Kerberos - Overzicht - een verificatieservice voor Open Network Systems

Inhoud

Inleiding

In een open netwerk computeromgeving kan een werkstation niet worden vertrouwd om zijn gebruikers correct te identificeren met netwerkdiensten. Kerberos biedt een alternatieve benadering waarbij een vertrouwde dienst voor de echtheidscontrole van derden wordt gebruikt om de identiteit van gebruikers te controleren. Dit document geeft een overzicht van het Kerberos-verificatiemodel zoals dat voor het MIT-project Athena wordt toegepast. Het beschrijft de protocollen die door klanten, servers en Kerberos worden gebruikt om authenticatie te bereiken. Het beschrijft ook het beheer en de replicatie van de vereiste gegevensbank. De visies van Kerberos zoals die door de gebruiker, programmeur en beheerder worden gezien, worden beschreven. Ten slotte wordt de rol van Kerberos in het grotere Athena-beeld gegeven, samen met een lijst van toepassingen die momenteel Kerberos voor gebruikersauthenticatie gebruiken. We beschrijven de toevoeging van Kerberos-verificatie aan het Sun Network File System als een casestudy voor het integreren van Kerberos met een bestaande toepassing.

Kerberos-auteurs

- 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 was lid van de staf van Project Athena tijdens de ontwerp- en initiële implementatiefase van Kerberos.
- Jeffrey I. Schiller, Project Athena, Massachusetts Institute of Technology, Cambridge, MA 02139, jis@ATHENA.MIT.EDU

Inleiding tot Kerberos

Dit document geeft een overzicht van Kerberos, een door Miller en Neuman ontworpen authenticatiesysteem. voor open netwerk computeromgevingen, en beschrijft onze ervaring met het gebruik ervan in het Project Athena van het MIT. In de paragraaf over Motivation, verklaren we waarom er een nieuw authenticatiemodel nodig is voor open netwerken en wat de vereisten zijn. De Wat is Kerberos? In de sectie worden de componenten van de Kerberos-software opgesomd en wordt beschreven hoe zij samenwerken bij het verlenen van de authenticatiedienst. In het vak Kerberos Names beschrijven we het naamgevingsschema van Kerberos.

Hoe Kerberos Works de bouwstenen van Kerberos authenticatie presenteert - het ticket en de authenticator. Dit leidt tot een discussie over de twee echtheidsprotocollen: de eerste authenticatie van een gebruiker aan Kerberos (analoog aan houtkap) en het protocol voor wederzijdse authenticatie van een potentiële consument en een potentiële producent van een netwerkdienst.

Kerberos vereist een gegevensbank met informatie over haar klanten; in de sectie van de Kerberos Database worden de gegevensbank, het beheer ervan en het protocol voor de wijziging ervan beschreven. De Kerberos van de BUITENKANT die in sectie kijkt beschrijft de interface van Kerberos aan zijn gebruikers, toepassingsprogrammeurs, en beheerders. In het gedeelte"Beeld groter" beschrijven we hoe het Project Athena Kerberos in de rest van de omgeving van Athena past. We beschrijven ook de interactie van verschillende Kerberos-authenticatiedomeinen, of gebieden; in ons geval, de relatie tussen het project Athena Kerberos en de Kerberos die in het laboratorium voor computerwetenschap van het MIT lopen.

In het gedeelte Problemen en openstaande problemen noemen we openstaande kwesties en problemen die nog niet zijn opgelost. De laatste paragraaf geeft de huidige status van Kerberos in Project Athena. In het Aanhangsel beschrijven we in detail hoe Kerberos op een dienst van het netwerkbestand wordt toegepast om gebruikers die toegang tot verre bestandssystemen willen verkrijgen voor authentiek te verklaren.

Kerberos-concepten

In dit artikel gebruiken we termen die dubbelzinnig, nieuw voor de lezer zijn of die anders elders worden gebruikt. Hieronder staat ons gebruik van die termen.

Gebruiker, client, server-Door gebruiker betekenen we een mens die een programma of service gebruikt. Een cliënt gebruikt ook iets, maar is niet noodzakelijkerwijs een persoon; het kan een programma zijn . Vaak bestaan netwerktoepassingen uit twee delen; een programma dat op één machine draait en op een externe service vraagt, en een ander programma dat op de afstandsmachine draait en die service uitvoert. We noemen die de client kant en server kant van de toepassing, respectievelijk. Vaak zal een client contact opnemen met een server namens een gebruiker.

Elke entiteit die het Kerberos-systeem gebruikt, of het nu een gebruiker of een netwerkserver is, is in één opzicht een client, omdat het de Kerberos-service gebruikt. Om klanten van Kerberos van klanten van andere services te onderscheiden gebruiken we de term main om zo een entiteit aan te duiden. Merk op dat een hoofd van Kerberos of een gebruiker of een server kan zijn. (In een latere sectie beschrijven we de naam van Kerberos-hoofdrolspelers.)

Service vs. Server—We gebruiken service als een abstracte specificatie van bepaalde acties die moeten worden uitgevoerd. Een proces dat deze acties uitvoert wordt een server genoemd. Op een gegeven moment kunnen er meerdere servers zijn die een bepaalde service uitvoeren (meestal op verschillende machines). Bijvoorbeeld, in Athena is er één BSD UNIX rlogin server actief op elk van onze timesharing machines.

Belangrijkste, Private Key, Wachtwoord-Kerberos gebruikt privé sleutelencryptie. Elk hoofd van Kerberos wordt toegewezen een groot aantal, zijn privé sleutel, die slechts aan dat hoofd en Kerberos bekend is. Bij een gebruiker is de privé-toets het resultaat van een eenrichtingsfunctie die op het wachtwoord van de gebruiker is toegepast. We gebruiken sleutel als kortstondig voor privésleutel.

Credentials - Helaas, heeft dit woord een speciale betekenis voor zowel het systeem van het Netwerk van de Zon als het systeem van Kerberos. We vermelden expliciet of we NFS-referenties of Kerberos-referenties bedoelen, anders wordt de term gebruikt in de normale Engelse taalzin.

Master en Slave-Het is mogelijk om Kerberos authenticatiesoftware te starten op meer dan één machine. Er is echter altijd maar één definitieve kopie van de Kerberos-databank. De machine die deze database opslaat, heet de meestermachine, of alleen de meester. Andere machines kunnen alleen-lezen exemplaren van de Kerberos-databank bezitten, en deze worden slaven genoemd.

Motivering achter Kerberos

In een niet-netwerkgebonden PC-omgeving kunnen hulpbronnen en informatie worden beschermd door de PC fysiek te beveiligen. In een tijdrovende computeromgeving beschermt het besturingssysteem gebruikers tegen elkaar en controleert het de beschikbare bronnen. Om te

bepalen wat elke gebruiker kan lezen of wijzigen, is het nodig dat het timesharing systeem elke gebruiker identificeert. Dit wordt bereikt wanneer de gebruiker inlogt.

In een netwerk van gebruikers die services van veel afzonderlijke computers nodig hebben, zijn er drie benaderingen die je kunt volgen om toegang te krijgen tot controle: Je kunt niets doen, vertrouwen op de machine waartoe de gebruiker is aangemeld om ongeoorloofde toegang te voorkomen. men kan van de gastheer verlangen dat hij zijn identiteit aantoont , maar het woord van de gastheer vertrouwen over wie de gebruiker is ; of iemand kan van de gebruiker eisen dat hij/zij zijn identiteit voor elke vereiste service aantoont.

