



CRESCO

**Centro computazionale di RicErca sui Sistemi
COMplessi**

**Dipartimento di Ingegneria dell'Informazione e Matematica
Applicata**

Università di Salerno

Responsabile Unità Operativa

Prof. Ciro D'Apice

LA1-SP I.2 - Sviluppo ed integrazione dell'architettura GRID e di grafica 3 D

Attività I.2.1 Integrazione ENEA GRID con altre architetture di GRID computing

1d – Integrazione con ENEA-GRID dei prodotti sviluppati, con il supporto dei ricercatori ENEA

Test delle funzioni prototipali in ambiente Windows gettok e puttok del kit
LSF per AFS

V. 1.0

Indice

1. Varie	3
1.1. Bibliografia e Riferimenti	3
1.2 Acronimi.....	<i>Errore. Il segnalibro non è definito.</i>
1.2 Glossario	<i>Errore. Il segnalibro non è definito.</i>
2. Contesto, obiettivi e scopo	7
3. Autenticazione nel mixed cluster di LSF	9
4. Mixed cluster LSF in ENEA-GRID.....	12
5. Conclusioni	13

1. Varie

1.1. Bibliografia e Riferimenti

- [1] Analisi dell'infrastruttura GRID ENEA ed individuazione dei punti critici ai fini dell'interoperabilità e dell'estensione delle caratteristiche di gestione e controllo dell'infrastruttura stessa
- [2] Studio di fattibilità per il porting in ambiente Windows delle routine gettok e puttok del kit LSF per AFS
- [3] Realizzazione delle funzioni prototipali in ambiente Windows gettok e puttok del kit LSF per AFS
- [4] AFS User Guide di IBM
- [5] Open AFS Guide
- [6] Programming with the Andrew Cache Manager
- [7] Administering Platform LSF
- [8] <http://www.cresco.enea.it>
- [9] <http://www.platform.com>
- [10] <http://www.openafs.org>

1.2. Acronimi

ACL	A ccess C ontrol L ist
AFS	A ndrew F ile S ystem
API	A pplication P rogram I nterface
CRESCO	C entro C omputazionale di Ric erca sui Sistemi Complessi
IOCTL	I nput O utput C on T ro L
KDC	K ey D istribution C enter
KTC	K ernel T oken C ache
KVNO	K ey V ersion N umber
LSF	L oad S haring F acility
PAG	P rocess A uthentication G roup

RPC	R emote P rocedure C all
TGS	T icket G ranting S ervice
TGT	T icket G ranting T icket
UID	U ser I dentifier

1.3. Glossario

Authentication Service (AS)

E' la parte di KDC che distribuisce i ticket sia i TGT sia altri tipi di ticket per l'accesso ai servizi.

Cache Manager

In AFS è il processo che gira sui sistemi client, trasferisce i dati da e verso il server, gestisce l'accesso, la visualizzazione, e la modifica dei file dalla directory principale /afs che rappresenta la radice di una struttura organizzata ad albero.

Clear Token

Parte del token di AFS che contiene al suo interno, fra l'altro, la Session Key (anche detta Handshake Key). Dai codici sorgenti esaminati la lunghezza del Clear Token dovrebbe essere di 24 byte.

Complex Mutual Authentication

Particolare tipo di Mutual Authentication che coinvolge tre chiavi di cifratura e tre parti. E' il tipo di autenticazione adottato da AFS

Handshake Key

Vedere Session Key.

Kerberos Authentication Server (kaserver)

E' uno dei processi server di AFS ed assicura che le comunicazioni sulla rete siano sicure. Verifica l'identità dell'utente al login e fornisce le strutture con le quali i partecipanti nelle transazioni verificano la loro identità uno rispetto all'altro. Gestisce l'Authentication DB.

KDC (Key Distribution Center)

E' il servizio che implementa l'autenticazione Kerberos mediante l'Authentication Service (AS) ed il Ticket Granting Service (TGS). Il KDC ha una copia di ogni encryption key associata a ciascun principal. La maggior parte delle

implementazioni di KDC archivia i principals in un database e quindi a volte si impiega il termine Kerberos database applicato al KDC.

Mutual Authentication

Meccanismo con il quale le parti provano la loro effettiva identità e basato sul fatto che le parti provano la loro identità mostrando la conoscenza di uno shared secret.

Principal

E' il nome con cui si fa riferimento alle *entry* nel database dell'Authentication Server. Ad ogni utente, host o servizio di un dato realm viene associato un principal. Il generico formato di un principal utente è : Nome[/Istanza]@REALM mentre per un servizio è Servizio/Hostname@REALM

Realm

(letteralmente regno) indica un dominio amministrativo di autenticazione che fissa i confini entro cui un server di autenticazione è autoritario nell'autenticare un utente, un host o un servizio.

Secret Key

Una chiave di cifratura permanente condivisa da un principal e dal KDC, impiegata per cifrare/decifrare la Session Key inclusa nel TGT dell'autenticazione iniziale. Nel caso il principal sia un utente la Secret Key è derivata dalla password Kerberos dell'utente stesso.

