Visão geral do Kerberos - um serviço de autenticação para sistemas de rede abertos

Contents

Introduction

Em um ambiente de computação de rede aberta, uma estação de trabalho não é confiável para identificar corretamente seus usuários para os serviços de rede. O Kerberos oferece uma abordagem alternativa em que um serviço confiável de autenticação de terceiros é usado para verificar a identidade dos usuários. Este artigo oferece uma visão geral do modelo de autenticação Kerberos como implantado para o projeto Athena do MIT. Descreve os protocolos usados por clientes, servidores e o Kerberos para conseguir a autenticação. Também descreve o gerenciamento e a duplicação da base de dados necessária. As exibições do Kerberos, como visualizadas pelo usuário, pelo programador e pelo administrador são descritas. Finalmente, o papel do Kerberos, no escopo mais amplo do projeto Athena, é apresentado, juntamente com uma lista de aplicações que atualmente usam o Kerberos para autenticação de usuários. Nós descrevemos a integração da ferramenta de autenticação Kerberos ao Sistema de Arquivo de Rede Sun como um estudo de caso para integração do Kerberos a um aplicativo existente.

Autores de Kerberos

- Jennifer G. Steiner, Project Athena, Massachusetts Institute of Technology, Cambridge, MA 02139, steiner@ATHENA.MIT.EDU
- Clifford Neuman, Department of Computer Science, FR-35, University of Washington, Seattle, WA 98195, bcn@CS.WASHINGTON.EDU. Clifford Neuman era membro da equipe do Project Athena durante a fase de projeto e implementação inicial de Kerberos.
- Jeffrey I. Schiller, Project Athena, Massachusetts Institute of Technology, Cambridge, MA 02139, jis@ATHENA.MIT.EDU

Introdução ao Kerberos

Este documento dá uma visão geral do Kerberos, um sistema de autenticação projetado por Miller e Neuman. para ambientes de computação de rede aberta e descreve nossa experiência usando-a no Project Athena do MIT. Na seção Motivação, explicamos por que um novo modelo de autenticação é necessário para redes abertas e quais são seus requisitos. O que é Kerberos? lista os componentes do software Kerberos e descreve como eles interagem no fornecimento do serviço de autenticação. Na seção Nomes de Kerberos, descrevemos o esquema de nomeação de Kerberos.

Como o Kerberos funciona apresenta os blocos componentes da autenticação Kerberos - o tíquete e o autenticador. Isso leva a uma discussão sobre os dois protocolos de autenticação: a autenticação inicial de um usuário para Kerberos (análogo ao login) e o protocolo para autenticação mútua de um consumidor em potencial e de um produtor em potencial de um serviço de rede.

Kerberos exige uma base de dados de informações sobre os seus clientes; a seção Banco de Dados Kerberos descreve o banco de dados, seu gerenciamento e o protocolo para sua modificação. A seção Kerberos From the Outside Looking In descreve a interface Kerberos para seus usuários, programadores de aplicativos e administradores. Na seção A imagem maior, descrevemos como o projeto Athena Kerberos se encaixa no resto do ambiente Athena. Também descrevemos a interação de diferentes domínios de autenticação Kerberos, ou domínios; no nosso caso, a relação entre o projeto Athena Kerberos e o Kerberos no Laboratório de Computação do MIT.

Na seção Problemas e Problemas Abertos, mencionamos problemas e problemas abertos ainda não resolvidos. A última seção mostra o status atual de Kerberos no Projeto Athena. No Apêndice, descrevemos em detalhes como o Kerberos é aplicado a um serviço de arquivos de rede para autenticar usuários que desejam obter acesso a sistemas de arquivos remotos.

Conceitos de Kerberos

Em todo este documento, usamos termos que podem ser ambíguos, novos para o leitor ou diferentes dos outros. Abaixo, indicamos o uso desses termos.

Usuário, Cliente, Servidor — Por usuário, queremos dizer um ser humano que usa um programa ou serviço. Um cliente também usa algo, mas não é necessariamente uma pessoa. pode ser um programa. Frequentemente, os aplicativos de rede consistem em duas partes; um programa que é executado em uma máquina e solicita um serviço remoto, e outro programa que é executado na máquina remota e executa esse serviço. Chamamos esses usuários do lado do cliente e do lado do servidor do aplicativo, respectivamente. Frequentemente, um cliente entrará em contato com um servidor em nome de um usuário.

Cada entidade que usa o sistema Kerberos, seja um usuário ou um servidor de rede, é, de certa forma, um cliente, já que usa o serviço Kerberos. Assim, para distinguir clientes Kerberos de clientes de outros serviços, usamos o termo principal para indicar tal entidade. Observe que um principal Kerberos pode ser um usuário ou um servidor. (Descrevemos o nome dos principais Kerberos em uma seção posterior.)

Serviço versus Servidor — Usamos o serviço como uma especificação abstrata de algumas ações a serem executadas. Um processo que executa essas ações é chamado de servidor. Em um determinado momento, pode haver vários servidores (geralmente em execução em diferentes máquinas) executando um determinado serviço. Por exemplo, no Athena, há um servidor de login UNIX BSD em execução em cada uma de nossas máquinas de compartilhamento de tempo.

Chave, Chave privada, Senha — Kerberos usa criptografia de chave privada. A cada principal do Kerberos é atribuído um grande número, sua chave privada, conhecida apenas por esse principal e Kerberos. No caso de um usuário, a chave privada é o resultado de uma função unidirecional aplicada à senha do usuário. Usamos a chave como abreviação para a chave privada.

Credenciais — Infelizmente, essa palavra tem um significado especial para o Sistema de arquivos de rede Sun e para o sistema Kerberos. Nós afirmamos explicitamente se queremos dizer credenciais NFS ou Kerberos, caso contrário, o termo é usado no sentido normal do idioma inglês.

Mestre e escravo — É possível executar o software de autenticação Kerberos em mais de uma máquina. No entanto, há sempre apenas uma cópia definitiva do banco de dados Kerberos. A máquina que abriga esse banco de dados é chamada de máquina-mestre, ou apenas de mestre. Outras máquinas podem possuir cópias somente leitura do banco de dados Kerberos, chamadas de escravos.

Motivação além de kerberos

Em um ambiente de computação pessoal não conectado em rede, os recursos e as informações podem ser protegidos com a proteção física do computador pessoal. Em um ambiente de computação de compartilhamento de tempo, o sistema operacional protege os usuários uns dos

outros e controla os recursos. Para determinar o que cada usuário pode ler ou modificar, é necessário que o sistema de compartilhamento de tempo identifique cada usuário. Isso é feito quando o usuário faz login.

