobre la red al lado del servidor de la aplicación. Cuando el servidor recibe este mensaje, hace una llamada al krb_rd_req de la rutina de la biblioteca. La rutina vuelve un juicio sobre la autenticidad de la identidad alegada del remitente.

Si la aplicación requiere que los mensajes enviados entre el cliente y servidor sean secretos, después las llamadas de la biblioteca se pueden hacer al krb_mk_priv (krb_rd_priv) para cifrar los mensajes (del decrypt) en la clave de la sesión que los ambos lados ahora comparten.

La tarea del administrador de Kerberos

El trabajo del administrador de Kerberos comienza con funcionar con un programa para inicializar la base de datos. Otro programa se debe funcionar con para registrar los principales esenciales en la base de datos, tal como el nombre del administrador de Kerberos con una instancia admin. El servidor de la autenticación de Kerberos y el servidor de la administración se deben poner en marcha. Si hay bases de datos auxiliares, el administrador debe arreglar que los programas para propagar las actualizaciones de base de datos del master a los esclavos estén golpeados con el pie apagado periódicamente.

Después de que se hayan tomado estos pasos iniciales, el administrador manipula la base de datos sobre la red, usando el programa Kadmin. Con ese programa, los nuevos principales pueden ser agregados, y las contraseñas pueden ser cambiadas.

Particularmente, cuando una nueva aplicación Kerberos se agrega al sistema, el administrador de Kerberos debe tomar algunas medidas para conseguirlo que trabaja. El servidor se debe registrar en la base de datos, y asignó una clave privada (esto es generalmente una clave al azar automáticamente generada). Entonces, un ciertos datos (clave incluyendo del servidor) se deben extraer de la base de datos y instalar en un archivo en la máquina del servidor. El archivo predeterminado es /etc/srvtab. La rutina de la biblioteca del krb_rd_req llamada por el servidor (véase la sección anterior) utiliza la información en ese archivo para desencriptar los mensajes enviados cifrados en la clave privada del servidor. El archivo de /etc/srvtab autentica el servidor mientras que una contraseña tecleada en una terminal autentica al usuario.

El administrador de Kerberos debe también asegurarse de que las máquinas del Kerberos sean físicamente seguras, y también sería sabio mantener los respaldos de la base de datos maestra.

Descripción detallada de Kerberos

En esta sección, describimos cómo los ajustes del Kerberos en el entorno Athena, incluyendo su uso por otros servicios de red y aplicaciones, y cómo obra recíprocamente con los terrenos de Kerberos remotos. Para una más descripción completa del entorno Athena, vea por favor a G.W. Treese.

Utilización de Kerberos de otros servicios de red

Se han modificado varias aplicaciones de red para utilizar el Kerberos. Los comandos rlogin and rsh primero intentan autenticar usando el Kerberos. Un usuario con los boletos válidos del Kerberos puede rlogin a otra máquina de Athena sin tener que configurar los archivos del .rhosts. Si la autenticación de Kerberos falla, los programas bajan en sus métodos usuales de autorización, en este caso, los archivos del .rhosts.

Hemos modificado el protocolo Post Office Protocol para utilizar el Kerberos para los usuarios de autenticidad que desean extraer su correo electrónico del “Post Office.” Un programa de la entrega de mensajes, llamado Zephyr, ha sido recientemente desarrollado en Athena, y utiliza el Kerberos para la autenticación también.

El programa para firmar para arriba a los usuarios nuevos, llamado registro, utiliza el Service Management System (SMS) y el Kerberos. Del SMS, determina si la información ingresada por el nuevo usuario Athena supuesto, tal como nombre y número de identificación MIT, es válida. Entonces marca con el Kerberos para ver si el nombre de usuario pedido es único. Si va todo bien, una nueva entrada se hace a la base de datos Kerberos, conteniendo el nombre de usuario y contraseña.

Para una explicación detallada del uso del Kerberos de asegurar el sistema de archivos de red de Sun, refiera por favor al apéndice.