In een gesloten omgeving waar alle machines onder strikte controle staan, kan men de eerste aanpak gebruiken. Wanneer de organisatie alle hosts die over het netwerk communiceren controleert, is dit een redelijke benadering.

In een opener omgeving zou je selectief alleen die gastheren onder organisatorische controle kunnen vertrouwen. In dat geval moet van elke gastheer worden geëist dat hij zijn identiteit aantoont. Deze benadering wordt toegepast op de rlogin- en rsh-programma's. In deze protocollen wordt de authenticatie uitgevoerd door het internetadres te controleren waar een verbinding is tot stand gebracht.

In het Athene-milieu moeten wij de verzoeken van niet onder organisatorische controle staande gastinstellingen kunnen honoreren. Gebruikers hebben volledige controle over hun werkstations: ze kunnen ze rebooten , ze op zichzelf zetten of zelfs hun eigen tapes opstapelen . Daarom moet de derde aanpak worden gevolgd. de gebruiker moet voor elke gewenste service zijn identiteit bewijzen. De server moet ook zijn identiteit bewijzen. Het is niet voldoende de host die een netwerkserver runt fysiek te beveiligen; iemand elders in het netwerk kan zich als de gegeven server voorstellen .

Ons milieu stelt verschillende eisen aan een identificatiemechanisme. Ten eerste moet het veilig zijn. Het omzeilen van het systeem moet moeilijk genoeg zijn dat een potentiële aanvaller het authenticatiemechanisme niet als zwakke schakel ziet. Iemand die het netwerk bekijkt, zou de informatie niet kunnen verkrijgen die nodig is om een andere gebruiker na te doen. Ten tweede moet het betrouwbaar zijn. Toegang tot veel diensten zal afhangen van de authenticatiedienst. Als het niet betrouwbaar is, zal het systeem van diensten als geheel niet bestaan. Ten derde moet het transparant zijn. Idealiter zou de gebruiker niet op de hoogte moeten zijn van de authenticatie die plaatsvindt. Tenslotte zou het schaalbaar moeten zijn. Veel systemen kunnen communiceren met Athena-gastheren. Niet al deze factoren zullen ons mechanisme ondersteunen, maar de software mag niet breken als ze dat deden.

Kerberos is het resultaat van ons werk om aan de bovenstaande eisen te voldoen. Wanneer een gebruiker naar een werkstation loopt, logt hij in. Voor zover de gebruiker kan vertellen, is deze eerste identificatie voldoende om te bewijzen dat de gebruiker een van de vereiste netwerkservers is voor de duur van de inlogsessie. De beveiliging van Kerberos is afhankelijk van de beveiliging van verschillende authenticatieservers, maar niet van het systeem waarvan gebruikers inloggen, noch van de beveiliging van de eindservers die zullen worden gebruikt. De authenticatieserver biedt een naar behoren geauthentiseerde gebruiker een manier om haar/zijn identiteit te bewijzen aan servers die verspreid over het netwerk zijn.

Verificatie is een fundamentele bouwsteen voor een veilige netwerkomgeving. Als een server bijvoorbeeld weet dat een klant een bepaalde identiteit heeft, kan hij beslissen of hij de service aanbiedt, of de gebruiker speciale rechten zou moeten krijgen, wie de rekening voor de service zou moeten ontvangen, enzovoort. Met andere woorden, vergunningverlenings- en boekhoudsystemen kunnen worden gebouwd op basis van de door Kerberos geboden

authenticatie, hetgeen leidt tot een beveiliging die gelijkwaardig is aan die van de enige PC of het timesharing systeem.

Wat is Kerberos?

Kerberos is een betrouwbare dienst voor de authenticatie van derden op basis van het model dat door Needham en Schroeder wordt gepresenteerd. Het wordt vertrouwd in de zin dat elk van zijn klanten Kerberos's oordeel over de identiteit van elk van zijn andere klanten als accuraat beschouwt. Time-postzegels (grote aantallen die de huidige datum en tijd vertegenwoordigen) zijn aan het oorspronkelijke model toegevoegd om te helpen bij het detecteren van terugspelen. Replay gebeurt wanneer een bericht van het netwerk wordt gestolen en later verschijnt. Zie Voydock en Kent voor een volledi ger beschrijving van de replay en andere kwesties met betrekking tot authenticatie.

Wat doet Kerberos?

Kerberos houdt een database bij van zijn klanten en hun privésleutels. De privé-toets is een groot aantal dat alleen bekend is bij Kerberos en de client waartoe het behoort. Als de client een gebruiker is, is het een versleuteld wachtwoord. Netwerkdiensten waarvoor een verificatieregister met Kerberos is vereist, evenals klanten die deze diensten willen gebruiken. Over de privé-toetsen wordt onderhandeld bij registratie.

Omdat Kerberos deze privé-sleutels kent, kan het berichten maken die de ene cliënt ervan overtuigen dat de andere werkelijk is wie het is. Kerberos genereert ook tijdelijke privé-sleutels, genaamd sessiekleutels, die aan twee klanten worden gegeven en aan niemand anders. Een sessiesleutel kan worden gebruikt om berichten tussen twee partijen te versleutelen.

Kerberos biedt drie verschillende beschermingsniveaus. Volgens de eisen van de aanvraag bepaalt de programmeur van de aanvraag wat passend is. Bijvoorbeeld, sommige toepassingen vereisen slechts dat de authenticiteit bij het initiëren van een netwerkverbinding wordt gevestigd, en kunnen ervan uitgaan dat de verdere berichten van een bepaald netwerkadres van de echt verklaarde partij voortkomen. Ons geautomatiseerde netwerkbestandssysteem gebruikt dit beveiligingsniveau.

Andere toepassingen vereisen verificatie van elk bericht, maar geven er niet toe of de inhoud van het bericht wordt bekendgemaakt of niet. Voor deze, verstrekt Kerberos veilige berichten. Toch wordt er een hoger beveiligingsniveau geboden door privé-berichten, waarin elk bericht niet alleen voor authentiek is, maar ook versleuteld. Particuliere berichten worden bijvoorbeeld door de Kerberos-server zelf gebruikt om wachtwoorden via het netwerk te verzenden.

Kerberos-softwarecomponenten

De uitvoering van Athena omvat verschillende modules:

- Kerberos-toepassingsbibliotheek
- encryptie
- database
- administratieprogramma ' s voor databanken
- beheerserver
- verificatieserver

- DBB-propagatiesoftware
- gebruikersprogramma's
- toepassingen

De Kerberos-toepassingsbibliotheek biedt een interface voor toepassingsklanten en toepassingsservers. Het bevat onder meer routines voor het creëren of lezen van authenticatieverzoeken, en de routines voor het creëren van veilige of privéberichten.

De encryptie in Kerberos is gebaseerd op DES, de standaard voor gegevensversleuteling. De encryptiebibliotheek implementeert die routines. Er worden verschillende versleutelingsmethoden geboden, met wisselwerking tussen snelheid en veiligheid. Er wordt ook een uitbreiding naar de DES Cipher Block Chaining (CBC)-modus, de propagerende CBC-modus genaamd, geboden. In de CBC wordt een fout alleen door het huidige blok van het algoritme verspreid terwijl in de PCBC de fout in het hele bericht wordt verspreid. Dit maakt het hele bericht nutteloos als er een fout optreedt, in plaats van slechts een deel ervan. De encryptiebibliotheek is een onafhankelijke module, en kan worden vervangen door andere DES-implementaties of een andere encryptiebibliotheek.