Em uma rede de usuários que necessitam de serviços de vários computadores separados, há três abordagens que podem ser seguidas para o controle de acesso: Não se pode fazer nada, dependendo da máquina na qual o usuário está conectado para impedir o acesso não autorizado; pode-se exigir que o anfitrião prove sua identidade, mas confiar na palavra do anfitrião sobre quem é o utilizador; ou pode exigir que o usuário prove sua identidade para cada serviço necessário.

Em um ambiente fechado onde todas as máquinas estão sob controle rígido, pode-se usar a primeira abordagem. Quando a organização controla todos os hosts que se comunicam pela rede, essa é uma abordagem razoável.

Em um ambiente mais aberto, é possível confiar seletivamente apenas nos hosts sob controle organizacional. Nesse caso, cada host deve ser obrigado a provar sua identidade. Os programas rlogin e rsh usam essa abordagem. Nesses protocolos, a autenticação é feita verificando-se o endereço de Internet do qual uma conexão foi estabelecida.

No ambiente Athena, devemos ser capazes de honrar solicitações de hosts que não estão sob controle organizacional. Os usuários têm controle total de suas estações de trabalho: eles podem reinicializá-los, trazê-los sozinhos ou até mesmo inicializar suas próprias fitas. Assim sendo, deve ser adotada a terceira abordagem; o usuário deve provar sua identidade para cada serviço desejado. O servidor também deve provar sua identidade. Não é suficiente proteger fisicamente o host que executa um servidor de rede; alguém em outro lugar da rede pode estar se mascarando como o servidor especificado.

Nosso ambiente impõe vários requisitos em um mecanismo de identificação. Primeiro, deve ser seguro. Evitar isso deve ser difícil o suficiente para que um invasor em potencial não ache o mecanismo de autenticação como o link fraco. Alguém que esteja observando a rede não deve conseguir obter as informações necessárias para representar outro usuário. Segundo, deve ser confiável. O acesso a muitos serviços dependerá do serviço de autenticação. Se não for fiável, o sistema de serviços no seu todo não será. Em terceiro lugar, deve ser transparente. O ideal é que o usuário não esteja ciente da autenticação que está ocorrendo. Finalmente, deve ser escalável. Muitos sistemas podem se comunicar com os hosts Athena. Nem todos eles suportarão o nosso mecanismo, mas o software não deve quebrar se o fizerem.

Kerberos é o resultado do nosso trabalho para atender aos requisitos acima. Quando um usuário caminha até uma estação de trabalho, ele faz login. Até onde o usuário pode dizer, essa identificação inicial é suficiente para provar sua identidade para todos os servidores de rede necessários durante a sessão de login. A segurança do Kerberos depende da segurança de vários servidores de autenticação, mas não do sistema a partir do qual os usuários fazem login, nem da segurança dos servidores finais que serão usados. O servidor de autenticação fornece a um usuário devidamente autenticado uma maneira de provar sua identidade para servidores espalhados pela rede.

A autenticação é um elemento fundamental para um ambiente de rede seguro. Se, por exemplo, um servidor sabe com certeza a identidade de um cliente, ele pode decidir se deve fornecer o serviço, se o usuário deve receber privilégios especiais, quem deve receber a conta do serviço e assim por diante. Em outras palavras, os esquemas de autorização e contabilidade podem ser construídos sobre a autenticação fornecida por Kerberos, resultando em segurança equivalente ao computador pessoal isolado ou ao sistema de compartilhamento de tempo.

O que é Kerberos?

Kerberos é um serviço de autenticação confiável de terceiros baseado no modelo apresentado por Needham e Schroeder. Acredita-se que cada um de seus clientes acredite que o julgamento de Kerberos sobre a identidade de cada um de seus outros clientes é preciso. Carimbos de data e hora (grandes números representando a data e hora atuais) foram adicionados ao modelo original para auxiliar na detecção de repetição. A repetição ocorre quando uma mensagem é roubada da rede e reenviada posteriormente. Para obter uma descrição mais completa da repetição e de outros problemas de autenticação, consulte Voydock e Kent.

O que faz o Kerberos?

Kerberos mantém um banco de dados de seus clientes e suas chaves privadas. A chave privada é um grande número conhecido somente por Kerberos e pelo cliente ao qual ela pertence. Caso o cliente seja um usuário, é uma senha criptografada. Serviços de rede que exigem registro de autenticação com Kerberos, assim como clientes que desejam usar esses serviços. As chaves privadas são negociadas no registro.

Como Kerberos conhece essas chaves privadas, ele pode criar mensagens que convencem um cliente de que outro é realmente quem ele alega ser. O Kerberos também gera chaves privadas temporárias, chamadas de chaves de sessão, que são dadas a dois clientes e a ninguém mais. Uma chave de sessão pode ser usada para criptografar mensagens entre duas partes.

Kerberos fornece três níveis distintos de proteção. O programador de aplicativos determina qual é apropriado, de acordo com os requisitos do aplicativo. Por exemplo, alguns aplicativos exigem apenas que a autenticidade seja estabelecida no início de uma conexão de rede e podem supor que outras mensagens de um determinado endereço de rede se originem da parte autenticada. Nosso sistema de arquivos de rede autenticado usa esse nível de segurança.

Outros aplicativos exigem autenticação de cada mensagem, mas não se importam se o conteúdo da mensagem é divulgado ou não. Para isso, Kerberos fornece mensagens seguras. No entanto, um nível mais alto de segurança é fornecido por mensagens privadas, em que cada mensagem não só é autenticada, como também criptografada. Mensagens privadas são usadas, por exemplo, pelo próprio servidor Kerberos para enviar senhas pela rede.

Componentes de software kerberos

A execução do Athena compreende vários módulos:

- biblioteca de aplicativos Kerberos
- biblioteca de criptografia
- biblioteca de banco de dados
- programas de administração de bases de dados
- servidor de administração
- servidor de autenticação
- software de propagação de db
- programas de usuário
- aplicativos

A biblioteca de aplicativos Kerberos fornece uma interface para clientes de aplicativos e servidores de aplicativos. Ele contém, entre outras, rotinas para criar ou ler solicitações de

autenticação e as rotinas para criar mensagens seguras ou privadas.

A criptografia no Kerberos é baseada no DES, o Data Encryption Standard. A biblioteca de criptografia implementa essas rotinas. Vários métodos de criptografia são fornecidos, com compensações entre velocidade e segurança. Também é fornecida uma extensão para o modo CBC (Cipher Block Chaining, encadeamento de bloco de cifra DES), chamado de modo CBC de propagação. No CBC, um erro é propagado somente pelo bloco atual da cifra, enquanto no PCBC o erro é propagado por toda a mensagem. Isso tornará a mensagem inteira inútil se ocorrer um erro, em vez de apenas uma parte dela. A biblioteca de criptografia é um módulo independente e pode ser substituída por outras implementações DES ou por uma biblioteca de criptografia diferente.

