Panoramica di Kerberos: un servizio di autenticazione per sistemi di rete aperti

Sommario

Introduzione

In un ambiente di elaborazione di rete, non è possibile ritenere attendibile una postazione di lavoro per l'identificazione corretta degli utenti sui servizi di rete. Kerberos fornisce un approccio alternativo offrendo un servizio di autenticazione di terze parti attendibile per verificare le identità degli utenti. In questo documento viene fornita una panoramica del modello di autenticazione Kerberos implementato per Project Athena di MIT. Vengono descritti i protocolli usati dai client, dai server e da Kerberos per eseguire l'autenticazione e le attività di gestione e replica del database necessarie. Inoltre, Kerberos viene spiegato dal punto di vista dell'utente, del programmatore e dell'amministratore. Viene illustrato il ruolo di Kerberos nel quadro più ampio di Athena e viene fornito un elenco di applicazioni che usano attualmente Kerberos per l'autenticazione degli utenti. Infine, viene trattato il case study di un'autenticazione Kerberos aggiunta al Network File System di Sun per integrare Kerberos nell'applicazione esistente.

Autori 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 è stato un membro dello staff del progetto Athena durante la fase di progettazione e implementazione iniziale di Kerberos.
- Jeffrey I. Schiller, Project Athena, Massachusetts Institute of Technology, Cambridge, MA 02139, jis@ATHENA.MIT.EDU

Introduzione a Kerberos

Questo documento offre una panoramica di Kerberos, un sistema di autenticazione progettato da Miller e Neuman. per ambienti di elaborazione in rete aperta e descrive la nostra esperienza nell'utilizzo di questo strumento al Progetto Athena del MIT. Nella sezione sulla motivazione, spieghiamo perché è necessario un nuovo modello di autenticazione per le reti aperte e quali sono i suoi requisiti. Il Kerberos? In questa sezione vengono elencati i componenti del software Kerberos e viene descritto come interagiscono nel fornire il servizio di autenticazione. Nella sezione Nomi Kerberos viene descritto lo schema di denominazione Kerberos.

Modalità di funzionamento di Kerberos Presentazione dei componenti di base dell'autenticazione Kerberos, ovvero il ticket e l'autenticatore. Di seguito vengono descritti i due protocolli di autenticazione disponibili: l'autenticazione iniziale di un utente a Kerberos (analoga all'accesso) e il protocollo per l'autenticazione reciproca di un potenziale consumatore e di un potenziale produttore di un servizio di rete.

Kerberos richiede una banca dati di informazioni sui suoi clienti; La sezione Database Kerberos descrive il database, la relativa gestione e il protocollo per la relativa modifica. La sezione Kerberos dall'esterno descrive l'interfaccia Kerberos ai relativi utenti, programmatori di applicazioni e amministratori. Nella sezione Immagine più grande, descriviamo come il Progetto Athena Kerberos si adatta al resto dell'ambiente Athena. Descriviamo inoltre l'interazione di diversi domini o aree di autenticazione Kerberos; nel nostro caso, la relazione tra il Progetto Athena Kerberos e il Kerberos in esecuzione al Laboratorio di Informatica del MIT.

Nella sezione Problemi e problemi aperti vengono menzionati i problemi aperti e quelli non ancora

risolti. L'ultima sezione fornisce lo stato attuale di Kerberos al Progetto Athena. Nell'Appendice viene descritto in dettaglio come Kerberos viene applicato a un servizio di file di rete per autenticare gli utenti che desiderano accedere ai file system remoti.

Concetti su Kerberos

In questo documento vengono utilizzati termini che possono essere ambigui, nuovi per il lettore o utilizzati diversamente altrove. Di seguito riportiamo l'utilizzo di questi termini.

Utente, Client, Server: per utente si intende un essere umano che utilizza un programma o un servizio. Anche un cliente usa qualcosa, ma non è necessariamente una persona; può essere un programma. Le applicazioni di rete sono spesso costituite da due parti: un programma in esecuzione su un computer che richiede un servizio remoto e un altro programma in esecuzione sul computer remoto che esegue tale servizio. Questi vengono chiamati rispettivamente lato client e lato server dell'applicazione. Spesso un client contatta un server per conto di un utente.

Ogni entità che utilizza il sistema Kerberos, sia che si tratti di un utente o di un server di rete, è in un certo senso un client, poiché utilizza il servizio Kerberos. Per distinguere i clienti Kerberos dai clienti di altri servizi, utilizziamo il termine principale per indicare tale entità. Si noti che un'entità Kerberos può essere un utente o un server. (La denominazione degli utenti/gruppi/ruoli di Kerberos verrà descritta in una sezione successiva).

Servizio vs. server: il servizio viene utilizzato come specifica astratta di alcune azioni da eseguire. Un processo che esegue queste azioni è denominato server. In un determinato momento, è possibile che più server (in genere in esecuzione su computer diversi) eseguano un determinato servizio. Ad esempio, ad Athena c'è un server rlogin BSD UNIX in esecuzione su ciascuno dei nostri sistemi di condivisione tempo.

Chiave, Chiave privata, Password: Kerberos utilizza la crittografia a chiave privata. A ogni entità Kerberos viene assegnato un numero elevato, ovvero la relativa chiave privata, nota solo a tale entità e a Kerberos. Nel caso di un utente, la chiave privata è il risultato di una funzione unidirezionale applicata alla password dell'utente. La chiave viene utilizzata come abbreviazione per la chiave privata.

Credenziali - Purtroppo, questa parola ha un significato speciale sia per Sun Network File System che per il sistema Kerberos. Viene indicato esplicitamente se si tratta di credenziali NFS o di credenziali Kerberos, altrimenti il termine viene utilizzato nel senso della lingua inglese.

Master e slave: è possibile eseguire il software di autenticazione Kerberos su più computer. Tuttavia, esiste sempre una sola copia definitiva del database Kerberos. Il computer che ospita il database è detto computer master o semplicemente master. Altre macchine possono possedere copie di sola lettura del database Kerberos, e queste sono chiamate schiavi.

Motivazione dietro Kerberos

In un ambiente di personal computing non di rete, le risorse e le informazioni possono essere protette fisicamente proteggendo il PC. In un ambiente informatico basato sulla condivisione dei tempi, il sistema operativo protegge gli utenti gli uni dagli altri e controlla le risorse. Per determinare le capacità di lettura o di modifica di ogni utente, è necessario che il sistema di condivisione tempo identifichi ogni utente. Questa operazione viene eseguita al momento dell'accesso dell'utente.