Een andere vervangbare module is het gegevensbeheersysteem. De huidige implementatie in Athena van de databank gebruikt ndbm, hoewel Ingres oorspronkelijk werd gebruikt. Ook andere bibliotheken voor gegevensbeheer kunnen worden gebruikt.

De behoeften van de Kerberos-gegevensbank zijn eenvoudig; voor elke aangever wordt een register bijgehouden met de naam , de privésleutel en de vervaldatum van de aangever , samen met enige administratieve informatie . (De vervaldatum is de datum waarna een vermelding niet langer geldig is. Bij registratie is deze instelling meestal ingesteld op een paar jaar in de toekomst.)

Andere gebruikersinformatie, zoals echte naam, telefoonnummer, enzovoort, wordt bewaard door een andere server, de Hesiod nameserver. Op deze manier kan gevoelige informatie, met name wachtwoorden, door Kerberos worden verwerkt met behulp van vrij hoge veiligheidsmaatregelen; dat de niet-gevoelige informatie van Hesiod anders wordt behandeld; het kan bijvoorbeeld niet versleuteld worden via het netwerk .

De Kerberos-servers gebruiken de database bibliotheek, net als de tools voor het beheer van de database.

De beheerserver (of KDBM-server) biedt een read-Writnetwerkinterface naar de database. De clientkant van het programma kan op elke machine van het netwerk worden uitgevoerd. De serverkant moet echter in de machine waarin de Kerberos-database is ondergebracht draaien om wijzigingen in de database aan te brengen.

De authenticatieserver (of Kerberos server) daarentegen voert alleen-lezen bewerkingen uit op de Kerberos-database, namelijk de authenticatie van hoofden en de generatie sessiesleutels. Aangezien deze server de Kerberos database niet wijzigt, kan deze op een machine met alleen-lezen kopie van de master Kerberos-database draaien.

Databaseverpropagatiesoftware beheert replicatie van de Kerberos-database. Het is mogelijk om kopieën van de database te hebben op verschillende machines, waarbij een kopie van de authenticatieserver op elke machine wordt uitgevoerd. Elk van deze slavemachines ontvangt een update van de Kerberos database van de master machine met bepaalde tussenpozen.

Tenslotte zijn er programma's voor eindgebruikers om in te loggen op Kerberos, een Kerberos-

wachtwoord te wijzigen en Kerberos-tickets te tonen of te vernietigen (tickets worden later uitgelegd).

Kerberos-namen

Een deel van het echt maken van een entiteit noemt het. Het proces van authenticatie is de verificatie dat de cliënt degene is die in een verzoek wordt genoemd. Waaruit bestaat een naam? In Kerberos worden zowel gebruikers als servers genoemd. Wat de authenticatieserver betreft, zij zijn gelijkwaardig. Een naam bestaat uit een voornaam, een instantie en een gebied, uitgedrukt als name.instance@realm.

De primaire naam is de naam van de gebruiker of de dienst. De instantie wordt gebruikt om onderscheid te maken tussen variaties op de primaire naam. Voor gebruikers kan een voorbeeld speciale privileges opleveren, zoals de 'root'- of 'admin'-instanties. Voor diensten in de omgeving van Athena is de instantie gewoonlijk de naam van de machine waarop de server draait. De rlogin service heeft bijvoorbeeld verschillende instanties op verschillende hosts: rlogin.priam is de rlogin server op de host die priam heet. Een Kerberos-ticket is alleen goed voor één server met naam. Als zodanig is een afzonderlijk ticket nodig om toegang te krijgen tot verschillende gevallen van dezelfde dienst. Het gebied is de naam van een administratieve entiteit die verificatiegegevens handhaaft. Bijvoorbeeld, verschillende instituties kunnen elk hun eigen Kerberos machine hebben, die een verschillende gegevensbank huisvesten. Ze hebben verschillende Kerberos-gebieden. (De resultaten worden verder besproken in de interactie met andere Kerberi.)

Hoe Kerberos werkt

In deze sectie worden de Kerberos-verificatieprotocollen beschreven. Zoals hierboven vermeld, is het Kerberos-verificatiemodel gebaseerd op het distributiprotocol van de Needham en Schroeder. Wanneer een gebruiker om een dienst verzoekt, moet zijn identiteit worden vastgesteld. Om dit te doen wordt er een ticket naar de server aangeboden, samen met het bewijs dat het ticket oorspronkelijk was afgegeven aan de gebruiker, niet gestolen. Er zijn drie fasen voor de authenticatie door Kerberos. In de eerste fase krijgt de gebruiker aanmeldingsgegevens die gebruikt moeten worden om toegang tot andere diensten te vragen. In de tweede fase vraagt de gebruiker om authenticatie voor een specifieke dienst. In de laatste fase presenteert de gebruiker deze aanmeldingsgegevens aan de eindserver.

Kerberos Credentials

Er zijn twee soorten geloofsbrieven gebruikt in het Kerberos authenticatiemodel: kaartjes en authenticatoren. Beide zijn gebaseerd op privé-sleutelencryptie, maar worden versleuteld met verschillende toetsen. Een ticket wordt gebruikt om de identiteit door te geven van de persoon aan wie het ticket is afgegeven tussen de authenticatieserver en de eindserver. Een ticket geeft ook informatie door die kan worden gebruikt om te verzekeren dat de persoon die het ticket gebruikt dezelfde persoon is waaraan het werd afgegeven. De authenticator bevat de aanvullende informatie die, vergeleken met de informatie in het ticket, aantoont dat de klant die het biljet voorstelt, dezelfde is als het ticket.

Een ticket is goed voor één server en één cliënt. Het bevat de naam van de server, de naam van de client, het internetadres van de client, een tijdstempel, een leven en een willekeurige sessiesleutel. Deze informatie wordt versleuteld met behulp van de sleutel van de server waarvoor het ticket wordt gebruikt. Zodra het ticket is afgegeven, kan het meerdere keren door de

genoemde client gebruikt worden om toegang te krijgen tot de genoemde server, tot het ticket vervalt. Merk op dat omdat het ticket versleuteld is in de sleutel van de server, het veilig is om de gebruiker toe te staan om het ticket naar de server door te geven zonder zich zorgen te hoeven maken over de gebruiker die het ticket wijzigt.

In tegenstelling tot het ticket kan de authenticator slechts eenmaal gebruikt worden. Er moet een nieuwe gegenereerd worden telkens wanneer een cliënt een dienst wil gebruiken. Dit levert geen probleem op omdat de cliënt zelf de authenticator kan bouwen. Een authenticator bevat de naam van de client, het IP-adres van het werkstation en de huidige werkstationtijd. De authenticator is versleuteld in de sessiesleutel die deel uitmaakt van het ticket.

Pak de eerste Kerberos-tijket

Wanneer de gebruiker op een werkstation aankomt, kan slechts één stukje informatie haar/zijn identiteit bewijzen: het wachtwoord van de gebruiker. De eerste uitwisseling met de authenticatieserver is ontworpen om de kans dat het wachtwoord wordt gecompromitteerd tot een minimum te beperken, terwijl een gebruiker tegelijkertijd niet in staat is om haar/zichzelf zonder kennis van dat wachtwoord op de juiste manier te authentificeren. Het proces van het loggen in lijkt voor de gebruiker hetzelfde te zijn als loggen in een timesharing-systeem. Achter de schermen echter is het heel anders.

De gebruiker wordt gevraagd om een andere gebruikersnaam. Zodra het is ingevoerd, wordt een verzoek naar de authenticatieserver gestuurd met de naam van de gebruiker en de naam van een speciale dienst die bekend staat als de dienst die de tickets verleent.