Outro módulo substituível é o sistema de gerenciamento de banco de dados. A implementação atual do Athena da biblioteca de banco de dados usa ndbm, embora o Ingres tenha sido originalmente usado. Outras bibliotecas de gerenciamento de banco de dados também podem ser usadas.

As necessidades do banco de dados Kerberos são claras; um registro é mantido para cada entidade de segurança, contendo o nome, a chave privada e a data de expiração do principal, juntamente com algumas informações administrativas. (A data de expiração é a data após a qual uma entrada não é mais válida. Geralmente, ele é definido para alguns anos no futuro no registro.)

Outras informações do usuário, como nome real, número de telefone e assim por diante, são mantidas por outro servidor, o servidor de nomes Hesiod. Desta forma, as informações sensíveis, nomeadamente as senhas, podem ser tratadas por Kerberos, utilizando medidas de segurança bastante elevadas; enquanto as informações não sensíveis conservadas pelo Hesiod são tratadas de forma diferente; pode, por exemplo, ser enviado sem criptografia pela rede.

Os servidores Kerberos usam a biblioteca de banco de dados, assim como as ferramentas para administrar o banco de dados.

O servidor de administração (ou servidor KDBM) fornece uma interface de rede de leitura/gravação para o banco de dados. O lado do cliente do programa pode ser executado em qualquer máquina na rede. O lado do servidor, no entanto, deve ser executado no computador que hospeda o banco de dados Kerberos para fazer alterações no banco de dados.

O servidor de autenticação (ou servidor Kerberos), por outro lado, executa operações somente leitura no banco de dados Kerberos, ou seja, a autenticação de principais e a geração de chaves de sessão. Como esse servidor não modifica o banco de dados Kerberos, ele pode ser executado em uma máquina que armazena uma cópia somente leitura do banco de dados Kerberos mestre.

O software de propagação de banco de dados gerencia a replicação do banco de dados Kerberos. É possível ter cópias do banco de dados em várias máquinas diferentes, com uma cópia do servidor de autenticação em execução em cada máquina. Cada uma dessas máquinas escravas recebe uma atualização do banco de dados Kerberos da máquina mestre em determinados intervalos.

Finalmente, há programas de usuário final para fazer login no Kerberos, alterar uma senha do Kerberos e exibir ou destruir tíquetes do Kerberos (tíquetes explicados posteriormente).

Nomes de Kerberos

Parte da autenticação de uma entidade está nomeando-a. O processo de autenticação é a verificação de que o cliente é o nome em uma solicitação. Em que consiste um nome? No Kerberos, os usuários e os servidores são nomeados. No que diz respeito ao servidor de autenticação, eles são equivalentes. Um nome consiste em um nome principal, uma instância e um território, expresso como name.instance@realm.

O nome principal é o nome do usuário ou do serviço. A instância é usada para distinguir variações no nome principal. Para os usuários, uma instância pode envolver privilégios especiais, como as instâncias "raiz" ou "admin". Para serviços no ambiente Athena, a instância é geralmente o nome da máquina em que o servidor é executado. Por exemplo, o serviço rlogin tem instâncias diferentes em hosts diferentes: rlogin.priam é o servidor rlogin no host chamado priam. Um tíquete Kerberos só é válido para um único servidor nomeado. Como tal, um tíquete separado é necessário para obter acesso a diferentes instâncias do mesmo serviço. O território é o nome de uma entidade administrativa que mantém dados de autenticação. Por exemplo, diferentes instituições podem ter suas próprias máquinas Kerberos, abrigando um banco de dados diferente. Eles têm territórios Kerberos diferentes. (Os territórios são discutidos mais adiante em Interação com Outros Kerberi.)

Como o Kerberos funciona

Esta seção descreve os protocolos de autenticação Kerberos. Como mencionado acima, o modelo de autenticação Kerberos é baseado no protocolo de distribuição de chaves Needham e Schroeder. Quando um usuário solicita um serviço, sua identidade deve ser estabelecida. Para isso, um tíquete é apresentado ao servidor, junto com a prova de que o tíquete foi originalmente emitido para o usuário, e não roubado. Há três fases para autenticação através do Kerberos. Na primeira fase, o usuário obtém credenciais a serem usadas para solicitar acesso a outros serviços. Na segunda fase, o usuário solicita autenticação para um serviço específico. Na fase final, o usuário apresenta essas credenciais ao servidor final.

Credenciais do Kerberos

Há dois tipos de credenciais usadas no modelo de autenticação Kerberos: bilhetes e autenticadores. Ambos são baseados na criptografia de chave privada, mas são criptografados usando chaves diferentes. Um tíquete é usado para passar com segurança a identidade da pessoa para quem o tíquete foi emitido entre o servidor de autenticação e o servidor final. O bilhete também transmite informações que podem ser utilizadas para garantir que a pessoa que o utiliza é a mesma pessoa para a qual foi emitido. O autenticador contém as informações adicionais que, quando comparadas com as do bilhete, provam que o cliente que apresenta o bilhete é o mesmo que o bilhete foi emitido.

Um tíquete é bom para um único servidor e um único cliente. Contém o nome do servidor, o nome do cliente, o endereço Internet do cliente, um carimbo de data/hora, uma vida útil e uma chave de sessão aleatória. Essas informações são criptografadas usando a chave do servidor para o qual o tíquete será usado. Uma vez emitido o tíquete, ele pode ser usado várias vezes pelo cliente nomeado para obter acesso ao servidor nomeado, até que o tíquete expire. Observe que como o tíquete é criptografado na chave do servidor, é seguro permitir que o usuário passe o tíquete para o servidor sem ter que se preocupar com a modificação do tíquete.

Ao contrário do tíquete, o autenticador só pode ser usado uma vez. Um novo deve ser gerado toda vez que um cliente quiser usar um serviço. Isso não apresenta um problema porque o cliente é capaz de criar o próprio autenticador. Um autenticador contém o nome do cliente, o endereço IP

da estação de trabalho e a hora atual da estação de trabalho. O autenticador é criptografado na chave da sessão que faz parte do tíquete.

Obtenha o tíquete inicial do Kerberos

Quando o usuário caminha até uma estação de trabalho, apenas uma informação pode provar sua identidade: a senha do usuário. A troca inicial com o servidor de autenticação foi projetada para minimizar a chance de a senha ser comprometida, ao mesmo tempo em que não permite que um usuário autentique-a adequadamente sem saber dessa senha. O processo de login parece ser o mesmo que fazer login em um sistema de timesharing. Nos bastidores, porém, é bem diferente.