In una rete di utenti che necessitano di servizi da molti computer separati, è possibile adottare tre approcci per il controllo dell'accesso: Non si può fare nulla, basandosi sulla macchina a cui l'utente è collegato per prevenire l'accesso non autorizzato; è possibile richiedere all'host di provare la propria identità, ma considerare attendibile la parola dell'host relativa all'utente; oppure richiedere all'utente di dimostrare la propria identità per ogni servizio richiesto.

In un ambiente chiuso in cui tutte le macchine sono sotto stretto controllo, si può utilizzare il primo approccio. Il controllo da parte dell'organizzazione di tutti gli host che comunicano in rete rappresenta un approccio ragionevole.

In un ambiente più aperto, è possibile considerare in modo selettivo solo gli host sotto il controllo dell'organizzazione. In tal caso, ciascun host deve dimostrare la propria identità. I programmi rlogin e rsh utilizzano questo approccio. In questi protocolli l'autenticazione viene eseguita controllando l'indirizzo Internet da cui è stata stabilita una connessione.

Nel contesto di Athena, dobbiamo essere in grado di onorare le richieste provenienti da ospiti che non sono sotto controllo organizzativo. Gli utenti hanno il controllo completo delle workstation: possono riavviare il sistema, riattivarlo da soli o avviare il sistema da soli. In quanto tale, occorre adottare il terzo approccio; l'utente deve dimostrare la propria identità per ogni servizio desiderato. Il server deve inoltre dimostrare la propria identità. Non è sufficiente proteggere fisicamente l'host che esegue un server di rete; è possibile che un utente in un altro punto della rete venga mascherato come server specificato.

Il nostro ambiente pone diversi requisiti su un meccanismo di identificazione. Innanzitutto, deve essere sicuro. Evitare di farlo deve essere abbastanza difficile da evitare che un potenziale aggressore trovi il meccanismo di autenticazione come il collegamento debole. Un utente che controlla la rete non dovrebbe essere in grado di ottenere le informazioni necessarie per rappresentare un altro utente. In secondo luogo, deve essere affidabile. L'accesso a molti servizi dipende dal servizio di autenticazione. Se non è affidabile, il sistema di servizi nel suo complesso non sarà affidabile. In terzo luogo, deve essere trasparente. Idealmente, l'utente non dovrebbe essere a conoscenza dell'autenticazione in corso. Infine, dovrebbe essere scalabile. Molti sistemi possono comunicare con gli host Athena. Non tutte queste caratteristiche supportano il nostro meccanismo, ma se lo facessero, il software non dovrebbe rompersi.

Kerberos è il risultato del nostro lavoro per soddisfare i requisiti di cui sopra. Quando un utente raggiunge una workstation, esegue l'accesso. A quanto risulta all'utente, questa identificazione iniziale è sufficiente per dimostrare la propria identità a tutti i server di rete richiesti per la durata della sessione di accesso. La protezione di Kerberos si basa sulla protezione di diversi server di autenticazione, ma non sul sistema dal quale gli utenti eseguono l'accesso né sulla protezione dei server finali che verranno utilizzati. Il server di autenticazione consente a un utente autenticato di provare la propria identità su server dislocati in rete.

L'autenticazione è un elemento fondamentale per un ambiente di rete sicuro. Se, ad esempio, un server conosce per certa l'identità di un cliente, può decidere se fornire il servizio, se concedere privilegi speciali all'utente, a chi spetterà la fattura per il servizio e così via. In altre parole, gli schemi di autorizzazione e contabilità possono essere basati sull'autenticazione fornita da Kerberos, che garantisce una protezione equivalente al personal computer isolato o al sistema di condivisione dei tempi.

Cos'è Kerberos?

Kerberos è un servizio di autenticazione attendibile di terze parti basato sul modello presentato da

Needham e Schroeder. È affidabile nel senso che ciascuno dei suoi clienti ritiene che il giudizio di Kerberos sull'identità di ciascuno dei suoi clienti sia accurato. Al modello originale sono stati aggiunti dei timestamp (grandi numeri che rappresentano la data e l'ora correnti) per facilitare il rilevamento della riproduzione. La riproduzione viene eseguita quando un messaggio viene rubato dalla rete e inviato nuovamente in seguito. Per una descrizione più completa della riproduzione e per altri problemi di autenticazione, vedere Voydock and Kent.

Cosa fa Kerberos?

Kerberos mantiene un database dei relativi client e delle relative chiavi private. La chiave privata è un numero elevato noto solo a Kerberos e al client a cui appartiene. Se il client è un utente, si tratta di una password crittografata. I servizi di rete che richiedono l'autenticazione vengono registrati con Kerberos, così come i client che desiderano utilizzare tali servizi. Le chiavi private vengono negoziate al momento della registrazione.

Dato che Kerberos conosce queste chiavi private, può creare messaggi che convincono un cliente che un altro è realmente chi sostiene di essere. Kerberos genera inoltre chiavi private temporanee, denominate chiavi di sessione, che vengono fornite a due client e nessun altro. È possibile utilizzare una chiave di sessione per crittografare i messaggi tra due parti.

Kerberos fornisce tre diversi livelli di protezione. Il programmatore dell'applicazione determina quale sia appropriato, in base ai requisiti dell'applicazione. Alcune applicazioni, ad esempio, richiedono solo che l'autenticità venga stabilita all'avvio di una connessione di rete e possono presupporre che ulteriori messaggi provenienti da un determinato indirizzo di rete provengano dal destinatario autenticato. Il nostro file system di rete autenticato utilizza questo livello di sicurezza.

Altre applicazioni richiedono l'autenticazione di ogni messaggio, ma non hanno importanza se il contenuto del messaggio viene divulgato o meno. Per questi motivi, Kerberos fornisce messaggi sicuri. I messaggi privati offrono tuttavia un livello di protezione superiore, in cui ogni messaggio non viene solo autenticato, ma anche crittografato. I messaggi privati vengono utilizzati, ad esempio, dal server Kerberos stesso per l'invio di password tramite la rete.

Componenti software Kerberos

L'attuazione di Athena comprende diversi moduli:

- Libreria applicazioni Kerberos
- libreria di crittografia
- libreria di database
- programmi di amministrazione del database
- server di amministrazione
- server di autenticazione
- software di propagazione db
- programmi utente
- applicazioni

La libreria delle applicazioni Kerberos fornisce un'interfaccia per i client e i server delle applicazioni. Contiene, tra le altre, le routine per la creazione o la lettura delle richieste di autenticazione e le routine per la creazione di messaggi sicuri o privati.