De verificatieserver controleert de client. Als dat zo is, genereert het een willekeurige sessiesleutel die later zal worden gebruikt tussen de client en de server die het ticket toekent. Het creëert vervolgens een ticket voor de server die de naam van de klant bevat, de naam van de server die het ticket toekent, de huidige tijd, een leven voor het ticket, het IP-adres van de klant en de willekeurige sessiesleutel die zojuist is gemaakt. Dit is allemaal versleuteld in een sleutel die alleen bekend is bij de server die het ticket verleent en de verificatieserver.

De authenticatieserver stuurt dan het ticket, samen met een kopie van de willekeurige sessiesleutel en enige extra informatie, terug naar de client. Deze reactie is versleuteld in de privésleutel van de client, alleen bekend bij Kerberos en de client, die is afgeleid van het wachtwoord van de gebruiker.

Wanneer de klant de reactie heeft ontvangen, wordt de gebruiker om het wachtwoord gevraagd. Het wachtwoord wordt geconverteerd naar een DES-toets en gebruikt om de respons te decrypteren van de authenticatieserver. Het ticket, de sessiesleutel en een deel van de andere informatie worden opgeslagen voor toekomstig gebruik, en de wachtwoord en DES-toets van de gebruiker worden uit het geheugen gewist.

Nadat de uitwisseling is voltooid, beschikt het werkstation over informatie die het kan gebruiken om de identiteit van zijn gebruiker te bewijzen gedurende de levensduur van het ticket dat het heeft toegewezen. Zolang de software op het werkstation niet eerder is geknoeid, bestaat er geen informatie die iemand anders in staat zal stellen zich te imiteren tot buiten de levensduur van het ticket.

Een Kerberos-service aanvragen

Op dit moment doen we net of de gebruiker al een ticket heeft voor de gewenste server. Om

toegang tot de server te krijgen, bouwt de toepassing een authenticator met de naam en IP adres van de cliënt, en de huidige tijd. De authenticator wordt dan versleuteld in de sessiesleutel die met het ticket voor de server is ontvangen. De client stuurt vervolgens de authenticator samen met het ticket naar de server op een manier die door de individuele toepassing wordt gedefinieerd.

Nadat de authenticator en het ticket door de server zijn ontvangen, decrypteert de server het ticket, gebruikt de sessiesleutel in het ticket om de authenticator te decrypteren, vergelijkt de informatie in het ticket met die in de authenticator, het IP-adres waar het verzoek is ontvangen, en de huidige tijd. Als alles overeenkomt, kan het verzoek worden voortgezet.

Er wordt aangenomen dat de klokken binnen enkele minuten gesynchroniseerd zijn. Als de tijd in het verzoek in de toekomst of in het verleden te lang is, behandelt de server het verzoek als een poging om een eerder verzoek opnieuw af te spelen. De server mag alle eerdere verzoeken ook bijhouden met nog geldige tijden. Om de terugspeelaanvallen verder te verijdelen, kan een verzoek dat wordt ontvangen met dezelfde ticket en tijdstempel als een reeds ontvangen verzoek worden verworpen.

Tenslotte, als de client aangeeft dat de server ook zijn identiteit moet bewijzen, voegt de server één toe aan de tijdstempel van de client die in de authenticator wordt verstuurd, versleutelt het resultaat in de sessiesleutel en stuurt het resultaat terug naar de client.

Aan het eind van deze beurs is de server zeker dat, volgens Kerberos, de klant is wie het is. Als wederzijdse authenticatie optreedt, is de klant er ook van overtuigd dat de server authentiek is. Bovendien delen de cliënt en de server een sleutel die niemand anders kent, en kunnen zij er veilig van uitgaan dat een redelijk recent bericht dat in die sleutel is versleuteld, afkomstig is van de andere partij.

Kerberos-serverTickets verkrijgen

Bedenk dat een ticket alleen goed is voor één server. Daarom is het noodzakelijk een afzonderlijk ticket te verkrijgen voor elke dienst die de cliënt wil gebruiken. Kaarten voor individuele servers kunnen worden verkregen bij de ticketverstrekkende dienst. Aangezien de kaartverkoopdienst zelf een dienst is, maakt het gebruik van het in de vorige afdeling beschreven protocol voor de toegang tot de dienst.

Wanneer een programma een ticket vereist dat nog niet is aangevraagd, stuurt het een verzoek naar de server die het ticket toekent. Het verzoek bevat de naam van de server waarvoor een ticket wordt aangevraagd, evenals het ticket waarmee de biljetten worden toegekend en een echtheidscontrole die is gebouwd zoals in de vorige paragraaf is beschreven.

De server die het biljet geeft, controleert vervolgens de authenticator en het ticket dat u geeft, zoals hierboven beschreven. Indien geldig, genereert de server die het ticket toekent een nieuwe willekeurige sessiesleutel die tussen de client en de nieuwe server gebruikt moet worden. Het bouwt dan een ticket voor de nieuwe server met de naam van de client, de servernaam, de huidige tijd, het IP-adres van de client en de nieuwe sessiesleutel die het net gegenereerd heeft. De levensduur van het nieuwe ticket is het minimum van de resterende levensduur voor het kaartje dat de tickets geeft en het standaard voor de dienst.

De server die het ticket toekent stuurt dan het ticket, samen met de sessiesleutel en andere informatie, terug naar de klant. Dit keer is het antwoord echter versleuteld met de sessiesleutel die deel uitmaakte van het ticket dat u een ticket geeft. Op deze manier hoeft de gebruiker het wachtwoord niet opnieuw in te voeren.

De Kerberos-database

Tot dit punt hebben we operaties besproken die alleen-lezen toegang tot de Kerberos-database vereisen. Deze bewerkingen worden uitgevoerd door de echtheidsdienst, die zowel op master- als slaafmachines kan draaien.

In deze sectie bespreken we bewerkingen die schrijven naar de database vereisen. Deze bewerkingen worden uitgevoerd door de administratieve dienst, de Kerberos Database Management Service (KDBM) genoemd. In de huidige implementatie is bepaald dat wijzigingen alleen in de Kerberos-databank mogen worden aangebracht; Slave kopieën zijn alleen-lezen. Daarom mag de KDBM-server alleen op de master Kerberos-machine draaien.

Merk op dat, terwijl authenticatie nog kan voorkomen (op slaven), beheerverzoeken niet kunnen worden bediend als de master machine omlaag is. In onze ervaring heeft dit geen probleem opgeleverd, omdat de verzoeken om administratie zeldzaam zijn.

Het KDBM behandelt verzoeken van gebruikers om hun wachtwoorden te veranderen. De clientkant van dit programma, die verzoeken via het netwerk naar het KDBM stuurt, is het kpasswd-programma. KDBM accepteert ook verzoeken van Kerberos beheerders, die hoofdrolspelers aan de database kunnen toevoegen zowel als wachtwoorden voor bestaande hoofdrolspelers wijzigen. De clientzijde van het beheerprogramma, dat ook verzoeken via het netwerk naar het KDBM stuurt, is het kadmin-programma.

De KDBM-server

De KDBM server accepteert verzoeken om hoofden aan de database toe te voegen of de wachtwoorden voor bestaande principes te wijzigen. Deze dienst is uniek omdat de kaartverkoopdienst er geen tickets voor zal geven. In plaats daarvan moet de authenticatiedienst zelf worden gebruikt (dezelfde dienst die wordt gebruikt om een ticket te bezorgen). Het doel hiervan is om van de gebruiker te eisen dat hij een wachtwoord invoert. Als dit niet het geval was, dan zou een gebruiker, als hij of zij het werkstation niet had bijgewoond, een voorbijganger kunnen oplopen en haar/zijn wachtwoord voor hen kunnen wijzigen, iets wat moet worden voorkomen. Als een beheerder haar/zijn werkstation niet bewaakt heeft, kan een passer ook een wachtwoord in het systeem wijzigen.