O usuário é solicitado a informar seu nome de usuário. Depois de inserida, uma solicitação é enviada ao servidor de autenticação contendo o nome do usuário e o nome de um serviço especial conhecido como serviço de concessão de tíquete.

O servidor de autenticação verifica se sabe sobre o cliente. Se sim, ele gera uma chave de sessão aleatória que será mais tarde usada entre o cliente e o servidor que concede tíquete. Em seguida, cria um tíquete para o servidor de concessão de tíquete que contém o nome do cliente, o nome do servidor de concessão de tíquete, a hora atual, uma vida útil do tíquete, o endereço IP do cliente e a chave de sessão aleatória recém-criada. Tudo isso é criptografado em uma chave conhecida somente pelo servidor de concessão de tíquete e pelo servidor de autenticação.

O servidor de autenticação envia o tíquete, juntamente com uma cópia da chave de sessão aleatória e algumas informações adicionais, de volta ao cliente. Essa resposta é criptografada na chave privada do cliente, conhecida somente por Kerberos e pelo cliente, derivada da senha do usuário.

Depois que a resposta for recebida pelo cliente, o usuário será solicitado a fornecer sua senha. A senha é convertida em uma chave DES e usada para descriptografar a resposta do servidor de autenticação. O tíquete e a chave da sessão, juntamente com algumas outras informações, são armazenados para uso futuro e a senha do usuário e a chave DES são apagadas da memória.

Uma vez concluída a troca, a estação de trabalho possui informações que pode usar para provar a identidade de seu usuário durante a vida útil do tíquete de concessão de tíquete. Desde que o software na estação de trabalho não tenha sido violado anteriormente, não existe nenhuma informação que permita que outra pessoa impersonalize o usuário além da vida útil do tíquete.

Solicitar um serviço Kerberos

Por enquanto, vamos fingir que o usuário já tem um tíquete para o servidor desejado. Para obter acesso ao servidor, o aplicativo cria um autenticador que contém o nome e o endereço IP do cliente e a hora atual. O autenticador é então criptografado na chave da sessão que foi recebida com o tíquete para o servidor. Em seguida, o cliente envia o autenticador junto com o tíquete para o servidor de uma maneira definida pelo aplicativo individual.

Quando o autenticador e o tíquete tiverem sido recebidos pelo servidor, o servidor descriptografa o tíquete, usa a chave da sessão incluída no tíquete para descriptografar o autenticador, compara as informações no tíquete com as do autenticador, o endereço IP do qual a solicitação foi recebida e a hora atual. Se tudo corresponder, isso permitirá que a solicitação continue.

Pressupõe-se que os relógios sejam sincronizados em vários minutos. Se o tempo na solicitação

for muito longo no futuro ou no passado, o servidor tratará a solicitação como uma tentativa de repetir uma solicitação anterior. O servidor também tem permissão para controlar todas as solicitações anteriores com marcas de hora que ainda são válidas. Para continuar a reproduzir os ataques, uma solicitação recebida com o mesmo tíquete e carimbo de data e hora que uma já recebida pode ser descartada.

Finalmente, se o cliente especificar que deseja que o servidor prove sua identidade também, o servidor adiciona um ao carimbo de data e hora que o cliente enviou no autenticador, criptografa o resultado na chave da sessão e envia o resultado de volta ao cliente.

Ao final dessa troca, o servidor tem certeza de que, segundo Kerberos, o cliente é quem diz ser. Se a autenticação mútua ocorrer, o cliente também está convencido de que o servidor é autêntico. Além disso, o cliente e o servidor compartilham uma chave que ninguém mais sabe e podem assumir com segurança que uma mensagem razoavelmente recente criptografada nessa chave foi originada pela outra parte.

Obter tíquetes de servidor Kerberos

Lembre-se de que um tíquete só é bom para um único servidor. Como tal, é necessário obter um tíquete separado para cada serviço que o cliente deseja usar. Os tíquetes para servidores individuais podem ser obtidos do serviço de concessão de tíquetes. Como o serviço de concessão de tíquete é, em si, um serviço, ele usa o protocolo de acesso ao serviço descrito na seção anterior.

Quando um programa exige um tíquete que ainda não foi solicitado, ele envia uma solicitação ao servidor de concessão de tíquete. A solicitação contém o nome do servidor para o qual um tíquete é solicitado, juntamente com o tíquete de concessão de tíquete e um autenticador criado conforme descrito na seção anterior.

O servidor que concede o tíquete verifica o autenticador e o tíquete de concessão de tíquete conforme descrito acima. Se for válido, o servidor que concede tíquete gera uma nova chave de sessão aleatória a ser usada entre o cliente e o novo servidor. Em seguida, ele cria um tíquete para o novo servidor contendo o nome do cliente, o nome do servidor, a hora atual, o endereço IP do cliente e a nova chave de sessão que ele acabou de gerar. O tempo de vida do novo tíquete é o mínimo da vida restante do tíquete de concessão de tíquete e o padrão do serviço.

O servidor que concede tíquete envia o tíquete, juntamente com a chave da sessão e outras informações, de volta ao cliente. Desta vez, no entanto, a resposta é criptografada na chave da sessão que fazia parte do tíquete de concessão de tíquete. Dessa forma, não há necessidade de o usuário digitar sua senha novamente.

O banco de dados Kerberos

Até este ponto, discutimos as operações que exigem acesso somente leitura ao banco de dados Kerberos. Essas operações são executadas pelo serviço de autenticação, que pode ser executado em máquinas mestre e escravo.

Nesta seção, discutimos as operações que exigem acesso de gravação ao banco de dados. Essas operações são executadas pelo serviço de administração, chamado de Kerberos Database Management Service (KDBM). A atual aplicação estipula que só podem ser efetuadas alterações à base de dados principal Kerberos; as cópias escravas são somente leitura. Portanto, o servidor

KDBM só pode ser executado na máquina Kerberos mestre.

Observe que, embora a autenticação ainda possa ocorrer (em escravos), as solicitações de administração não poderão ser atendidas se a máquina mestre estiver inativa. Em nossa experiência, isso não apresentou um problema, pois os pedidos de administração são raros.

O KDBM lida com solicitações dos usuários para alterar suas senhas. O lado do cliente deste programa, que envia solicitações ao KDBM pela rede, é o programa kpasswd. O KDBM também aceita solicitações de administradores Kerberos, que podem adicionar principais ao banco de dados, bem como alterar senhas para principais existentes. O lado do cliente do programa de administração, que também envia solicitações ao KDBM pela rede, é o programa kadmin.