Interacción con otro Kerberi

Se espera que diversas organizaciones administrativas quieran utilizar el Kerberos para la autenticación de usuario. También se espera que en muchos casos, los usuarios en una organización quieran utilizar los servicios en otra. Dominios administrativos del Kerberos admite múltiple. La especificación de los nombres en el Kerberos incluye un campo llamado el reino. Este campo contiene el nombre del dominio administrativo dentro del cual el usuario debe ser autenticado.

Registran en un solo reino y validarán solamente a los servicios generalmente las credenciales publicadas por un servidor de autenticación para ese reino. Registran a un usuario generalmente en un solo reino (el terreno local), pero es posible para que ella/él obtenga las credenciales publicadas por otro reino (el terreno remoto), basándose en la autenticación proporcionada por el terreno local. Las credenciales válidas en un terreno remoto indican el reino en el cual autenticaron al usuario originalmente. Los servicios en el terreno remoto pueden elegir si honrar esas credenciales, dependiendo del grado de seguridad requerido y del nivel de confianza en el reino que autenticó inicialmente al usuario.

Para realizar la autenticación del cruz-reino, es necesario que los administradores de cada par de reinos seleccionan una clave para ser compartidos entre sus reinos. Un usuario en el terreno local puede entonces pedir un ticket distribuido del servidor de autenticación local para el servidor de distribución de tickets en el terreno remoto. Cuando se utiliza ese boleto, el servidor de distribución de tickets remoto reconoce que la petición no es de su propio reino, y utiliza la clave previamente intercambiada para desencriptar el ticket distribuido. Entonces publica un boleto como normalmente, salvo que el campo del reino para el cliente contiene el nombre del reino en el cual autenticaron al cliente originalmente.

Este acercamiento se podía ampliar para permitir que uno se autentique con una serie de terrenos hasta alcanzar el reino con el servicio deseado. Para hacer el, aunque, sería necesario registrar la trayectoria entera que fue tomada, y no apenas el nombre del reino inicial en el cual autenticaron al usuario. En una situación semejante, todo que es sabido por el servidor es que A dice que B dice que el C dice que el usuario es fulano. Esta declaración puede ser confiada en solamente si todo el mundo a lo largo de la trayectoria también se confía en.

Problemas de Kerberos y problemas abiertos

Hay varios temas y problemas abiertos asociados al mecanismo de autenticación de Kerberos. Entre los problemas sea cómo decidir el curso de la vida correcto para un boleto, cómo permitir los proxys, y cómo garantizar la integridad de la estación de trabajo.

El problema del curso de la vida del boleto es una cuestión de elegir el balance adecuado entre la Seguridad y la conveniencia. Si la vida de un boleto es larga, después si se roban o se colocan mal un boleto y su clave de la sesión asociada, pueden ser utilizados por un período de tiempo más largo. Tal información puede ser robada si un usuario olvida terminar la sesión de una estación de trabajo pública. Alternativamente, si han autenticado a un usuario en un sistema que permite a los usuarios múltiples, otro usuario con el acceso a arraigar pudo poder encontrar la información necesitada para utilizar los boletos robados. El problema con el donante un boleto de un curso de la vida corto, sin embargo, es que cuando expira, el usuario tendrá que obtener un nuevo que requiera al usuario ingresar la contraseña otra vez.

Un problema sin resolver es problema de proxy. ¿Cómo puede un usuario autenticado permitir que un servidor adquiera otros servicios de red en nombre su? Un ejemplo donde estaría importante esto es el uso de un servicio que acceda a los archivos protegidos directamente de un fileserver. Otro ejemplo de este problema es lo que llamamos reenvío de autenticación. Si registran en un puesto de trabajo y abre una sesión a un usuario a un host remoto, sería agradable si el usuario tenía acceso a los mismos servicios disponibles localmente, mientras que funciona con un programa sobre el host remoto. Qué hace este difícil es que el usuario no pudo confiar en el host remoto, así el reenvío de autenticación no es deseable en todos los casos. No tenemos actualmente una solución a este problema.