Wanneer de KDBM-server een verzoek ontvangt, geeft hij toestemming door de geauthentiseerde hoofdnaam van de aanvrager van de wijziging te vergelijken met de hoofdnaam van het doel van het verzoek. Als zij hetzelfde zijn, is het verzoek toegestaan. Als ze niet het zelfde zijn, raadpleegt de KDBM server een toegangscontrolelijst (opgeslagen in een bestand op het master Kerberos systeem). Als de hoofdnaam van de verzoeker in dit bestand staat, is het verzoek toegestaan, anders wordt hij afgewezen.

Bij conventie verschijnen de namen met een NULL-instantie (de standaardinstantie) niet in het bestand van de toegangscontrolelijst; in plaats daarvan wordt een admin-instantie gebruikt. Om die reden moet een gebruiker om een beheerder van Kerberos te worden een beheerinstantie voor die gebruikersnaam worden gecreëerd en aan de toegangscontrolelijst toegevoegd. Deze conventie stelt een beheerder in staat om een ander wachtwoord te gebruiken voor het beheer van Kerberos en vervolgens gebruikt hij/zij voor normale inloggen.

Alle verzoeken om toegang tot het KDBM-programma, ongeacht of deze zijn toegestaan of geweigerd, worden geregistreerd.

De Kadmin- en Koolwd-programma's

De beheerders van Kerberos gebruiken het kadmin programma om hoofdrolspelers aan de databank toe te voegen, of de wachtwoorden van bestaande hoofdrolspelers te veranderen. Een beheerder moet het wachtwoord voor zijn naam van de beheerder invoeren wanneer hij het kadmin-programma aanvraagt. Dit wachtwoord wordt gebruikt om een ticket naar de KDBM-server op te halen.

De gebruikers kunnen hun wachtwoorden van Kerberos veranderen met behulp van het snelwerkende programma. Ze moeten hun oude wachtwoord invoeren wanneer ze zich op het programma beroepen. Dit wachtwoord wordt gebruikt om een ticket naar de KDBM-server op te halen.

Kerberos-databases replicatie

Elk Kerberos gebied heeft een meestermachine Kerberos, die de hoofdkopie van de authenticatiedatabank huisvest. Het is mogelijk (hoewel niet noodzakelijk) om aanvullende, alleen-lezen exemplaren van de gegevensbank over slavemachines elders in het systeem te hebben. De voordelen van het hebben van meerdere exemplaren van de gegevensbank zijn die gewoonlijk voor replicatie worden genoemd: betere beschikbaarheid en betere prestaties . Als de master machine is ingedrukt, kan de authenticiteit nog steeds worden bereikt op één van de slavenmachines. Het vermogen om echtheidscontrole uit te voeren op een van meerdere machines vermindert de kans op een knelpunt in de hoofdmachine.

Het bewaren van meerdere exemplaren van de gegevensbank leidt tot het probleem van de consistentie van gegevens. Wij hebben vastgesteld dat zeer eenvoudige methoden volstaan om inconsistentie te verhelpen. De database wordt elk uur gedumpt. De gegevensbank wordt in zijn geheel naar de slavenmachines gestuurd, die vervolgens hun eigen databases bijwerken. Een programma op de master host, genaamd kprop, verstuurt de update naar een peer programma, kpropd genaamd, uitgevoerd op elk van de slavenmachines. Eerst stuurt kprop een checksum van de nieuwe database die het gaat verzenden. De checksum is versleuteld in de Kerberos master database-toets, die zowel de master- als de slaafse Kerberos-machines bezitten. De gegevens worden vervolgens via het netwerk overgebracht naar de kaart op de slavenmachine. De server van het slavenpropageren berekent een checksum van de gegevens die het heeft ontvangen, en als het overeenkomt met het checksum dat door de master wordt verstuurd, wordt de nieuwe informatie gebruikt om de database van de slavin bij te werken.

Alle wachtwoorden in de Kerberos-database zijn versleuteld in de master-database-toets. Daarom is de informatie die van master wordt doorgegeven aan slave over het netwerk niet nuttig voor een afluisteraar. Het is echter van essentieel belang dat alleen de informatie van de kapitein door de slaven wordt aanvaard en dat vervalsing van gegevens, dus de checksum, wordt opgespoord.

Kerberos van buitenaf

In deze sectie worden Kerberos vanuit het praktische gezichtspunt beschreven, eerst zoals door de gebruiker gezien, dan vanuit het gezichtspunt van de toepassingsprogrammeur en ten slotte, door de taken van de Kerberos-beheerder.

Toetsing van Kerberos-gebruiker

Als alles goed gaat, zal de gebruiker nauwelijks merken dat Kerberos aanwezig is. Bij onze UNIX-implementatie wordt het ticket naar de Kerberos verkregen als onderdeel van het inlogproces. Het wijzigen van het Kerberos-wachtwoord van een gebruiker maakt deel uit van het ingevoerde programma. En Kerberos-tickets worden automatisch vernietigd wanneer een gebruiker zich uitlogt.

Als de inlogsessie van de gebruiker langer duurt dan de levensduur van het ticket (momenteel 8 uur), zal de gebruiker de aanwezigheid van Kerberos opmerken omdat de volgende keer dat een Kerberos-geauthentiseerde toepassing wordt uitgevoerd, dit mislukt. Het Kerberos-ticket is verlopen. Op dat punt kan de gebruiker het kinit-programma uitvoeren om een nieuw ticket te bemachtigen voor de server. Als je inlogt, moet er een wachtwoord worden opgegeven om het te krijgen. Een gebruiker die de klist uit nieuwsgierigheid uitvoert, kan verbaasd zijn over alle kaartjes die in stilte namens hem/haar zijn verkregen voor diensten waarvoor Kerberos-authenticatie vereist is.

Kerberos uit het gezichtspunt van de programmeur

Een programmeur die een Kerberos-toepassing schrijft, zal vaak authenticatie toevoegen aan een reeds bestaande netwerktoepassing bestaande uit een client- en serverkant. We noemen dit proces 'Kerberizing' een programma. Kerberoering houdt doorgaans in dat een oproep wordt gedaan naar de Kerberos-bibliotheek om verificatie uit te voeren op het eerste verzoek om een onderhoudsdienst. Het kan ook oproepen naar de DES-bibliotheek omvatten om berichten en gegevens te versleutelen die vervolgens tussen applicatie-client en toepassingsserver worden verzonden.

De meest gebruikte bibliotheekfuncties zijn krb_mk_req aan de clientkant en krb_rd_req aan de serverkant. De krb_mk_req routine neemt als parameters de naam, instantie en domein van de doelserver in, die zullen worden aangevraagd, en mogelijk een checksum van de te verzenden gegevens. De client stuurt dan het bericht dat door de krb_mk_req aanroep via het netwerk wordt teruggestuurd naar de serverkant van de toepassing. Wanneer de server dit bericht ontvangt, belt hij de bibliotheek routine krb_rd_req. Deze routine retourneert een vonnis over de authenticiteit van de vermeende identiteit van de afzender .

Als de toepassing vereist dat berichten die tussen client en server verzonden worden, geheim zijn, kan bibliotheekoproepen naar krb_mk_priv (krb_rd_priv) worden gemaakt om berichten in de sessiesleutel te versleutelen die beide partijen nu delen.