O servidor KDBM

O servidor KDBM aceita solicitações para adicionar principals ao banco de dados ou alterar as senhas de principais existentes. Este serviço é único porque o serviço de concessão de bilhetes não emitirá bilhetes para ele. Em vez disso, o próprio serviço de autenticação deve ser usado (o mesmo serviço usado para obter um tíquete de concessão de tíquete). O objetivo disso é exigir que o usuário digite uma senha. Se isso não acontecesse, então se um usuário deixasse sua estação de trabalho sem vigilância, um transeunte poderia se aproximar e mudar sua senha para eles, algo que deveria ser evitado. Da mesma forma, se um administrador deixou sua estação de trabalho desprotegida, um transeunte poderia alterar qualquer senha no sistema.

Quando o servidor KDBM recebe uma solicitação, ele a autoriza comparando o nome principal autenticado do solicitante da alteração com o nome principal do destino da solicitação. Se forem iguais, a solicitação é permitida. Se não forem iguais, o servidor KDBM consulta uma lista de controle de acesso (armazenada em um arquivo no sistema Kerberos mestre). Se o nome principal do solicitante for encontrado neste arquivo, a solicitação será permitida, caso contrário ela será negada.

Por convenção, os nomes com uma instância NULL (a instância padrão) não aparecem no arquivo da lista de controle de acesso; em vez disso, uma instância admin é usada. Portanto, para que um usuário se torne um administrador do Kerberos, uma instância de administrador para esse nome de usuário deve ser criada e adicionada à lista de controle de acesso. Essa convenção permite que um administrador use uma senha diferente para a administração do Kerberos e, em seguida, usaria para o login normal.

Todas as solicitações ao programa KDBM, permitidas ou negadas, são registradas.

Os programas kadmin e kpasswd

Os administradores do Kerberos usam o programa kadmin para adicionar principais ao banco de dados ou alterar as senhas dos principais existentes. É necessário que um administrador digite a senha para o nome da instância do administrador ao chamar o programa kadmin. Esta senha é usada para buscar um tíquete para o servidor KDBM.

Os usuários podem alterar suas senhas Kerberos usando o programa kpasswd. É necessário que eles insiram sua senha antiga ao chamar o programa. Esta senha é usada para buscar um tíquete para o servidor KDBM.

Replicação de banco de dados kerberos

Cada território Kerberos tem uma máquina Kerberos mestre, que hospeda a cópia mestre do banco de dados de autenticação. É possível (embora não seja necessário) ter cópias adicionais e somente leitura do banco de dados em máquinas escravas em outros lugares do sistema. As vantagens de ter várias cópias do banco de dados são aquelas geralmente citadas para replicação: maior disponibilidade e melhor desempenho. Se a máquina mestre estiver inoperante, a autenticação ainda poderá ser obtida em uma das máquinas escravas. A capacidade de executar autenticação em qualquer uma das várias máquinas reduz a probabilidade de um gargalo na máquina mestre.

Manter várias cópias do banco de dados apresenta o problema da consistência dos dados. Considerámos que são suficientes métodos muito simples para lidar com a incoerência. O banco de dados principal é despejado a cada hora. O banco de dados é enviado, em sua totalidade, para as máquinas de escravos, que depois atualizam seus próprios bancos de dados. Um programa no host mestre, chamado kprop, envia a atualização para um programa de peer, chamado kprod, executado em cada uma das máquinas escravas. O primeiro kprop envia uma soma de verificação do novo banco de dados que está prestes a enviar. A soma de verificação é criptografada na chave do banco de dados mestre Kerberos, que as máquinas mestre e escravo Kerberos possuem. Os dados são transferidos pela rede para o kprod na máquina escrava. O servidor de propagação escrava calcula uma soma de verificação dos dados recebidos e, se corresponder à soma de verificação enviada pelo mestre, as novas informações são usadas para atualizar o banco de dados do escravo.

Todas as senhas no banco de dados Kerberos são criptografadas na chave do banco de dados mestre Portanto, as informações passadas do mestre para o escravo pela rede não são úteis para um espião. No entanto, é essencial que apenas as informações do host mestre sejam aceitas pelos escravos e que seja detectada adulteração de dados, portanto, o checksum.

Visão externa do Kerberos

Esta seção descreve o Kerberos do ponto de vista prático, primeiro como visto pelo usuário, depois do ponto de vista do programador de aplicativos e, finalmente, através das tarefas do administrador do Kerberos.

Ponto de vista do usuário do Kerberos

Se tudo der certo, o usuário dificilmente perceberá que Kerberos está presente. Em nossa implementação UNIX, o tíquete de concessão de tíquete é obtido de Kerberos como parte do processo de login. A alteração da senha Kerberos de um usuário faz parte do programa de senha. E os tíquetes Kerberos são automaticamente destruídos quando um usuário faz logoff.

Se a sessão de login do usuário durar mais do que o tempo de vida do tíquete de concessão de tíquete (atualmente, 8 horas), o usuário notará a presença de Kerberos porque na próxima vez que um aplicativo autenticado por Kerberos for executado, ele falhará. O tíquete Kerberos para ele expirou. Nesse ponto, o usuário pode executar o programa de quini para obter um novo tíquete para o servidor de concessão de tíquete. Como ao fazer login, uma senha deve ser fornecida para obtê-la. Um usuário executando o comando klist por curiosidade pode se surpreender com todos os tíquetes que foram obtidos silenciosamente em seu nome para serviços que exigem autenticação Kerberos.

Kerberos do ponto de vista do programador

Um programador que está escrevendo um aplicativo Kerberos frequentemente adicionará autenticação a um aplicativo de rede já existente que consiste em um cliente e um servidor. Nós chamamos esse processo de "Kerberização" de programa. O Kerberizing geralmente envolve fazer uma chamada para a biblioteca Kerberos para executar a autenticação na solicitação inicial de serviço. Também pode envolver chamadas para a biblioteca DES para criptografar mensagens e dados que são enviados posteriormente entre o cliente do aplicativo e o servidor do aplicativo.

As funções de biblioteca mais comumente usadas são krb_mk_req no lado do cliente e krb_rd_req no lado do servidor. A rotina krb_mk_req toma como parâmetros o nome, a instância e o território do servidor de destino, que será solicitado, e possivelmente uma soma de verificação dos dados a serem enviados. Em seguida, o cliente envia a mensagem retornada pela chamada krb_mk_req pela rede para o lado do servidor do aplicativo. Quando o servidor recebe esta mensagem, ele faz uma chamada para a rotina de biblioteca krb_rd_req. A rotina retorna um julgamento sobre a autenticidade da suposta identidade do remetente.