Otro problema, y uno que es importante en el entorno Athena, es cómo garantizar la integridad del software que se ejecuta en un puesto de trabajo. Esto no está tanto de un problema en las estaciones de trabajo privadas puesto que el usuario que lo utilizará tiene control sobre él. En las estaciones de trabajo pública, sin embargo, alguien pudo haber venido adelante y haber modificado el programa del login para salvar la contraseña de usuario. La única solución disponible en nuestro entorno es actualmente hacerla difícil para que la gente modifique el software que se ejecuta en las estaciones de trabajo pública. Una mejor solución requeriría que la clave del usuario nunca deje un sistema que el usuario conozca pueda ser confiado en. Una manera que esto podría ser hecha sería si el usuario poseyó una tarjeta inteligente capaz de hacer los cifrados requeridos en el protocolo de autenticación.

Estado de Kerberos

Una versión prototipo del Kerberos entró la producción en septiembre de 1986. Desde enero de 1987, el Kerberos ha sido los únicos medios del proyecto Athena de autenticar sus 5,000 usuarios, 650 puestos de trabajo, y 65 servidores. Además, el Kerberos ahora se está utilizando en lugar de los archivos del .rhosts para el acceso que controla en varios de los sistemas de tiempo compartido de Athena.

Reconocimientos de Kerberos

El Kerberos fue diseñado inicialmente por Steve Miller y Clifford Neuman con las sugerencias de Jeff Schiller y de Jerry Saltzer. Desde entonces, numeroso otras personas han estado implicadas con el proyecto. Entre ellas son 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, Ted T'so, Win Treese, y Stan Zanarotti.

Somos agradecidos a Dan Geer, a Kathy Lieben, a Josh Lubarr, a Ken Raeburn, a Jerry Saltzer, Ed Steiner, Robert van a Renesse, y a Win Treese cuyas sugerencias mejoraron mucho proyectos anteriores de este papel.

Jedlinsky, J.T. Kohl, y W.E. Sommerfeld, “el sistema de notificación Zephyr,” en las actas de conferencia del Usenix (Winter, 1988).

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

R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh, y B. Lyon, “diseño y implementación del Sistema de archivo de red de Sun,” en las actas de conferencia del Usenix (Summer, 1985).

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

Un componente crucial del sistema de estación de trabajo del proyecto Athena es la interposición de la red entre el puesto de trabajo y su almacenamiento de archivo privado (directorio de inicio) del usuario. Todo el almacenamiento privado reside en un conjunto de los ordenadores (actualmente VAX 11/750s) que se dedican a este propósito. Esto permite que ofrezcamos a servicios encendido público - las estaciones de trabajo Unix disponibles. Cuando un usuario abre una sesión a uno de éstos público - los puestos de trabajo disponibles, bastante entonces validan su nombre y contraseña contra localmente un archivo de contraseña del residente, utilizamos el Kerberos para determinar su autenticidad. El programa del login indica para un nombre de usuario (como en cualquier sistema Unix). Este nombre de usuario se utiliza para traer un ticket distribuido del Kerberos. El programa del login utiliza la contraseña para generar una clave DES para desencriptar el boleto. Si el desciframiento es acertado, el directorio de inicio del usuario es situado consultando el servicio de asignación de nombres de Hesiod y montado con el NFS. El programa del login entonces da la vuelta al control al shell del usuario, que entonces puede funcionar con los archivos tradicionales de la personalización por usuario porque el directorio de inicio ahora “se asocia” al puesto de trabajo. El servicio de Hesiod también se utiliza para construir una entrada en el archivo de la contraseña local. (Esto está en.beneficio de los programas que miran para arriba la información en /etc/passwd.)

De varias opciones para la entrega del servicio de archivo remoto, elegimos el sistema de archivos de red de Sun. Sin embargo este sistema no puede enredar con nuestras necesidades en una manera crucial. El NFS asume que todos los puestos de trabajo entran en dos categorías (según lo visto desde el punto de vista de un servidor de archivos): confiado en y untrusted. Los sistemas no confiables no pueden acceder ninguna archivos en absoluto, confiado en pueden. Los sistemas confiables se confían en totalmente. Se asume que un sistema confiable es manejado por la administración amigable. Específicamente, es posible de un puesto de trabajo de confianza disfrazarse pues cualquier usuario válido del sistema del servicio de archivo y accede así a apenas sobre cada archivo en el sistema. (Solamente los archivos poseídos por la “raíz” se eximen.)