De beheerdershandleiding van Kerberos

De taak van de Kerberos-beheerder begint met het uitvoeren van een programma om de database te initialiseren. Een ander programma moet worden uitgevoerd om belangrijke hoofdrolspelers in de database te registreren, zoals de naam van de Kerberos-beheerder met een admininstantie. De Kerberos-verificatieserver en de beheerserver moeten worden opgezet. Als er slavendatabases zijn, moet de beheerder ervoor zorgen dat de programma's om de updates van de database van master naar slaven te verspreiden periodiek worden afgebroken.

Nadat deze eerste stappen zijn gezet, manipuleert de beheerder de database via het netwerk met behulp van het kadmin-programma. Door dat programma kunnen nieuwe principes worden toegevoegd en kunnen wachtwoorden worden gewijzigd.

Met name wanneer een nieuwe Kerberos-toepassing aan het systeem wordt toegevoegd, moet de Kerberos-beheerder een paar stappen ondernemen om het te laten werken. De server moet in de

database worden geregistreerd en toegewezen aan een private key (meestal is dit een automatisch gegenereerde willekeurige toets). Vervolgens moeten bepaalde gegevens (inclusief de sleutel van de server) uit de database worden gehaald en in een bestand op de machine van de server worden geïnstalleerd. Het standaardbestand is /etc/srvtab. De bibliotheekroutine van krb_rd_req die door de server wordt opgeroepen (zie de vorige sectie) gebruikt de informatie in dat bestand om berichten te decrypteren die versleuteld zijn in de privésleutel van de server. Het bestand /etc/srvtab authenticeert de server als een wachtwoord dat getypt is in een terminal en authenticeert de gebruiker.

De beheerder van Kerberos moet ook verzekeren dat de machines van Kerberos fysiek veilig zijn, en zou ook wijs zijn om steunen van de Mastergegevensbank te handhaven.

Het grotere Kerberos-beeld

In deze sectie beschrijven we hoe Kerberos in de omgeving van Athena past, inclusief het gebruik ervan door andere netwerkservices en toepassingen, en hoe het interageert met afgelegen Kerberos-gebieden. Zie G.W. Treese voor een volledigere beschrijving van de omgeving van Athena.

Gebruik van Kerberos door andere netwerkservices

Verschillende netwerktoepassingen zijn aangepast om Kerberos te gebruiken. De opdrachten rlogin en rsh proberen eerst het gebruik van Kerberos te authentiseren. Een gebruiker met geldige Kerberos-tickets kan naar een andere Athena-machine inloggen zonder dat hij .rgastheren-bestanden hoeft in te stellen. Als de Kerberos-authenticatie mislukt, vallen de programma's terug op hun gebruikelijke methoden van toestemming, in dit geval de .rgastheren-bestanden.

We hebben het Post Office Protocol gewijzigd zodat we Kerberos kunnen gebruiken voor het authentiseren van gebruikers die hun e-mail willen ophalen van het "postkantoor". Er is onlangs in Athena een programma voor het leveren van berichten ontwikkeld, Zephyr genaamd, dat ook Kerberos voor authenticatie gebruikt.

Het programma om nieuwe gebruikers aan te sluiten, het genoemde register, gebruikt zowel het Service Management System (sms) als Kerberos. Van SMS bepaalt het of de informatie die door de nieuwe Athena-gebruiker wordt ingevoerd, zoals naam en MIT-identificatienummer, geldig is. Het controleert vervolgens met Kerberos of de gevraagde gebruikersnaam uniek is. Als alles goed gaat, wordt er een nieuwe vermelding gemaakt in de Kerberos-database, die de gebruikersnaam en het wachtwoord bevat.

Voor een gedetailleerde discussie over het gebruik van Kerberos om Sun's Network File System te beveiligen, raadpleeg de appendix.

Interactie met andere Kerberi

Naar verwachting zullen verschillende administratieve organisaties Kerberos willen gebruiken voor gebruikersverificatie. Ook wordt verwacht dat de gebruikers van de ene organisatie in veel gevallen gebruik zullen willen maken van de diensten van de andere. Kerberos ondersteunt meerdere administratieve domeinen. De specificatie van namen in Kerberos omvat een veld dat het rijk wordt genoemd. Dit veld bevat de naam van het administratieve domein waarin de gebruiker gewaarmerkt moet worden.

De diensten worden gewoonlijk in één enkel gebied geregistreerd en zullen slechts geloofsbrieven accepteren die door een authenticatieserver voor dat gebied worden verstrekt. Een gebruiker wordt gewoonlijk in één gebied (het lokale rijk) geregistreerd, maar het is voor haar/hem mogelijk om geloofsbrieven te verkrijgen die door een ander gebied (het verafgelegen gebied) worden afgegeven, op de kracht van de authenticatie die door het lokale domein wordt geboden. Credentials die geldig zijn in een afgelegen gebied geven het veld aan waarin de gebruiker oorspronkelijk geauthentiseerd was. De diensten op het verre gebied kunnen kiezen of om deze geloofsbrieven te honoreren, afhankelijk van de graad van veiligheid vereist en het niveau van vertrouwen in het gebied dat aanvankelijk de gebruiker voor authentiek verklaarde.

Om verificatie over de grenzen te kunnen uitvoeren, is het noodzakelijk dat de beheerders van elk paar gebieden een sleutel selecteren die tussen hun gebieden moet worden gedeeld. Een gebruiker in het lokale domein kan dan een ticket aanvragen dat hem op de lokale authenticatieserver geeft voor de ticketserver in het afstandsgebied. Wanneer dat ticket gebruikt wordt, erkent de server die het toegangsbewijs voor tickets op afstand geeft dat het verzoek niet van zijn eigen domein komt, en gebruikt hij de eerder uitgewisselde sleutel om het ticket te decrypteren. Het geeft dan een ticket uit zoals het normaal zou zijn, behalve dat het veld realm voor de client de naam bevat van het gebied waarin de client oorspronkelijk gewaarmerkt was.

Deze benadering kan worden uitgebreid om zichzelf door een reeks gebieden te authentiseren tot het gebied met de gewenste service wordt bereikt. Om dit te doen, zou het echter nodig zijn om het hele pad dat is genomen op te nemen, en niet alleen de naam van het eerste rijk waarin de gebruiker echt was bevonden. In zo'n situatie is het enige dat bekend is bij de server dat A zegt dat B zegt dat C zegt dat de gebruiker zo-en-zo is. Deze verklaring kan alleen worden vertrouwd als iedereen langs het pad ook vertrouwd is.

Kerberos-problemen en openstaande problemen

Er zijn een aantal problemen en openstaande problemen die verband houden met het Kerberos - authenticatiemechanisme. Eén van de kwesties is hoe je de juiste levensduur van een ticket kunt kiezen, hoe je volmachten kunt toestaan en hoe je de integriteit van een werkstation kunt garanderen.

Het probleem van de kaartverkoop is een kwestie van het kiezen van de juiste ruil tussen veiligheid en gemak. Als het leven van een ticket lang is, dan kan een ticket en de bijbehorende sessiesleutel gestolen of misplaatst worden, voor een langere periode gebruikt worden. Deze informatie kan worden gestolen als een gebruiker vergeet een werkstation te verlaten. Als een gebruiker op een systeem is geauthentiseerd dat meerdere gebruikers toestaat, kan een andere gebruiker met toegang tot wortel de informatie vinden nodig om gestolen kaartjes te gebruiken. Het probleem met het geven van een ticket tijdens een korte levensduur is echter dat wanneer het verlopen is, de gebruiker een nieuw ticket moet bemachtigen waarvoor de gebruiker het wachtwoord opnieuw moet invoeren.