Se o aplicativo exigir que as mensagens enviadas entre o cliente e o servidor sejam secretas, então as chamadas de biblioteca podem ser feitas para krb_mk_priv (krb_rd_priv) para criptografar (descriptografar) mensagens na chave da sessão que ambos os lados agora compartilham.

O trabalho do administrador do Kerberos

O trabalho do administrador do Kerberos começa com a execução de um programa para inicializar o banco de dados. Outro programa deve ser executado para registrar os principais principais no banco de dados, como o nome do administrador do Kerberos com uma instância de administrador. O servidor de autenticação Kerberos e o servidor de administração devem ser iniciados. Se houver bancos de dados escravos, o administrador deve providenciar para que os programas para propagar atualizações de banco de dados do mestre para os escravos sejam lançados periodicamente.

Depois que essas etapas iniciais forem executadas, o administrador manipulará o banco de dados pela rede, usando o programa kadmin. Por meio desse programa, novos princípios podem ser adicionados e as senhas podem ser alteradas.

Em particular, quando um novo aplicativo Kerberos é adicionado ao sistema, o administrador do Kerberos deve executar algumas etapas para que ele funcione. O servidor deve ser registrado no banco de dados e uma chave privada deve ser atribuída ao servidor (normalmente é uma chave aleatória gerada automaticamente). Em seguida, alguns dados (incluindo a chave do servidor) devem ser extraídos do banco de dados e instalados em um arquivo na máquina do servidor. O arquivo padrão é /etc/srvtab. A rotina da biblioteca krb_rd_req chamada pelo servidor (consulte a seção anterior) usa as informações nesse arquivo para descriptografar mensagens enviadas criptografadas na chave privada do servidor. O arquivo /etc/srvtab autentica o servidor como uma senha digitada em um terminal autentica o usuário.

O administrador do Kerberos também deve garantir que as máquinas Kerberos estejam fisicamente seguras e também deve manter backups do banco de dados mestre.

Análise mais ampla do banco de dados Kerberos

Nesta seção, descrevemos como o Kerberos se encaixa no ambiente Athena, incluindo seu uso por outros serviços de rede e aplicativos, e como ele interage com os domínios remotos do

Kerberos. Para obter uma descrição mais completa do ambiente Athena, consulte G.W. Treese.

Uso de Kerberos por outros serviços de rede

Vários aplicativos de rede foram modificados para usar Kerberos. Os comandos rlogin e rsh primeiro tentam autenticar usando Kerberos. Um usuário com tíquetes Kerberos válidos pode fazer login novamente em outra máquina Athena sem precisar configurar arquivos .rhosts. Se a autenticação Kerberos falhar, os programas voltarão aos métodos de autorização habituais, nesse caso, os arquivos .rhosts.

Modificamos o Protocolo de Correio para usar Kerberos para autenticar usuários que desejam recuperar seus e-mails da "agência de correios". Um programa de envio de mensagens, chamado Zephyr, foi desenvolvido recentemente na Athena, e também usa Kerberos para autenticação.

O programa de inscrição de novos usuários, chamado de registro, usa tanto o Service Management System (SMS) quanto o Kerberos. A partir do SMS, ele determina se as informações inseridas pelo usuário futuro Athena, como nome e número de identificação do MIT, são válidas. Em seguida, ele verifica com Kerberos se o nome de usuário solicitado é exclusivo. Se tudo correr bem, uma nova entrada é feita no banco de dados Kerberos, contendo o nome de usuário e a senha.

Para uma discussão detalhada sobre o uso de Kerberos para proteger o sistema de arquivos de rede da Sun, consulte o apêndice.

Interação com outros Kerberi

Espera-se que diferentes organizações administrativas queiram usar Kerberos para autenticação de usuário. Espera-se também que, em muitos casos, os usuários de uma organização queiram usar serviços em outra. O Kerberos suporta vários domínios administrativos. A especificação de nomes no Kerberos inclui um campo chamado domínio. Este campo contém o nome do domínio administrativo no qual o usuário deve ser autenticado.

Os serviços são geralmente registrados em um único território e aceitarão apenas as credenciais emitidas por um servidor de autenticação para esse território. Um usuário é geralmente registrado em um único território (o território local), mas é possível para ele obter credenciais emitidas por outro território (o território remoto), com base na autenticação fornecida pelo território local. As credenciais válidas em um território remoto indicam o território no qual o usuário foi autenticado originalmente. Os serviços no território remoto podem escolher se devem honrar essas credenciais, dependendo do grau de segurança necessário e do nível de confiança no território que autenticou inicialmente o usuário.

Para executar a autenticação entre domínios, é necessário que os administradores de cada par de territórios selecionem uma chave a ser compartilhada entre seus territórios. Um usuário no território local pode solicitar um tíquete de concessão de tíquete do servidor de autenticação local para o servidor de concessão de tíquete no território remoto. Quando esse tíquete é usado, o servidor remoto de concessão de tíquete reconhece que a solicitação não é de seu próprio território e usa a chave previamente trocada para descriptografar o tíquete de concessão de tíquete. Em seguida, ele emite um tíquete como faria normalmente, exceto que o campo de território do cliente contém o nome do território no qual o cliente foi originalmente autenticado.

Essa abordagem pode ser estendida para permitir que se autentique por meio de uma série de domínios até alcançar o território com o serviço desejado. Para fazer isso, no entanto, seria

necessário registrar todo o caminho que foi percorrido, e não apenas o nome do território inicial em que o usuário foi autenticado. Em tal situação, tudo o que o servidor sabe é que A diz que B diz que C diz que o usuário é assim. Essa instrução só pode ser confiável se todos ao longo do caminho também forem confiáveis.

Questões e problemas abertos do Kerberos

Há vários problemas e problemas abertos associados ao mecanismo de autenticação Kerberos. Entre os problemas estão como decidir a vida útil correta de um tíquete, como permitir proxies e como garantir a integridade da estação de trabalho.

O problema do tempo de vida da passagem é a escolha da contrapartida adequada entre segurança e conveniência. Se a vida útil de um tíquete for longa, então se um tíquete e sua chave de sessão associada forem roubados ou colocados incorretamente, eles poderão ser usados por um período mais longo. Essas informações podem ser roubadas se um usuário se esquecer de fazer logoff de uma estação de trabalho pública. Como alternativa, se um usuário tiver sido autenticado em um sistema que permita vários usuários, outro usuário com acesso ao root poderá encontrar as informações necessárias para usar tíquetes roubados. O problema de dar um tíquete por um tempo de vida curto, no entanto, é que quando ele expirar, o usuário terá que obter um novo que exija que o usuário digite a senha novamente.