En nuestro entorno, la Administración de un puesto de trabajo (en el sentido tradicional de la Administración de sistema Unix) está en las manos del usuario actualmente usando ella. No hacemos ningún secreto de la contraseña de raíz en nuestros puestos de trabajo, pues realizamos que un usuario extraño puede romperse verdad adentro por el mismo hecho de que él se está sentando en la misma ubicación física que la máquina y tiene acceso a todas las funciones de la consola. Por lo tanto no podemos confiar en verdad nuestros puestos de trabajo en la interpretación NFS de la confianza. Para no prohibir a los controles de acceso apropiados en nuestro entorno nos tuvimos que hacer algunas modificaciones al software de la base NFS, e integramos el Kerberos en el esquema.

NFS no modificado de Kerberos

En la implementación del NFS con la cual comenzamos (de la Universidad de Wisconsin), la autenticación fue proporcionada bajo la forma de pedazo de datos incluidos en cada petición NFS (llamada los “credenciales” en terminología NFS). Estos credenciales contienen la información sobre la Identificación de usuario única (UID) del solicitante y una lista de los identificadores del grupo (GIDs) de la calidad de miembro del solicitante. Esta información entonces es utilizada por el servidor NFS para la verificación de acceso. La diferencia entre un puesto de trabajo de confianza y no confiable es independientemente de si sus credenciales son validadas por el servidor NFS.

NFS modificado por Kerberos

En nuestro entorno, los servidores NFS deben validar las credenciales de un puesto de trabajo si y solamente si las credenciales indican el UID del usuario del puesto de trabajo, y de no otro.

Una solución evidente sería cambiar naturaleza de las credenciales de los simples indicadores del UID y del GIDs a las verdaderas informaciones autenticadas de Kerberos. Sin embargo una multa de rendimiento significativa sería pagada si esta solución fue adoptada. Las credenciales se intercambian en cada funcionamiento de NFS incluyendo todo el disco leído y escriben las actividades. Incluyendo una autenticación de Kerberos en cada transacción del disco agregaría un número justo de verdaderos cifrados (hechos en el software) por la transacción y, según nuestros cálculos de la envolvente, habría entregado el rendimiento inaceptable. (También habría requerido poner las rutinas de biblioteca Kerberos en el espacio de la dirección del corazón.)

Necesitamos un acercamiento híbrido, descrito más abajo. La idea básica es tener los credenciales del mapa del servidor NFS recibidos de las estaciones de trabajo del cliente, a los credenciales válidos (y posiblemente diversos) en el sistema del servidor. Esta asignación se realiza en el corazón del servidor en cada transacción NFS y es puesta en el tiempo del “soporte” por un proceso del nivel de usuario que enganche a la autenticación moderada de Kerberos antes

de establecer un mapeo de credencial del kernel válido.

Para implementar esto agregamos una nueva llamada del sistema al corazón (requerido solamente en los sistemas del servidor, no en los sistemas del cliente) que prevé el control de la función de mapeo que las credenciales entrantes de las correspondencias de las estaciones de trabajo del cliente a las credenciales válidas para el uso en el servidor (eventualmente). La función de mapeo básica asocia el tuple:

a un credencial NFS válido en el sistema del servidor. El CLIENT-IP-ADDRESS se extrae del paquete de pedidos NFS suministrado por el sistema del cliente. Nota: toda la información en los credenciales generados por el cliente excepto el UID-ON-CLIENT se desecha.

Si existe ninguna asignación, se configura el servidor reacciona en una de dos maneras, dependiendo él. En nuestra configuración amistosa omitimos las peticiones unmappable en las credenciales para el usuario “nadie” quién no tiene ningún acceso privilegiado y tiene un UID único. Los servidores hostiles vuelven un error de acceso a NFS cuando ningún mapeo válido se puede encontrar para un credencial NFS entrante.