Een open probleem is het proxy-probleem. Hoe kan een geauthentiseerde gebruiker een server toestaan om andere netwerkdiensten in haar/zijn naam te verwerven? Een voorbeeld waar dit belangrijk zou zijn is het gebruik van een dienst die direct van een server toegang tot beschermde bestanden zal krijgen. Een ander voorbeeld van dit probleem is wat we authenticatie-verzending noemen. Als een gebruiker zich in een werkstation heeft aangemeld en zich naar een externe host inlogt, zou het fijn zijn als de gebruiker toegang had tot dezelfde lokaal beschikbare services, terwijl er een programma op de externe host wordt uitgevoerd. Wat dit moeilijk maakt is dat de gebruiker de afstandshost niet kan vertrouwen, dus is het verzenden van de authenticatie in alle gevallen niet wenselijk. Op dit moment hebben wij geen oplossing voor dit probleem.

Een ander probleem, dat belangrijk is in de Athena-omgeving, is hoe de integriteit van de software die op een werkstation actief is kan worden gegarandeerd. Dit is niet zozeer een probleem op particuliere werkstations, aangezien de gebruiker die het zal gebruiken er de controle over heeft. Op openbare werkstations is het echter mogelijk dat er iemand is gekomen die het inlogprogramma heeft aangepast om het wachtwoord van de gebruiker op te slaan. De enige oplossing die momenteel in onze omgeving beschikbaar is, is om het voor mensen moeilijk te maken om software te wijzigen die op de openbare werkstations actief is. Een betere oplossing zou vereisen dat de sleutel van de gebruiker nooit een systeem verlaat waarvan de gebruiker weet dat het kan worden vertrouwd. Eén manier waarop dit zou kunnen worden gedaan, zou zijn als de gebruiker over een smartcard beschikte die in staat was de versleutelingen te doen die in het verificatieprotocol worden vereist.

Kerberos-status

Een prototype van Kerberos werd in september 1986 in productie genomen. Sinds januari 1987 is Kerberos het enige middel van Project Athena om de 5.000 gebruikers, 650 werkstations en 65 servers te authentificeren. Bovendien wordt Kerberos nu gebruikt in plaats van .rgastheren files for control access in verscheidene timesharing systemen van Athena.

Kerberos-erkenning

Kerberos werd oorspronkelijk ontworpen door Steve Miller en Clifford Neuman met suggesties van Jeff Schiller en Jerry Saltzer. Sindsdien zijn er veel meer mensen bij het project betrokken. Onder hen zijn 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, Bill Sommerfeld, Ted T'so En Stan Zanarotti.

We zijn dankbaar voor Dan Geer, Kathy Lieben, Josh Lubarr, Ken Raeburn, Jerry Saltzer, Ed Steiner, Robbert van Renesse en Win Treese, wier suggesties de eerdere ontwerpen van dit document veel hebben verbeterd.

Jedlinsky, J.T. Kohl, en W.E. Sommerfeld, "The Zephyr notification System," in de Conferentie van Usenix (Winter, 1988).

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

R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh, en B. Lyon, "Design and Implementation of the Sun Network Filessystem" in Usenix Conference Proceedings (zomer, 1985).

Bijlage: Kerberos-toepassing op het VN-netwerkbestandssysteem (NFS)

Een belangrijk onderdeel van het werkstationsysteem van Project Athena is de koppeling van het netwerk tussen het werkstation van de gebruiker en de particuliere opslag van bestanden (home folder). Alle particuliere opslag bevindt zich op een verzameling computers (momenteel VAX 11/750) die aan dit doel zijn toegewezen. Op die manier kunnen we diensten aanbieden op openbare UNIX-werkstations. Wanneer een gebruiker zich inlogt bij een van deze publiekelijk beschikbare werkstations, dan valideren we haar/zijn naam en wachtwoord tegen een lokaal

gevestigd wachtwoordbestand, dan gebruiken we Kerberos om de authenticiteit van het bestand te bepalen. Het inlogprogramma wordt gevraagd om een gebruikersnaam (voor elk UNIX-systeem). Deze gebruikersnaam wordt gebruikt om een Kerberos-ticket te bemachtigen. Het inlogprogramma gebruikt het wachtwoord om een DES-toets te genereren voor het decrypteren van het ticket. Als decryptie succesvol is, wordt de huisfolder van de gebruiker geplaatst door de Hesiod naming service te raadplegen en door NFS gemonteerd. Het logprogramma verandert vervolgens de controle over het shell van de gebruiker, dat dan de traditionele aanpassingsbestanden per gebruiker kan uitvoeren omdat de home folder nu "bevestigd" aan het werkstation is. De Hesiod-service wordt ook gebruikt om een onderdeel in het lokale wachtwoordbestand te maken. (Dit is in het belang van programma's die informatie in /etc/passwd opzoeken.)

Van verschillende opties voor de levering van de afstandsbediening, kozen we voor Sun's Network File System. Dit systeem beantwoordt echter niet aan onze behoeften. NFS veronderstelt dat alle werkstations in twee categorieën vallen (zoals bekeken vanuit het standpunt van een bestandsserver): vertrouwd en onbetrouwbaar. Onvertrouwde systemen hebben helemaal geen toegang tot bestanden, vertrouwde kan dat niet. Trusted systemen zijn volledig betrouwbaar. Er wordt aangenomen dat een betrouwbaar systeem wordt beheerd door een vriendelijk beheer. Meer in het bijzonder is het mogelijk van een betrouwbaar werkstation om zich voor te stellen als elke geldige gebruiker van het bestandsservicesysteem en zo toegang te krijgen tot ongeveer elk bestand op het systeem. (Alleen bestanden die eigendom zijn van "root" zijn vrijgesteld.)

In onze omgeving is het beheer van een werkstation (in de traditionele zin van UNIX-systeembeheer) in handen van de gebruiker die het momenteel gebruikt. We maken geen geheim van het basiswachtwoord op onze werkstations, omdat we beseffen dat een werkelijk onvriendelijke gebruiker kan inbreken door het feit dat hij op dezelfde fysieke locatie zit als de machine en toegang heeft tot alle console-functies. Daarom kunnen we onze werkstations niet echt vertrouwen in de NFS-interpretatie van vertrouwen. Om goede toegangscontroles in onze omgeving mogelijk te maken, moesten we een aantal wijzigingen aanbrengen in de basis-NFS-software en Kerberos integreren in het schema.

Kerberos ongewijzigde NFS

Bij de implementatie van NFS waarmee we begonnen zijn (van de Universiteit van Wisconsin) werd authenticatie aangeboden in de vorm van een stuk gegevens opgenomen in elk NFS - verzoek (een genaamd "creditential" in de NFS terminologie). Dit krediet bevat informatie over de unieke gebruikersidentificatie (UID) van de aanvrager en een lijst van de groepsidentificatoren (GID's) van het lidmaatschap van de aanvrager. Deze informatie wordt vervolgens gebruikt door de NFS server voor toegangscontrole. Het verschil tussen een vertrouwd en een onbetrouwbaar werkstation is of de referenties ervan al dan niet zijn geaccepteerd door de NFS server.

Aangepaste Kerberos NFS

In onze omgeving moeten NFS-servers aanmeldingsgegevens van een werkstation accepteren als en alleen als de aanmeldingsgegevens de UID van de gebruiker van het werkstation aangeven, en niet meer.