Um problema aberto é o problema de proxy. Como um usuário autenticado pode permitir que um servidor adquira outros serviços de rede em seu nome? Um exemplo onde isso seria importante é o uso de um serviço que obterá acesso a arquivos protegidos diretamente de um servidor de arquivos. Outro exemplo desse problema é o que chamamos de encaminhamento de autenticação. Se um usuário estiver conectado em uma estação de trabalho e fizer login em um host remoto, seria interessante se ele tivesse acesso aos mesmos serviços disponíveis localmente, ao executar um programa no host remoto. O que dificulta isso é que o usuário pode não confiar no host remoto, portanto, o encaminhamento de autenticação não é desejável em todos os casos. Atualmente, não temos uma solução para este problema.

Outro problema, importante no ambiente Athena, é como garantir a integridade do software em execução em uma estação de trabalho. Isso não é um grande problema em estações de trabalho privadas, já que o usuário que vai usá-lo tem controle sobre ele. Em estações de trabalho públicas, no entanto, alguém pode ter aparecido e modificado o programa de login para salvar a senha do usuário. A única solução atualmente disponível em nosso ambiente é tornar difícil para as pessoas modificar o software em execução nas estações de trabalho públicas. Uma solução melhor exigiria que a chave do usuário nunca deixasse um sistema que o usuário saiba que possa ser confiável. Uma maneira de fazer isso seria se o usuário tivesse um smartcard capaz de fazer as criptografias necessárias no protocolo de autenticação.

Status do Kerberos

Uma versão protótipo de Kerberos entrou em produção em setembro de 1986. Desde janeiro de 1987, Kerberos é o único meio do Project Athena de autenticar seus 5.000 usuários, 650 estações de trabalho e 65 servidores. Além disso, agora o Kerberos está sendo usado no lugar de arquivos .rhosts para controlar o acesso em vários sistemas de compartilhamento de tempo do Athena.

Reconhecimentos de Kerberos

Kerberos foi inicialmente projetado por Steve Miller e Clifford Neuman com sugestões de Jeff Schiller e Jerry Saltzer. Desde então, muitas outras pessoas participaram no projeto. Entre eles estão 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. em Treese e Stan Zanarotti.

Estamos gratos a Dan Geer, Kathy Lieben, Josh Lubarr, Ken Raeburn, Jerry Saltzer, Ed Steiner, Robbert van Renesse e Win Treese, cujas sugestões melhoraram bastante versões anteriores deste artigo.

Jedlinsky, J.T. Kohl e W.E. Sommerfeld, "The Zephyr Notification System", em Usenix Conference Proceedings (inverno, 1988).

M.A. Rosenstein, D.E. Geer e P.J. Levine, em Usenix Conference Proceedings (inverno, 1988).

R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh e B. Lyon, "Design and Implementation of the Sun Network Filesystem", nos Procedimentos de Conferência de Usenix (verão de 1985).

Anexo: Aplicação do Kerberos ao Network File System (NFS) da SUN

Um componente chave do sistema de estação de trabalho Project Athena é a interposição da rede entre a estação de trabalho do usuário e seu armazenamento de arquivos privado (diretório inicial). Todo o armazenamento privado reside em um conjunto de computadores (atualmente VAX 11/750s) dedicados a essa finalidade. Isso nos permite oferecer serviços em estações de trabalho UNIX disponíveis publicamente. Quando um usuário faz login em uma dessas estações de trabalho disponíveis publicamente, em vez de validar seu nome e senha em um arquivo de senha residente localmente, usamos Kerberos para determinar sua autenticidade. O programa de login solicita um nome de usuário (como em qualquer sistema UNIX). Este nome de usuário é usado para buscar um tíquete de concessão de tíquete Kerberos. O programa de login usa a senha para gerar uma chave DES para descriptografar o tíquete. Se a descriptografia for bem-sucedida, o diretório inicial do usuário será localizado consultando o serviço de nomenclatura Hesod e montado por meio do NFS. Em seguida, o programa de login passa o controle para o shell do usuário, que pode executar os arquivos de personalização tradicionais por usuário, pois o diretório inicial agora está "anexado" à estação de trabalho. O serviço Hesiod também é usado para construir uma entrada no arquivo de senha local. (Isso beneficia os programas que pesquisam informações em /etc/passwd.)

De várias opções para a entrega do serviço de arquivos remotos, escolhemos o Network File System da Sun. No entanto, este sistema não consegue corresponder às nossas necessidades de uma forma crucial. O NFS assume que todas as estações de trabalho se enquadram em duas categorias (conforme visualizado do ponto de vista de um servidor de arquivos): confiável e não confiável. Os sistemas não confiáveis não podem acessar nenhum arquivo, o que é confiável. Os sistemas confiáveis são completamente confiáveis. Pressupõe-se que um sistema confiável seja gerido por uma gestão amigável. Especificamente, é possível de uma estação de trabalho confiável se mascarar como qualquer usuário válido do sistema de serviço de arquivos e, assim, obter acesso a praticamente todos os arquivos do sistema. (Apenas os arquivos de "raiz" estão isentos.)

Em nosso ambiente, o gerenciamento de uma estação de trabalho (no sentido tradicional de

gerenciamento de sistemas UNIX) está nas mãos do usuário que a está usando atualmente. Não fazemos nenhum segredo da senha raiz em nossas estações de trabalho, pois percebemos que um usuário realmente hostil pode entrar pelo fato de que ele está sentado no mesmo local físico da máquina e tem acesso a todas as funções do console. Portanto, não podemos realmente confiar em nossas estações de trabalho na interpretação de confiança do NFS. Para permitir controles de acesso adequados em nosso ambiente, tivemos que fazer algumas modificações no software NFS básico e integrar Kerberos ao esquema.

NFS de Kerberos não modificado

Na implementação da NFS com a qual começamos (da Universidade de Wisconsin), a autenticação foi fornecida na forma de um pedaço de dados incluído em cada solicitação da NFS (chamada de "credencial" na terminologia da NFS). Esta credencial contém informações sobre o identificador de usuário exclusivo (UID) do solicitante e uma lista dos identificadores de grupo (GIDs) da associação do solicitante. Essas informações são usadas pelo servidor NFS para verificação de acesso. A diferença entre uma estação de trabalho confiável e uma não confiável é se suas credenciais são aceitas ou não pelo servidor NFS.

NFS modificado de Kerberos

Em nosso ambiente, os servidores NFS devem aceitar credenciais de uma estação de trabalho se e somente se as credenciais indicarem o UID do usuário da estação de trabalho, e nenhum outro.