La crittografia in Kerberos si basa su DES, lo standard di crittografia dei dati. La libreria di crittografia implementa queste routine. Sono disponibili diversi metodi di crittografia, con

compromessi tra velocità e sicurezza. Viene inoltre fornita un'estensione della modalità CBC (Cipher Block Chaining) DES, denominata modalità CBC di propagazione. In CBC, un errore viene propagato solo attraverso il blocco corrente della cifratura, mentre in PCBC, l'errore viene propagato attraverso il messaggio. In questo modo, se si verifica un errore, l'intero messaggio viene reso inutile e non solo parte di esso. La libreria di crittografia è un modulo indipendente e può essere sostituita con altre implementazioni DES o con una libreria di crittografia diversa.

Un altro modulo sostituibile è il sistema di gestione del database. L'attuale implementazione Athena della libreria di database utilizza ndbm, anche se Ingres è stato originariamente utilizzato. È possibile utilizzare anche altre librerie di gestione di database.

Le esigenze della banca dati Kerberos sono chiare; per ogni entità viene mantenuto un record contenente il nome, la chiave privata e la data di scadenza dell'entità, insieme ad alcune informazioni amministrative. La data di scadenza è la data dopo la quale una voce non è più valida. In genere è impostato su alcuni anni nel futuro alla registrazione.)

Altre informazioni utente, quali il nome reale, il numero di telefono e così via, vengono conservate da un altro server, il server dei nomi Hesiod. In questo modo le informazioni sensibili, ossia le password, possono essere gestite da Kerberos utilizzando misure di sicurezza piuttosto elevate; mentre le informazioni non sensibili conservate da Hesiod sono trattate in modo diverso; ad esempio, può essere inviato in rete in modo non crittografato.

I server Kerberos utilizzano la libreria di database e gli strumenti per l'amministrazione del database.

Il server di amministrazione (o server KDBM) fornisce un'interfaccia di rete di lettura/scrittura al database. Il lato client del programma può essere eseguito su qualsiasi computer della rete. Il lato server, tuttavia, deve essere eseguito sul computer che ospita il database Kerberos per poter apportare modifiche al database.

Il server di autenticazione (o server Kerberos), invece, esegue operazioni di sola lettura sul database Kerberos, ovvero l'autenticazione delle entità e la generazione delle chiavi di sessione. Poiché questo server non modifica il database Kerberos, può essere eseguito su un computer che ospita una copia di sola lettura del database Kerberos master.

Il software di propagazione del database gestisce la replica del database Kerberos. È possibile avere copie del database su diversi computer, con una copia del server di autenticazione in esecuzione su ciascun computer. Ognuno di questi computer slave riceve un aggiornamento del database Kerberos dal computer master a intervalli specificati.

Infine, esistono programmi per gli utenti finali che consentono di accedere a Kerberos, modificare una password Kerberos e visualizzare o eliminare i ticket Kerberos (i ticket vengono illustrati più avanti).

Nomi Kerberos

Parte dell'autenticazione di un'entità è la denominazione. Il processo di autenticazione consiste nella verifica che il client sia quello indicato in una richiesta. In cosa consiste un nome? In Kerberos vengono denominati sia gli utenti che i server. Per quanto riguarda il server di autenticazione, sono equivalenti. Un nome è costituito da un nome primario, un'istanza e un realm, espresso come name.instance@realm.

Il nome primario è il nome dell'utente o del servizio. La variante viene utilizzata per distinguere tra le variazioni sul nome primario. Per gli utenti, un'istanza può comportare privilegi speciali, ad esempio le istanze "root" o "admin". Per i servizi nell'ambiente Athena, l'istanza è in genere il nome del computer in cui viene eseguito il server. Ad esempio, il servizio rlogin ha istanze diverse su host diversi: rlogin.priam è il server rlogin sull'host denominato priam. Un ticket Kerberos è valido solo per un singolo server denominato. Pertanto, è necessario un ticket separato per accedere a istanze diverse dello stesso servizio. Il realm è il nome di un'entità amministrativa che gestisce i dati di autenticazione. Ad esempio, le diverse istituzioni possono disporre ognuna di una propria macchina Kerberos che ospita un database diverso. Hanno ambiti Kerberos diversi. (I realm vengono ulteriormente trattati in Interazione con altri Kerberi.)

Come funziona Kerberos

In questa sezione vengono descritti i protocolli di autenticazione Kerberos. Come accennato in precedenza, il modello di autenticazione Kerberos si basa sul protocollo di distribuzione delle chiavi Needham e Schroeder. Quando un utente richiede un servizio, è necessario stabilire la propria identità. A tale scopo, viene presentato un ticket al server, insieme alla prova che il ticket è stato originariamente rilasciato all'utente, non rubato. L'autenticazione tramite Kerberos prevede tre fasi. Nella prima fase l'utente ottiene le credenziali da utilizzare per richiedere l'accesso ad altri servizi. Nella seconda fase l'utente richiede l'autenticazione per un servizio specifico. Nella fase finale, l'utente presenta tali credenziali al server finale.

Credenziali Kerberos

Il modello di autenticazione Kerberos prevede due tipi di credenziali: biglietti e autenticatori. Entrambe sono basate sulla crittografia a chiave privata, ma vengono crittografate utilizzando chiavi diverse. Un ticket viene utilizzato per passare in modo sicuro l'identità della persona alla quale è stato emesso tra il server di autenticazione e il server finale. Un biglietto trasmette anche informazioni che possono essere utilizzate per assicurarsi che la persona che lo utilizza sia la stessa a cui è stato rilasciato. L'autenticatore contiene le informazioni aggiuntive che, se confrontate con quelle nel ticket, dimostrano che il client che presenta il ticket è lo stesso a cui è stato emesso il ticket.

Un ticket è valido per un singolo server e un singolo client. Contiene il nome del server, il nome del client, l'indirizzo Internet del client, un indicatore orario, una durata e una chiave di sessione casuale. Queste informazioni vengono crittografate utilizzando la chiave del server per cui verrà utilizzato il ticket. Una volta emesso, il ticket può essere utilizzato più volte dal client per ottenere l'accesso al server specificato, fino alla scadenza del ticket. Poiché il ticket è crittografato nella chiave del server, è consigliabile consentire all'utente di passarlo al server senza doversi preoccupare di modificarlo.

A differenza del ticket, l'autenticatore può essere utilizzato una sola volta. È necessario generarne uno nuovo ogni volta che un client desidera utilizzare un servizio. Il problema non si verifica perché il client è in grado di creare l'autenticatore stesso. Un autenticatore contiene il nome del client, l'indirizzo IP della workstation e l'ora corrente della workstation. L'autenticatore viene crittografato nella chiave di sessione che fa parte del ticket.

Ottieni ticket Kerberos iniziale

Quando l'utente raggiunge una workstation, solo un'informazione può dimostrare la propria identità: la password dell'utente. Lo scambio iniziale con il server di autenticazione è progettato

per ridurre al minimo la possibilità che la password venga compromessa, ma allo stesso tempo non consente all'utente di autenticarsi correttamente senza che sia a conoscenza della password. Il processo di accesso è identico a quello di un sistema di condivisione tempo. Dietro le quinte, però, è ben diverso.