Een voor de hand liggende oplossing zou zijn de aard van de geloofsbrieven te veranderen van louter indicatoren van UID en GID's in volledig geblazen Kerberos-gewaarmerkte gegevens. Als deze oplossing wordt aangenomen, zou er echter een belangrijke prestatiereslag worden betaald. Credentials worden uitgewisseld op elke NFS - transactie, met inbegrip van alle Schijf lezen en

schrijven activiteiten. Het opnemen van een Kerberos-authenticatie op elke disktransactie zou een redelijk aantal volledig opgemerkte encrypties (gedaan in software) per transactie toevoegen en zou, volgens onze envelopberekeningen, onaanvaardbare prestaties hebben geleverd. (Het zou ook nodig hebben om de bibliotheekroutines van Kerberos in de kerneladresruimte te plaatsen.)

We hadden een hybride aanpak nodig, zoals hieronder beschreven. Het basisidee is om de NFS server map aanmeldingsgegevens te laten ontvangen van de werkstations van de client, naar een geldig (en mogelijk verschillend) gecrediteerd op het serversysteem. Deze mapping wordt uitgevoerd in het kanaal van de server op elke NFS-transactie en wordt ingesteld op "berg"-tijd door een proces op gebruikersniveau dat Kerberos gematigde verificatie uitvoert voordat een geldige Pernelen wordt toegewezen voor het in kaart brengen.

Om dit te implementeren voegden we een nieuwe systeemaanroep aan het netwerk toe (alleen vereist op serversystemen, niet op clientsystemen) dat voorziet in de controle van de mapping-functie die inzendingen van inzendingen van client-werkstations instelt op aanmeldingsgegevens die geldig zijn voor gebruik op de server (indien aanwezig). De basiskaartfunctie geeft de teple in kaart:

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

naar een geldig NFS-gecrediteerd op het serversysteem. Het CLIENT-IP-ADRES wordt afgeleid uit het NFS-aanvraagpakket dat door het clientsysteem is geleverd. Opmerking: alle informatie in de door de klant gegenereerde gecrediteerde creditzijde, met uitzondering van de UID-ON-CLIENT, wordt weggegooid.

Als er geen mapping bestaat, reageert de server op een van twee manieren, afhankelijk van de configuratie. In onze vriendschappelijke configuratie blijven de oninstelbare verzoeken in de aanmeldingsgegevens van de gebruiker 'niemand' die geen bevoorrechte toegang heeft en een unieke UID heeft, standaard. Onvriendelijke servers geven een NFS-toegangsfout terug als er geen geldige mapping kan worden gevonden voor een inkomende NFS-gecrediteerd.

Onze nieuwe systeemaanroep wordt gebruikt om items toe te voegen en te verwijderen van de kaart die de centrale inwoner heeft. Het biedt ook de mogelijkheid om alle items die zich op een bepaalde UID in het serversysteem indelen te spoelen of alle items van een bepaalde CLIENT-IP-ADRES te spoelen.

We wijzigden de steundatum (die NFS-verzoeken om bedragen op serversystemen behandelt) om een nieuw transactietype te accepteren, het verzoek om Kerberos-authenticatie in kaart te brengen. In het kader van het montageproces biedt het clientsysteem in feite een Kerberos-authenticator samen met een indicatie van haar/zijn UID-ON-CLIENT (versleuteld in de Kerberos-authenticator) op het werkstation. De steundatum van de server converteert de belangrijkste naam van Kerberos naar een lokale gebruikersnaam. Deze gebruikersnaam wordt vervolgens in een speciaal bestand opgezocht om de lijst met UID's en GID's van de gebruiker op te geven. Voor efficiëntie is dit bestand een ndbm-gegevensbestand met de gebruikersnaam als de toets. Op basis van deze informatie wordt een NFS-gecrediteerd en aan het personeel overgedragen als geldige mapping van het <CLIENT-IP-ADRES, CLIENT-UID>-vakje voor dit verzoek.

Bij het monteren wordt een verzoek naar de steundatum verzonden om de eerder toegevoegde afbeelding uit de kern te verwijderen. Het is ook mogelijk om op logout-tijd een verzoek te verzenden om alle mapping voor de huidige gebruiker op de server in kwestie ongeldig te maken, zodat alle resterende afbeeldingen die er nog zijn (hoewel ze dat niet zouden moeten) worden opgeruimd voordat het werkstation voor de volgende gebruiker beschikbaar wordt gesteld.

Kerberos-beveiligingsimplicaties van de gewijzigde NFS

Deze implementatie is niet helemaal veilig. Om te beginnen, worden de gebruikersgegevens nog over het netwerk verzonden in een niet gecodeerd, en daarom ontvankelijk, formulier. De lage, per-transactie authenticatie is gebaseerd op een <CLIENT-IP-ADRES, CLIENT-UID> paar dat niet versleuteld is in het aanvraagpakket. Deze informatie zou kunnen worden gesmeed en aldus de veiligheid in gevaar gebracht. Houd er echter rekening mee dat alleen wanneer een gebruiker actief gebruik maakt van zijn/haar bestanden (dat wil zeggen, terwijl hij aangemeld is), er geldige afbeeldingen zijn en dat deze vorm van aanval daarom beperkt is tot wanneer de gebruiker in kwestie inlogt. Wanneer een gebruiker niet ingelogd is, zal geen hoeveelheid IP-adresvervalsing onbevoegde toegang tot haar/zijn bestanden toestaan.

Kerberos-referenties

1. S.P. Miller, B.C. Neuman, J.I. Schiller en J.H. Saltzer, Sectie E.2.1: Kerberos Authentication and Authorization System, M.I.T. Project Athena, Cambridge, Massachusetts (21 december 1987).
2. E. Balkovich, S.R. Lerman, en R.P. Parmelee, "Computing in het hoger onderwijs: The Athena Experience," Communications of the ACM, 28(11), blz. 1214-1224, ACM (november 1985).
3. R.M. Needham en M.D. Schroeder, "Use Encryption for Authentication in Large Networks of Computers," Communications of the ACM, Vol. 21(12), blz. 993-999 (december, 1978).
4. V.L. Voydock en S.T. Kent, "Security Mechanisms in High-Level Network Protocols," Computing Surveys, V.L. 15(2), ACM (juni 1983).
5. National Bureau of Standards, "Data Encryption Standard", Federal Information Processing Standards Publication 46, Regeringsdrukkerij, Washington, DC (1977).
6. SP Dyer, "Hesiod," in Usenix Conference Proceedings (Winter, 1988).
7. W.J. Bryant, tutorial van de Kerberos-programmeur, MIT Project Athena (in voorbereiding).
8. W.J. Bryant, Kerberos Administrator's Manual, MIT Project Athena (in voorbereiding).
9. G.W. Treese, "Berkeley Unix op 1000 werkstations: Athena Wijzigingen in 4.3BSD," in de Conferentie van Usenix (Winter, 1988).
10. C.A. DellaFera, M.W. Eichin, R.S. French, D.C. Jedlinsky, J.T. Kohl, en W.E. Sommerfeld, "The Zephyr notification System," in de Conferentie van Usenix (Winter, 1988).
11. M.A. Rosenstein, D.E. Geer, en P.J. Levine, in Usenix Conference Proceedings (Winter, 1988).
12. R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh, en B. Lyon, "Design and Implementation of the Sun Network Filessystem" in Usenix Conference Proceedings (zomer, 1985).

Gerelateerde informatie

- Categoriepagina voor Kerberos-ondersteuning
- Technische ondersteuning en documentatie – Cisco Systems