Uma solução óbvia seria mudar a natureza das credenciais de meros indicadores de UID e GIDs para dados autenticados de Kerberos totalmente soprados. No entanto, se esta solução fosse adotada, seria paga uma penalidade significativa em termos de desempenho. As credenciais são trocadas em todas as operações de NFS, incluindo todas as atividades de leitura e gravação de disco. A inclusão de uma autenticação Kerberos em cada transação de disco adicionaria um número razoável de criptografia completa (feita em software) por transação e, de acordo com nossos cálculos de envelope, teria oferecido desempenho inaceitável. (Também seria necessário colocar as rotinas da biblioteca Kerberos no espaço de endereço do kernel.)

Precisávamos de uma abordagem híbrida, descrita abaixo. A ideia básica é ter as credenciais de mapa do servidor NFS recebidas das estações de trabalho do cliente, para uma credencial válida (e possivelmente diferente) no sistema do servidor. Esse mapeamento é executado no kernel do servidor em cada transação NFS e é configurado no tempo de "montagem" por um processo no nível do usuário que envolve a autenticação moderada do Kerberos antes de estabelecer um mapeamento de credenciais de kernel válido.

Para implementar isso, adicionamos uma nova chamada do sistema ao kernel (necessária somente em sistemas de servidor, não em sistemas cliente) que fornece o controle da função de mapeamento que mapeia credenciais de entrada das estações de trabalho do cliente para credenciais válidas para uso no servidor (se houver). A função básica de mapeamento mapeia a tupla:

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

para uma credencial NFS válida no sistema do servidor. O CLIENT-IP-ADDRESS é extraído do pacote de solicitação NFS fornecido pelo sistema cliente. Note: todas as informações na credencial gerada pelo cliente, exceto UID-ON-CLIENT, são descartadas.

Se não houver mapeamento, o servidor reage de uma das duas maneiras, dependendo de estar configurado. Em nossa configuração amigável, as solicitações não mapeáveis são padronizadas nas credenciais do usuário "ninguém" que não tem acesso privilegiado e tem um UID exclusivo. Servidores não amigáveis retornam um erro de acesso NFS quando não é possível encontrar um mapeamento válido para uma credencial NFS recebida.

Nossa nova chamada do sistema é usada para adicionar e excluir entradas do mapa residente do kernel. Ele também permite limpar todas as entradas que mapeiam para um UID específico no sistema do servidor ou limpar todas as entradas de um CLIENTE-IP-ADDRESS específico.

Modificamos o daemon de montagem (que lida com solicitações de montagem NFS em sistemas de servidor) para aceitar um novo tipo de transação, a solicitação de mapeamento de autenticação Kerberos. Basicamente, como parte do processo de montagem, o sistema cliente fornece um autenticador Kerberos junto com uma indicação de seu UID-ON-CLIENT (criptografado no autenticador Kerberos) na estação de trabalho. O daemon de montagem do servidor converte o nome principal do Kerberos em um nome de usuário local. Esse nome de usuário é pesquisado em um arquivo especial para fornecer a lista de UIDs e GIDs do usuário. Para maior eficiência, este arquivo é um arquivo de banco de dados ndbm com o nome de usuário como a chave. A partir dessas informações, uma credencial NFS é construída e entregue ao kernel como o mapeamento válido da tupla <CLIENT-IP-ADDRESS, CLIENT-UID> para essa solicitação.

No momento da desmontagem, uma solicitação é enviada ao daemon de montagem para remover o mapeamento adicionado anteriormente do kernel. Também é possível enviar uma solicitação no momento do logoff para invalidar todo o mapeamento para o usuário atual no servidor em questão, limpando assim todos os mapeamentos restantes que existem (embora não devam) antes que a estação de trabalho seja disponibilizada para o próximo usuário.

Implicações de segurança do Kerberos do NFS modificado

Essa implementação não é totalmente segura. Para começar, os dados do usuário ainda são enviados pela rede em um formulário não criptografado e, portanto, interceptável. A autenticação de baixo nível por transação é baseada em um par <CLIENT-IP-ADDRESS, CLIENT-UID> fornecido não criptografado no pacote de solicitação. Essas informações podem ser forjadas e, portanto, a segurança pode ser comprometida. No entanto, deve-se observar que somente enquanto um usuário estiver usando seus arquivos ativamente (isto é, enquanto estiver conectado) são mapeamentos válidos implementados e, portanto, essa forma de ataque é limitada a quando o usuário em questão estiver conectado. Quando um usuário não está conectado, nenhuma quantidade de falsificação de endereço IP permitirá acesso não autorizado a seus arquivos.

Referências de Kerberos

1. S.P. Miller, B.C. Neuman, J.I. Schiller e J.H. Saltzer, Seção E.2.1: Sistema de autenticação e autorização Kerberos, M.I.T. Project Athena, Cambridge, Massachusetts (21 de dezembro de 1987).
2. E. Balkovich, S.R. Lerman e R.P. Parmelee, "Computação no Ensino Superior: The Athena Experience," Communications of the ACM, Vol. 28(11), pp. 1214-1224, ACM (novembro de 1985).
3. R.M. Needham e M.D. Schroeder, "Using Encryption for Authentication in Large Networks of

Computers," Communications of the ACM, vol. 21(12), pp. 993-999 (dezembro de 1978).
4. V.L. Voydock e S.T. Kent, "Security Mechanisms in High-Level Network Protocols," Computing Surveys, Vol. 15(2), ACM (junho de 1983).
5. National Bureau of Standards, "Data Encryption Standard," Federal Information Processing Standards Publications 46, Government Printing Office, Washington, DC (1977).
6. SP Dyer, "Hesiod", nos Procedimentos da Conferência Usenix (Winter, 1988).
7. W.J. Bryant, Tutorial do Programador Kerberos, Projeto Athena do MIT (em preparação).
8. W.J. Bryant, Kerberos Administrator's Manual, MIT Project Athena (em preparação).
9. G.W. Treese, "Berkeley Unix em 1000 estações de trabalho: Athena muda para 4.3BSD", nos Procedimentos de Conferência de Usenix (inverno, 1988).
10. C.A. DellaFera, M.W. Eichin, R.S. Francês, D.C. Jedlinsky, J.T. Kohl e W.E. Sommerfeld, "The Zephyr Notification System", em Usenix Conference Proceedings (inverno, 1988).
11. M.A. Rosenstein, D.E. Geer e P.J. Levine, em Usenix Conference Proceedings (inverno, 1988).
12. R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh e B. Lyon, "Design and Implementation of the Sun Network Filesystem", nos Procedimentos de Conferência de Usenix (verão de 1985).

Informações Relacionadas

- Página de suporte do Kerberos
- Suporte Técnico e Documentação - Cisco Systems