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

Modificamos la daemon del soporte (que maneja las peticiones de montaje NFS en los sistemas del servidor) para validar un nuevo tipo de transacción, la petición de la asignación de la autenticación de Kerberos. Básicamente, como parte del proceso del montaje, el sistema del cliente proporciona un autenticador de Kerberos junto con una indicación de su UID-ON-CLIENT (cifrado en el autenticador de Kerberos) en el puesto de trabajo. La daemon del soporte del servidor convierte el nombre de mandante identidad única de Kerberos en un nombre de usuario local. Este nombre de usuario entonces se mira para arriba en un archivo especial para rendir el UID del usuario y la lista del GIDs. Para la eficacia, este archivo es un archivo de base de datos del ndbm con el nombre de usuario como la clave. De esta información, un credencial NFS se construye y se da al corazón como el mapeo válido del <CLIENT-IP-ADDRESS, tuple CLIENT-UID> para esta petición.

En el tiempo del unmount una petición se envía a la daemon del soporte de quitar la asignación previamente agregada del corazón. Es también posible enviar una petición en el tiempo del logout para invalidar toda la asignación para el Usuario usuario actual en el servidor en la pregunta, así limpiando cualquier asignación restante que exista (ella no debe sin embargo) antes de que el puesto de trabajo se haga disponible para el usuario siguiente.

Consecuencias en la seguridad de Kerberos de NFS modificado

Esta implementación no es totalmente segura. Para empezar, los datos del usuario todavía se envían a través de la red en un unencrypted, y por lo tanto interceptable, forma. El de bajo nivel, autenticación de la por-transacción se basa en un <CLIENT-IP-ADDRESS, los pares CLIENT-UID> proporcionados unencrypted en el paquete de pedidos. Esta información podría ser forjada y la Seguridad comprometió así. Sin embargo, debe ser observado que solamente mientras que un usuario está utilizando activamente su los archivos (es decir, mientras que está abierto una sesión) son mapeos válidos en el lugar y por lo tanto esta forma de ataque están limitados a cuando abren una sesión al usuario en la pregunta. Cuando no abren una sesión a un usuario, ninguna cantidad de falsificación de la dirección IP permitirá el acceso no autorizado a su los

archivos.

Referencias de Kerberos

1. S.P. Miller, A.C. Neuman, J.I. Schiller, y J.H. Saltzer, sección E.2.1: Autenticación de Kerberos y sistema de autorización, M.I.T. Proyecto Athena, Cambridge, Massachusetts (de diciembre el 21 de 1987).
2. E. Balkovich, S.R. Lerman, y R.P. Parmelee, “computando en la educación superior: La experiencia de Athena,” comunicaciones del ACM, vol. 28(11), págs. 1214-1224, ACM (noviembre de 1985).
3. R.M. Needham y M.D. Schroeder, “usando la encripción para autenticación en las Redes grandes de las Computadoras,” comunicaciones del ACM, vol. 21(12), 993-999 págs. (diciembre de 1978).
4. V.L. Voydock y S.T. Kent, “mecanismos de seguridad en los protocolos de red de alto nivel,” Sondeos de computación, vol. 15(2), ACM (junio de 1983).
5. Oficina nacional de estándares, “Data Encryption Standard,” publicación 46 de los Estándares de procesamiento de la información federales, Oficina gubernamental de impresiones, Washington, DC (1977).
6. Tintóreo SP, “Hesiod,” en las actas de conferencia del Usenix (Winter, 1988).
7. W.J. Bryant, la guía del Programador Kerberos, MIT Project Athena (en la preparación).
8. W.J. Bryant, el manual del administrador de Kerberos, MIT Project Athena (en la preparación).
9. G.W. Treese, “Berkeley Unix en 1000 puestos de trabajo: Cambios de Athena hasta el 4.3BSD," en las actas de conferencia del Usenix (Winter, 1988).
10. C.A. DellaFera, M.W. Eichin, R.S. french, D.C. Jedlinsky, J.T. Kohl, y W.E. Sommerfeld, “el sistema de notificación Zephyr,” en las actas de conferencia del Usenix (Winter, 1988).
11. M.A. Rosenstein, D.E. Geer, y P.J. Levine, en las actas de conferencia del Usenix (Winter, 1988).
12. R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh, y B. Lyon, “diseño y implementación del Sistema de archivo de red de Sun,” en las actas de conferencia del Usenix (Summer, 1985).

Información Relacionada

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