All'utente viene richiesto di immettere il proprio nome utente. Una volta immessa, viene inviata una richiesta al server di autenticazione contenente il nome dell'utente e il nome di un servizio speciale noto come servizio di concessione ticket.

Il server di autenticazione verifica di essere a conoscenza del client. In questo caso, viene generata una chiave di sessione casuale che verrà successivamente utilizzata tra il client e il server di concessione dei ticket. Viene quindi creato un ticket per il server di concessione ticket contenente il nome del client, il nome del server di concessione ticket, l'ora corrente, la durata del ticket, l'indirizzo IP del client e la chiave di sessione casuale appena creata. Il tutto crittografato in una chiave nota solo al server che rilascia i ticket e al server di autenticazione.

Il server di autenticazione invia quindi al client il ticket, insieme a una copia della chiave di sessione casuale e ad alcune informazioni aggiuntive. Questa risposta viene crittografata nella chiave privata del client, nota solo a Kerberos e al client, che deriva dalla password dell'utente.

Una volta ricevuta la risposta dal client, all'utente viene richiesta la password. La password viene convertita in una chiave DES e utilizzata per decrittografare la risposta dal server di autenticazione. Il ticket e la chiave di sessione, insieme ad alcune altre informazioni, vengono memorizzati per un utilizzo futuro e la password dell'utente e la chiave DES vengono cancellate dalla memoria.

Una volta completato lo scambio, la workstation possiede informazioni che può utilizzare per provare l'identità del proprio utente per tutta la durata del ticket di concessione ticket. Finché il software sulla workstation non è stato precedentemente manomesso, non esistono informazioni che consentano a qualcun altro di impersonare l'utente oltre la durata del ticket.

Richiedi un servizio Kerberos

Per il momento, facciamo finta che l'utente abbia già un biglietto per il server desiderato. Per accedere al server, l'applicazione crea un autenticatore contenente il nome e l'indirizzo IP del client e l'ora corrente. L'autenticatore viene quindi crittografato nella chiave di sessione ricevuta con il ticket per il server. Il client invia quindi l'autenticatore insieme al ticket al server nel modo definito dalla singola applicazione.

Dopo che l'autenticatore e il ticket sono stati ricevuti dal server, quest'ultimo decrittografa il ticket, utilizza la chiave di sessione inclusa nel ticket per decrittografare l'autenticatore, confronta le informazioni contenute nel ticket con quelle contenute nell'autenticatore, l'indirizzo IP da cui è stata ricevuta la richiesta e l'ora corrente. Se tutto corrisponde, la richiesta può procedere.

Si presume che gli orologi vengano sincronizzati in pochi minuti. Se l'ora nella richiesta è troppo lontana nel futuro o nel passato, il server considera la richiesta come un tentativo di ripetere una richiesta precedente. Il server è inoltre autorizzato a tenere traccia di tutte le richieste passate con timestamp ancora validi. Al fine di scongiurare ulteriori attacchi di tipo replay, una richiesta ricevuta con lo stesso ticket e timestamp di una richiesta già ricevuta può essere scartata.

Infine, se il client specifica che desidera che anche il server dimostri la propria identità, il server ne aggiunge una all'indicatore orario inviato dal client nell'autenticatore, cripta il risultato nella chiave

di sessione e invia nuovamente il risultato al client.

Alla fine di questo scambio, il server è certo che, secondo Kerberos, il cliente è chi dice di essere. In caso di autenticazione reciproca, il client è inoltre convinto che il server sia autentico. Inoltre, il client e il server condividono una chiave che nessun altro conosce, e possono tranquillamente supporre che un messaggio ragionevolmente recente crittografato in quella chiave abbia avuto origine dall'altra parte.

Ottieni ticket server Kerberos

Tenere presente che un ticket è valido solo per un singolo server. È quindi necessario ottenere un ticket separato per ogni servizio che il cliente desidera utilizzare. I ticket per i singoli server possono essere ottenuti dal servizio di concessione ticket. Poiché il servizio di concessione ticket è esso stesso un servizio, utilizza il protocollo di accesso al servizio descritto nella sezione precedente.

Quando un programma richiede un ticket che non è stato ancora richiesto, invia una richiesta al server di concessione ticket. La richiesta contiene il nome del server per il quale viene richiesto un ticket, insieme al ticket di concessione ticket e a un autenticatore creato come descritto nella sezione precedente.

Il server di concessione dei ticket controlla quindi l'autenticatore e il ticket di concessione dei ticket come descritto in precedenza. Se valido, il server che concede i ticket genera una nuova chiave di sessione casuale da utilizzare tra il client e il nuovo server. Viene quindi creato un ticket per il nuovo server contenente il nome del client, il nome del server, l'ora corrente, l'indirizzo IP del client e la nuova chiave di sessione appena generata. La durata del nuovo ticket è il valore minimo della durata rimanente del ticket di concessione ticket e il valore predefinito del servizio.

Il server di concessione dei ticket invia quindi il ticket, insieme alla chiave di sessione e ad altre informazioni, al client. Questa volta, tuttavia, la risposta viene crittografata nella chiave di sessione che faceva parte del ticket di concessione ticket. In questo modo, non sarà necessario che l'utente immetta nuovamente la password.

Database Kerberos

Fino a questo punto sono state discusse operazioni che richiedono l'accesso in sola lettura al database Kerberos. Queste operazioni vengono eseguite dal servizio di autenticazione, che può essere eseguito sia su computer master che slave.

In questa sezione vengono descritte le operazioni che richiedono l'accesso in scrittura al database. Queste operazioni vengono eseguite dal servizio di amministrazione, denominato KDBM (Kerberos Database Management Service). L'attuale attuazione prevede che le modifiche possano essere apportate solo alla banca dati principale di Kerberos; le copie slave sono di sola lettura. Pertanto, il server KDBM può essere eseguito solo sul computer Kerberos master.

Si noti che, sebbene l'autenticazione possa ancora essere eseguita (sugli slave), le richieste di amministrazione non possono essere elaborate se il computer master è inattivo. In base alla nostra esperienza, ciò non ha posto problemi, in quanto le richieste amministrative sono poco frequenti.

Il KDBM gestisce le richieste di modifica delle password degli utenti. Il lato client di questo

programma, che invia richieste al KDBM tramite la rete, è il programma kpasswd. Il KDBM accetta inoltre le richieste degli amministratori Kerberos, che possono aggiungere entità al database, nonché modificare le password per le entità esistenti. Il lato client del programma di amministrazione, che invia le richieste anche al KDBM attraverso la rete, è il programma kadmin.