Secret Token

Parte del token riferita anche con il termine ticket dall'analisi del codice e dalle verifiche effettuate può avere diverse lunghezze (40, 56, 150, 166 byte).

Session Key

Una chiave di cifratura temporanea impiegata fra due principal, con un tempo di vita limitato alla durata di un relativo ticket. Nei codici sorgenti esaminati viene anche denominata Handshake Key.

Shared Secret

Informazione conosciuta solo dalle parti che si stanno mutuamente autenticando; esse potrebbero venirne a conoscenza tramite una terza parte considerata fidata dalle prime due. Dovrebbe essere costituita dalla Session Key.

TGS (Ticket-Granting Service)

La parte del KDC che distribuisce i ticket ai client per gli specifici servizi. Il processo utente comunica con il TGS grazie ad un TGT.

TGT (Ticket-Granting Ticket)

Uno speciale ticket Kerberos che consente al client di ottenere, in maniera trasparente altri ticket.

Ticket

Un insieme di credenziali elettroniche che verifica l'identità di un client per la fruizione di uno specifico servizio.

Token

Struttura impiegata da AFS per l'autorizzazione e l'accesso al file system. Al suo interno contiene un ticket. Nei codici esaminati è suddivisa in due sezioni fondamentali il Secret Token (anche riferito con il termine Ticket) ed il Clear Token.

Vice

In ambiente AFS indica l'insieme dei server ed il termine deriva dal nome del software che gira su di essi. Il termine è anche usato come suffisso per l'identificazione, nei codici sorgenti, di strutture e dati (ViceIOCTL, ViceID).

2. Contesto, obiettivi e scopo

Il Progetto CRESCO: Centro computazionale di RicErca sui Sistemi Complessi ha come obiettivo la realizzazione, presso il Centro Ricerche ENEA di Portici (NA), di un importante Polo di calcolo multidisciplinare per lo studio dei sistemi complessi di natura biologica e tecnologica, collegato con *facilities* di elaborazione, archiviazione e visualizzazione 3D di dati ubicate presso i Centri ENEA di Brindisi e Trisaia (MT).

Sul piano delle applicazioni e dei contenuti scientifici, il Progetto si focalizza sulle seguenti linee d'attività:

- Linea di Attività 1 (LA1): Realizzazione del Polo di Calcolo e sviluppo di nuove funzionalità di GRID Computing
l'implementazione di soluzioni innovative in tema di architetture di sistemi di calcolo e di GRID computing per le attività di R&S di punta dell'ENEA che richiedano l'utilizzo di risorse computazionali estremamente importanti. Sul versante infrastrutturale il Progetto prevede la realizzazione di una piattaforma di calcolo di assoluta eccellenza in grado di erogare una potenza dell'ordine del Teraflop, corredata di una nuova classe di funzionalità GRID innovative;
- Linea di Attività 2 (LA2): Sviluppo di tecnologie e modelli computazionali per la descrizione di sistemi complessi di origine biologica e di materiali innovativi
lo studio di oggetti biologici dal punto di vista "sistemico" (in silico cell) e lo studio di sistemi naturali (comunità animali e sociali) secondo il paradigma dei sistemi complessi;
- Linea di Attività 3 (LA3): Sviluppo di modelli di simulazione ed analisi delle Reti tecnologiche complesse e delle loro interdipendenze
lo studio di sistemi tecnologici complessi e delle loro mutue interazioni, e la realizzazione di opportuni strumenti per il modelling, la simulazione e il controllo.

Nell'ambito delle attività previste dalla Linea di Attività 1 il sottoprogetto identificato come SP I.2, *Sviluppo ed integrazione dell'architettura GRID e di grafica 3D*, prevede, fra l'altro, l'attività di integrazione dell'ENEA-GRID con altre architetture di *GRID computing* e nell'ambito di tale attività, nella quale il Dipartimento di Ingegneria dell'Informazione e Matematica Applicata dell'Università degli Studi di Salerno è coinvolto, è prevista, primariamente, un'analisi dell'infrastruttura ENEA-GRID [1] con l'individuazione dei punti critici ai fini dell'interoperabilità, in particolar modo con sistemi Windows, e dell'estensione delle caratteristiche di gestione e controllo dell'infrastruttura stessa.

Uno dei problemi emersi dal documento di analisi è quello relativo al non supportato trasferimento del token da parte del gestore delle risorse LSF dal mondo Unix a quello

Windows. In effetti LSF non supporta AFS in ambiente Windows e quindi, nel momento in cui si desidera attivare un job su una macchina Windows, il mancato trasferimento del token AFS (che ovviamente consente all'utente di poter accedere al proprio volume) determina di fatto l'impossibilità di avere il proprio spazio a disposizione sull'host Windows e, nel caso il sistema di autorizzazione risultasse basato sulle ACL di AFS, l'operazione sarebbe di fatto impossibile. È dunque opportuno verificare se sia possibile effettuare l'operazione di trasferimento del token da e verso sistemi Windows, partendo dai codici sorgenti delle routine a ciò destinate e disponibili per il solo mondo Unix.