Server KDBM

Il server KDBM accetta le richieste di aggiunta di entità al database o di modifica delle password per entità esistenti. Questo servizio è unico in quanto il servizio di concessione ticket non emetterà ticket per esso. È invece necessario utilizzare il servizio di autenticazione (lo stesso utilizzato per ottenere un ticket di concessione ticket). Lo scopo di questa operazione è richiedere all'utente di immettere una password. In caso contrario, se un utente lascia incustodita la propria postazione di lavoro, un passante potrebbe cambiare la propria password, cosa che dovrebbe essere impedita. Allo stesso modo, se un amministratore lascia incustodita la sua workstation, un passante può cambiare qualsiasi password del sistema.

Quando il server KDBM riceve una richiesta, la autorizza confrontando il nome principale autenticato del richiedente della modifica con il nome principale della destinazione della richiesta. Se sono uguali, la richiesta è consentita. In caso contrario, il server KDBM consulta un elenco di controllo di accesso (archiviato in un file nel sistema Kerberos master). Se in questo file viene trovato il nome principale del richiedente, la richiesta è consentita, altrimenti viene negata.

Per convenzione, i nomi con un'istanza NULL (l'istanza predefinita) non vengono visualizzati nel file ACL; viene invece utilizzata un'istanza admin. Per diventare un amministratore di Kerberos, è pertanto necessario creare un'istanza di amministrazione per tale nome utente e aggiungerla all'elenco di controllo di accesso. Questa convenzione consente a un amministratore di utilizzare una password diversa per l'amministrazione Kerberos rispetto a quella utilizzata per il normale accesso.

Vengono registrate tutte le richieste al programma KDBM, consentite o negate.

I programmi kadmin e kpasswd

Gli amministratori di Kerberos utilizzano il programma kadmin per aggiungere entità al database o modificare le password delle entità esistenti. Un amministratore deve immettere la password per il nome dell'istanza dell'amministratore quando richiama il programma kadmin. Questa password viene utilizzata per recuperare un ticket per il server KDBM.

Gli utenti possono modificare le password Kerberos utilizzando il programma kpasswd. Per richiamare il programma, è necessario immettere la vecchia password. Questa password viene utilizzata per recuperare un ticket per il server KDBM.

Replica database Kerberos

Ogni area di autenticazione Kerberos dispone di un computer Kerberos master che ospita la copia master del database di autenticazione. È possibile (anche se non necessario) avere ulteriori copie di sola lettura del database su macchine slave in altre parti del sistema. I vantaggi derivanti dall'utilizzo di più copie del database sono quelli generalmente indicati per la replica: maggiore disponibilità e migliori prestazioni. Se la macchina master non funziona, è comunque possibile eseguire l'autenticazione su una delle macchine slave. La possibilità di eseguire l'autenticazione su uno qualsiasi dei diversi computer riduce la probabilità di un collo di bottiglia sul computer

master.

La conservazione di più copie del database introduce il problema della coerenza dei dati. Abbiamo scoperto che metodi molto semplici sono sufficienti per affrontare le incoerenze. Il database master viene scaricato ogni ora. Il database viene inviato interamente ai computer slave, che aggiornano i propri database. Un programma sull'host master, denominato kprop, invia l'aggiornamento a un programma peer, denominato kpropd, in esecuzione su ciascuno dei computer slave. Il primo kprop invia un checksum del nuovo database che sta per inviare. Il checksum viene crittografato nella chiave del database master Kerberos, in possesso sia del computer Kerberos master che di quello slave. I dati vengono quindi trasferiti sulla rete al kpropd sulla macchina slave. Il server di propagazione slave calcola un checksum dei dati ricevuti e, se corrisponde al checksum inviato dal master, le nuove informazioni vengono utilizzate per aggiornare il database dello slave.

Tutte le password nel database Kerberos sono crittografate nella chiave master del database. Pertanto, le informazioni passate dal master allo slave in rete non sono utili per gli intercettatori. Tuttavia, è essenziale che solo le informazioni provenienti dall'host master siano accettate dagli slave e che venga rilevata la manomissione dei dati, quindi il checksum.

Kerberos dall'esterno che guarda in

In questa sezione viene descritto Kerberos dal punto di vista pratico, dapprima dal punto di vista dell'utente, quindi dal punto di vista del programmatore dell'applicazione e infine attraverso le attività dell'amministratore Kerberos.

Vista dell'utente Kerberos

Se tutto va bene, l'utente difficilmente noterà che Kerberos è presente. Nell'implementazione UNIX, il ticket di concessione ticket viene ottenuto da Kerberos come parte del processo di accesso. La modifica della password Kerberos di un utente fa parte del programma passwd. I ticket Kerberos vengono eliminati automaticamente alla disconnessione di un utente.

Se la sessione di accesso dell'utente dura più a lungo della durata del ticket di concessione ticket (attualmente 8 ore), l'utente noterà la presenza di Kerberos perché alla successiva esecuzione di un'applicazione autenticata da Kerberos, si verificherà un errore. Il ticket Kerberos per tale elemento sarà scaduto. A questo punto, l'utente può eseguire il programma Kinit per ottenere un nuovo ticket per il server di concessione ticket. Come per l'accesso, è necessario fornire una password per ottenerla. Un utente che esegue il comando klist per curiosità può essere sorpreso da tutti i biglietti che sono stati ottenuti in silenzio per suo conto per servizi che richiedono l'autenticazione Kerberos.

Kerberos dal punto di vista dei programmatori

Un programmatore che scrive un'applicazione Kerberos spesso aggiunge l'autenticazione a un'applicazione di rete già esistente costituita da un lato client e server. Chiamiamo questo processo "Kerberizzazione" un programma. La kerberizzazione implica in genere una chiamata alla libreria Kerberos per eseguire l'autenticazione alla richiesta iniziale di servizio. Può inoltre comportare chiamate alla libreria DES per crittografare messaggi e dati che vengono successivamente inviati tra il client dell'applicazione e il server dell'applicazione.

Le funzioni di libreria più utilizzate sono krb_mk_req sul lato client e krb_rd_req sul lato server. La

routine krb_mk_req accetta come parametri il nome, l'istanza e il realm del server di destinazione che verrà richiesto ed eventualmente un checksum dei dati da inviare. Il client invia quindi il messaggio restituito dalla chiamata krb_mk_req attraverso la rete al lato server dell'applicazione. Quando il server riceve questo messaggio, effettua una chiamata alla routine di libreria krb_rd_req. La routine restituisce un giudizio sull'autenticità della presunta identità del mittente.

Se l'applicazione richiede che i messaggi inviati tra il client e il server siano segreti, è possibile effettuare chiamate di libreria a krb_mk_priv (krb_rd_priv) per crittografare (decrittografare) i messaggi nella chiave di sessione condivisa da entrambi i lati.

Processo dell'amministratore Kerberos

Il processo dell'amministratore Kerberos inizia con l'esecuzione di un programma per inizializzare il database. È necessario eseguire un altro programma per registrare entità essenziali nel database, ad esempio il nome dell'amministratore Kerberos con un'istanza di amministrazione. È necessario avviare il server di autenticazione Kerberos e il server di amministrazione. Se esistono database slave, l'amministratore deve disporre che i programmi per la propagazione degli aggiornamenti del database dal master agli slave vengano avviati periodicamente.

Dopo aver eseguito i passaggi iniziali, l'amministratore modifica il database tramite la rete, utilizzando il programma kadmin. Tramite questo programma è possibile aggiungere nuove entità e modificare le password.

In particolare, quando una nuova applicazione Kerberos viene aggiunta al sistema, l'amministratore deve eseguire alcune operazioni per consentirne il funzionamento. Il server deve essere registrato nel database e assegnato a una chiave privata (in genere una chiave casuale generata automaticamente). Alcuni dati, inclusa la chiave del server, devono quindi essere estratti dal database e installati in un file sul computer del server. Il file predefinito è /etc/srvtab. La routine della libreria krb_rd_req richiamata dal server (vedere la sezione precedente) utilizza le informazioni contenute in tale file per decrittografare i messaggi inviati crittografati nella chiave privata del server. Il file /etc/srvtab autentica il server come una password digitata presso un terminale autentica l'utente.

L'amministratore di Kerberos deve inoltre verificare che i computer Kerberos siano fisicamente sicuri e che sia consigliabile mantenere i backup del database master.

La più grande immagine di Kerberos

In questa sezione viene descritto come Kerberos si adatta all'ambiente di Athena, incluso il suo utilizzo da parte di altri servizi e applicazioni di rete, e come interagisce con i reami remoti di Kerberos. Per una descrizione più completa dell'ambiente di Athena, vedi G.W. Treese.

Uso di Kerberos da parte di altri servizi di rete

Diverse applicazioni di rete sono state modificate per l'utilizzo di Kerberos. I comandi rlogin e rsh tentano innanzitutto di eseguire l'autenticazione con Kerberos. Un utente con ticket Kerberos validi può accedere nuovamente a un altro computer Athena senza dover configurare i file .rhosts. Se l'autenticazione Kerberos ha esito negativo, i programmi utilizzano i metodi di autorizzazione tradizionali, in questo caso i file .rhosts.

Il protocollo dell'ufficio postale è stato modificato in modo da utilizzare Kerberos per

l'autenticazione degli utenti che desiderano recuperare la posta elettronica dall'ufficio postale. Un programma di consegna dei messaggi, chiamato Zephyr, è stato recentemente sviluppato ad Athena, e utilizza Kerberos anche per l'autenticazione.

Il programma per l'iscrizione di nuovi utenti, denominato register, utilizza sia il sistema di gestione dei servizi (SMS) che Kerberos. Da SMS, determina se le informazioni inserite dal futuro utente Athena, come il nome e il numero di identificazione MIT, sono valide. Quindi verifica con Kerberos se il nome utente richiesto è univoco. In caso di esito positivo, viene creata una nuova voce nel database Kerberos, contenente il nome utente e la password.

Per una trattazione dettagliata sull'uso di Kerberos per proteggere il Network File System di Sun, fare riferimento all'appendice.

Interazione con altri Kerberi

Èprevisto che diverse organizzazioni amministrative utilizzino Kerberos per l'autenticazione degli utenti. Si prevede inoltre che in molti casi gli utenti di un'organizzazione desiderino utilizzare i servizi di un'altra organizzazione. Kerberos supporta più domini amministrativi. La specifica dei nomi in Kerberos include un campo denominato realm. Questo campo contiene il nome del dominio amministrativo in cui l'utente deve essere autenticato.

I servizi sono in genere registrati in un singolo realm e accettano solo le credenziali rilasciate da un server di autenticazione per tale realm. Un utente è generalmente registrato in un singolo realm (il realm locale), ma è possibile per lei/lui ottenere credenziali emesse da un altro realm (il realm remoto), sulla base dell'autenticazione fornita dal realm locale. Le credenziali valide in un realm remoto indicano il realm in cui l'utente è stato originariamente autenticato. I servizi nel realm remoto possono scegliere se onorare tali credenziali, a seconda del livello di sicurezza richiesto e del livello di fiducia nel realm che inizialmente ha autenticato l'utente.

Per eseguire l'autenticazione tra realm, è necessario che gli amministratori di ogni coppia di realm selezionino una chiave da condividere tra i rispettivi realm. Un utente nel realm locale può quindi richiedere un ticket di concessione ticket al server di autenticazione locale per il server di concessione ticket nel realm remoto. Quando il ticket viene utilizzato, il server remoto per la concessione dei ticket riconosce che la richiesta non proviene dal proprio realm e utilizza la chiave scambiata in precedenza per decrittografare il ticket per la concessione dei ticket. Quindi emette un ticket come di solito, tranne per il fatto che il campo del realm per il client contiene il nome del realm in cui il client è stato originariamente autenticato.

Questo approccio potrebbe essere esteso per consentire di autenticarsi attraverso una serie di realm fino a raggiungere il realm con il servizio desiderato. Per fare questo, tuttavia, sarebbe necessario registrare l'intero percorso preso, e non solo il nome del realm iniziale in cui l'utente è stato autenticato. In una situazione di questo tipo, l'unica cosa nota al server è che A dice che B dice che C dice che l'utente è così e così. Questa istruzione può essere considerata attendibile solo se tutti gli utenti del percorso sono considerati attendibili.

Problemi Kerberos e problemi aperti

Il meccanismo di autenticazione Kerberos presenta diversi problemi e problemi aperti. Tra i problemi da affrontare vi sono le modalità per decidere la durata corretta di un ticket, come consentire i proxy e come garantire l'integrità della workstation.

Il problema della durata dei biglietti consiste nella scelta del giusto compromesso tra sicurezza e convenienza. Se la durata di un ticket è lunga, se un ticket e la relativa chiave di sessione vengono rubati o spostati, possono essere utilizzati per un periodo di tempo più lungo. Tali informazioni possono essere rubate se un utente dimentica di disconnettersi da una postazione di lavoro pubblica. In alternativa, se un utente è stato autenticato su un sistema che consente più utenti, un altro utente con accesso alla radice potrebbe essere in grado di trovare le informazioni necessarie per utilizzare i ticket rubati. Tuttavia, il problema legato all'assegnazione di una breve durata a un ticket è che alla scadenza di quest'ultimo l'utente dovrà ottenerne una nuova che gli richieda di immettere nuovamente la password.

Un problema aperto è il problema del proxy. In che modo un utente autenticato può consentire a un server di acquisire altri servizi di rete per suo conto? Un esempio importante è l'utilizzo di un servizio che acceda ai file protetti direttamente da un file server. Un altro esempio di questo problema è il cosiddetto inoltro dell'autenticazione. Se un utente accede a una workstation e a un host remoto, è preferibile che disponga dell'accesso agli stessi servizi disponibili localmente durante l'esecuzione di un programma sull'host remoto. Ciò che rende questo difficile è che l'utente potrebbe non considerare attendibile l'host remoto, quindi l'inoltro dell'autenticazione non è desiderabile in tutti i casi. Attualmente non abbiamo una soluzione a questo problema.

Un altro problema, importante nell'ambiente Athena, è come garantire l'integrità del software in esecuzione su una workstation. Non si tratta di un problema nelle workstation private, poiché l'utente che le utilizzerà ha il controllo su di esso. Nelle workstation pubbliche, tuttavia, è possibile che qualcuno sia venuto con sé e abbia modificato il programma di login per salvare la password dell'utente. L'unica soluzione attualmente disponibile nel nostro ambiente è rendere difficile per gli utenti modificare il software in esecuzione sulle workstation pubbliche. Una soluzione migliore richiederebbe che la chiave dell'utente non lasci mai un sistema che l'utente sa essere affidabile. Una soluzione potrebbe essere quella di disporre di una smart card in grado di eseguire la crittografia richiesta nel protocollo di autenticazione.

Stato Kerberos

Una versione prototipo di Kerberos entrò in produzione nel settembre del 1986. Dal gennaio del 1987, Kerberos è l'unico mezzo utilizzato dal progetto Athena per autenticare i suoi 5.000 utenti, 650 workstation e 65 server. Inoltre, Kerberos è ora utilizzato al posto dei file .rhosts per controllare l'accesso in diversi sistemi di condivisione del tempo di Athena.

Riconoscimenti Kerberos

Kerberos è stato inizialmente progettato da Steve Miller e Clifford Neuman con suggerimenti da Jeff Schiller e Jerry Saltzer. Da allora, molte altre persone sono state coinvolte nel progetto. Tra loro ci sono 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 e Stan Zanarotti.

Siamo grati a Dan Geer, Kathy Lieben, Josh Lubarr, Ken Raeburn, Jerry Saltzer, Ed Steiner, Robbert van Renesse, e Win Treese, i cui suggerimenti hanno migliorato notevolmente le bozze precedenti di questo articolo.

Jedlinsky, J.T. Kohl e W.E. Sommerfeld, "The Zephyr Notification System", in Usenix Conference Procedures (Inverno, 1988).

M.A. Rosenstein, D.E. Geer, e P.J. Levine, in Usenix Conference Procedures (Winter, 1988).

R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh e B. Lyon, "Design and Implementation of the Sun Network Filesystem", in Usenix Conference Procedures (Estate, 1985).

Appendice Applicazione Kerberos al Network File System (NFS) di SUN

Un componente chiave del sistema di workstation Project Athena è l'interposizione della rete tra la workstation dell'utente e il suo archivio di file privato (home directory). Tutto lo storage privato risiede su un set di computer (attualmente VAX 11/750) dedicati a questo scopo. Questo ci consente di offrire servizi su workstation UNIX disponibili pubblicamente. Quando un utente accede a una di queste workstation disponibili pubblicamente, invece di convalidare il proprio nome e la propria password in base a un file di password residente localmente, utilizziamo Kerberos per determinare la sua autenticità. Il programma di login richiede un nome utente (come su qualsiasi sistema UNIX). Questo nome utente viene utilizzato per recuperare un ticket di concessione ticket Kerberos. Il programma di login utilizza la password per generare una chiave DES per decrittografare il ticket. Se la decrittografia ha esito positivo, la home directory dell'utente viene individuata consultando il servizio di denominazione Hesiod e montata tramite NFS. Il programma di accesso passa quindi il controllo sulla shell dell'utente, che può quindi eseguire i tradizionali file di personalizzazione per utente poiché la home directory è ora "collegata" alla workstation. Il servizio Hesiod viene inoltre utilizzato per creare una voce nel file di password locale. (a vantaggio dei programmi che cercano informazioni in /etc/passwd).

Tra le varie opzioni per la fornitura di servizi file remoti, abbiamo scelto Sun's Network File System. Tuttavia questo sistema non riesce a conciliarsi con le nostre esigenze in modo cruciale. NFS presume che tutte le workstation rientrino in due categorie (dal punto di vista di un file server): attendibili e non attendibili. I sistemi non attendibili non possono accedere ai file. I sistemi affidabili sono totalmente affidabili. Si presuppone che un sistema attendibile sia gestito da una gestione intuitiva. In particolare, da una workstation affidabile è possibile mascherare qualsiasi utente valido del file service system e ottenere così l'accesso a quasi tutti i file del sistema. (Sono esentati solo i file di proprietà di "root".)

Nel nostro ambiente, la gestione di una workstation (nel senso tradizionale della gestione dei sistemi UNIX) è nelle mani dell'utente che la sta utilizzando. Non facciamo alcun segreto della password principale sulle nostre workstation, in quanto ci rendiamo conto che un utente davvero poco amichevole può sfondare dal fatto che si trova nella stessa posizione fisica della macchina e ha accesso a tutte le funzioni della console. Non possiamo quindi fidarci delle nostre workstation nell'interpretazione NFS della fiducia. Per consentire controlli di accesso appropriati nel nostro ambiente abbiamo dovuto apportare alcune modifiche al software NFS di base e integrare Kerberos nello schema.

Kerberos NFS non modificato

Nell'implementazione di NFS con cui abbiamo iniziato (presso l'Università del Wisconsin), l'autenticazione è stata fornita sotto forma di una parte di dati inclusi in ogni richiesta NFS (chiamata "credenziale" nella terminologia NFS). Questa credenziale contiene informazioni sull'identificatore utente univoco (UID) del richiedente e un elenco degli identificatori di gruppo (GID) dell'appartenenza del richiedente. Queste informazioni vengono quindi utilizzate dal server NFS per il controllo degli accessi. La differenza tra una workstation trusted e una non trusted consiste nel fatto che le relative credenziali vengano o meno accettate dal server NFS.

Kerberos: NFS modificato

Nel nostro ambiente, i server NFS devono accettare credenziali da una workstation se e solo se le credenziali indicano l'UID dell'utente della workstation e nessun altro.

Una soluzione ovvia sarebbe quella di modificare la natura delle credenziali da semplici indicatori di UID e GID a dati autenticati Kerberos completi. Tuttavia, se questa soluzione fosse adottata, verrebbe pagata una penale significativa. Le credenziali vengono scambiate in ogni operazione NFS, incluse tutte le attività di lettura e scrittura su disco. L'inclusione dell'autenticazione Kerberos su ciascuna transazione su disco aggiungerebbe un numero ragionevole di cifrature complete (eseguite nel software) per transazione e, secondo i nostri calcoli di busta, avrebbe fornito prestazioni inaccettabili. Sarebbe stato inoltre necessario inserire le routine della libreria Kerberos nello spazio degli indirizzi del kernel.

Avevamo bisogno di un approccio ibrido, come descritto di seguito. L'idea di base è quella di far sì che il server NFS esegua il mapping delle credenziali ricevute dalle workstation client a una credenziale valida (e probabilmente diversa) nel sistema server. Questa mappatura viene eseguita nel kernel del server su ciascuna transazione NFS e viene impostata al momento del "mount" da un processo a livello utente che esegue l'autenticazione moderata Kerberos prima di stabilire un mapping valido delle credenziali del kernel.

Per implementare questo abbiamo aggiunto una nuova chiamata di sistema al kernel (richiesta solo sui sistemi server, non sui sistemi client) che fornisce il controllo della funzione di mapping che mappa le credenziali in ingresso dalle workstation client alle credenziali valide per l'uso sul server (se presenti). La funzione di mappatura di base mappa la tupla:

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

a una credenziale NFS valida nel sistema server. L'indirizzo IP-CLIENT viene estratto dal pacchetto di richiesta NFS fornito dal sistema client. Nota: tutte le informazioni nelle credenziali generate dal client, ad eccezione di UID-ON-CLIENT, vengono eliminate.

Se non esiste alcun mapping, il server reagisce in uno dei due modi seguenti, a seconda della configurazione. Nella configurazione semplice, le richieste non mappabili vengono inserite automaticamente nelle credenziali dell'utente "none", che non dispone di alcun accesso privilegiato e che dispone di un UID univoco. I server non descrittivi restituiscono un errore di accesso NFS quando non è possibile trovare un mapping valido per una credenziale NFS in ingresso.

La nuova chiamata di sistema viene utilizzata per aggiungere ed eliminare voci dalla mappa residente del kernel. Consente inoltre di scaricare tutte le voci mappate a un UID specifico sul sistema server o tutte le voci da un determinato INDIRIZZO IP CLIENT.

Il daemon di montaggio (che gestisce le richieste di montaggio NFS sui sistemi server) è stato modificato per accettare un nuovo tipo di transazione, la richiesta di mapping dell'autenticazione Kerberos. Fondamentalmente, come parte del processo di installazione, il sistema client fornisce un autenticatore Kerberos insieme a un'indicazione del suo UID-ON-CLIENT (criptato nell'autenticatore Kerberos) sulla workstation. Il daemon di montaggio del server converte il nome principale Kerberos in un nome utente locale. Questo nome utente viene quindi cercato in un file speciale per ottenere l'elenco UID e GID dell'utente. Per una maggiore efficienza, questo file è un database ndbm con il nome utente come chiave. Da queste informazioni, una credenziale NFS viene costruita e consegnata al kernel come mapping valido della tupla <CLIENT-IP-ADDRESS,

CLIENT-UID> per questa richiesta.

Al momento dello smontaggio viene inviata una richiesta al daemon di montaggio per rimuovere il mapping aggiunto in precedenza dal kernel. È inoltre possibile inviare una richiesta al momento della disconnessione per invalidare tutti i mapping per l'utente corrente sul server in questione, eliminando in tal modo tutti i mapping rimanenti esistenti (anche se non dovrebbero) prima che la workstation sia resa disponibile per l'utente successivo.

Implicazioni della protezione Kerberos per il NFS modificato

Questa implementazione non è completamente sicura. Innanzitutto, i dati utente vengono ancora inviati in rete in formato non crittografato e quindi intercettabile. L'autenticazione di basso livello per singola transazione si basa su una coppia <CLIENT-IP-ADDRESS, CLIENT-UID> fornita non crittografata nel pacchetto della richiesta. Queste informazioni potrebbero essere falsificate e quindi la sicurezza compromessa. Tuttavia, si noti che solo mentre un utente sta utilizzando attivamente i suoi file (cioè, mentre è connesso) sono mappature valide in atto e quindi questa forma di attacco è limitata a quando l'utente in questione è connesso. Quando un utente non ha effettuato l'accesso, la falsificazione dell'indirizzo IP non consentirà l'accesso non autorizzato ai suoi file.

Riferimenti Kerberos

1. S.P. Miller, B.C. Neuman, J.I. Schiller e J.H. Saltzer, sezione E.2.1: Kerberos Authentication and Authorization System, M.I.T. Project Athena, Cambridge, Massachusetts (21 dicembre 1987).
2. E. Balkovich, S.R. Lerman e R.P. Parmelee, "Computing in Higher Education: The Athena Experience," Comunicazioni dell'ACM, vol. 28(11), pp. 1214-1224, ACM (novembre 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 (dicembre 1978).
4. V.L. Voydock and S.T. Kent, "Security Mechanism in High-Level Network Protocols," Computing Surveys, vol. 15(2), ACM (giugno 1983).
5. National Bureau of Standards, "Data Encryption Standard," Federal Information Processing Standards Publication 46, Ufficio Stampa Governativo, Washington, DC (1977).
6. SP Dyer, "Hesiod", in Usenix Conference Procedures (Inverno, 1988).
7. W.J. Bryant, Tutorial dei programmatori Kerberos, MIT Project Athena (in preparazione).
8. W.J. Bryant, Kerberos Administrator's Manual, MIT Project Athena (in preparazione).
9. G.W. Treese, "Berkeley Unix on 1000 Workstation: Athena diventa 4.3BSD," in Usenix Conference Procedures (inverno, 1988).
10. C.A. DellaFera, M.W. Eichin, R.S. French, D.C. Jedlinsky, J.T. Kohl e W.E. Sommerfeld, "The Zephyr Notification System", in Usenix Conference Procedures (Winter, 1988).
11. M.A. Rosenstein, D.E. Geer, e P.J. Levine, in Usenix Conference Procedures (Winter, 1988).
12. R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh e B. Lyon, "Design and Implementation of the Sun Network Filesystem", in Usenix Conference Procedures (Estate, 1985).

Informazioni correlate

- Pagina di supporto Kerberos
- Documentazione e supporto tecnico – Cisco Systems