Lo studio di fattibilità per l'operazione del *porting* in ambiente Windows [2] del trasferimento del token AFS per LSF si è mosso lungo due direzioni:

1. studio ed analisi del kit che serve a LSF per trattare i token AFS denominato *Platform LSF Source Code Package for AFS*, disponibile per il solo mondo Unix, e nel seguito riferito semplicemente come kit LSF;
2. studio ed analisi dei sorgenti di OpenAFS per Windows per stabilire dei punti in comune fra i due ambienti che potrebbero essere utili per l'operazione di *porting*.

Le due routine che sembrano avere un ruolo chiave nel trasferimento del token da un sistema ad un altro sono le routine **gettok** e **puttok**, presenti nel kit LSF per AFS; per cui si è provveduto alla realizzazione di versioni prototipali di routine idonee alla lettura ed alla scrittura di un token AFS su sistemi Windows ed i risultati conseguiti sono riportati in [3].

Il documento presente è invece relativo all'attività di test, di cui si è manifestata la necessità nelle conclusioni di [3], dei componenti prototipali sviluppati, sui sistemi Windows appartenenti alla GRID ENEA.

Tali attività sono state svolte, così come previsto dal piano progettuale interagendo strettamente con i ricercatori ed i tecnici dell'ENEA.

3. Autenticazione nel mixed cluster di LSF

Nella terminologia LSF con il termine *mixed cluster* si intende un cluster LSF nel quale siano presenti sia sistemi UNIX che sistemi Windows.

Come abbiamo già avuto modo di osservare in [1] [2] e [3] l'obiettivo conclusivo è quello di indagare le varie possibilità per integrare i sistemi Windows nella GRID ENEA ed in tale ottica risulta conveniente configurare nel cluster LSF anche i sistemi Windows che devono far parte della GRID ENEA

Per consentire delle operazioni *cross-platform* è necessario creare delle relazioni fra gli account del mondo Unix e quelli del mondo Windows. In effetti i meccanismi di LSF consentono di specificare una corrispondenza degli account Unix con quelli di un Dominio di Windows.

Specificando nella configurazione di LSF un *LSF user domain* si abilita un *user mapping* predefinito che rende possibile indicare l'utente Windows semplicemente specificandone il nome.

Quindi se l'account Windows appartenente all' *LSF user domain* ha lo stesso *username* dell'account UNIX allora l'*user mapping* predefinito farà sì che LSF schedi e tracci i job dei due account come se tali job appartengano allo stesso utente. In altre parole sull'*execution host* LSF eseguirà i job impiegando l'account appropriato per quell'host, ossia l'account del dominio Windows sul sistema Windows e l'account Unix sul sistema Unix.

In effetti è possibile specificare un numero illimitato di domini Windows come *LSF user domain* ed ovviamente, in tal caso, account con lo stesso username ma appartenenti a differenti domini saranno trattati da LSF come utenti differenti.

Un altro parametro che può essere specificato insieme all'*LSF user domain* è *LSF execute domain* per specificare il dominio che deve essere impiegato in fase di esecuzione senza effettuare i vari tentativi che verrebbero fatti nell'ordine in cui i domini sono riportati nel parametro di cui sopra.

E' possibile anche eseguire i job come un utente locale Windows; in tal caso tutti gli utenti aventi stessa *username* e stessa password, sui vari nodi Windows del cluster, saranno trattati da LSF come se fossero un solo utente e sull'*execution host*, LSF automaticamente eseguirà il job impiegando l'account locale appropriato.

Nel caso gli account presenti abbiano differenti username nei differenti ambienti risulta necessario configurare l'opportuna corrispondenza per i singoli account ma in tale fase non ci soffermeremo su tale eventualità in quanto, nel caso specifico di ENEA-GRID è necessario porre attenzione primariamente agli aspetti connessi con i domini Windows.

A titolo esemplificativo si consideri di avere un account (*user1*) valido sia in 3 domini Windows che in ambiente Unix. In funzione del tipo di cluster installato e configurato LSF riconoscerà solo parte dei 4 account elencati:

Cluster esclusivamente UNIX: in tal caso non è attivo nessun *mapping* fra i vari utenti e quindi LSF riconoscerà esclusivamente un utente, ossia *user1* di Unix;

Cluster esclusivamente Windows: in tal caso non risulterà attiva nessuna corrispondenza fra i tre domini per cui LSF riconoscerà tre differenti utenti, *DomainA\user1*, *DomainB\user1*, *DomainC\user1*;

Mixed Cluster UNIX-Windows: in tal caso bisogna abilitare l'*user mapping* predefinito per uno degli account Windows (ad esempio quello del dominio A) in modo che sia consentita l'esecuzione di job *cross-platform* fra UNIX e Windows. Specificando come *LSF user domain* il DomainA, LSF riconoscerà tre differenti utenti:

- *user1* (l'account UNIX e quello del dominio A saranno riconosciuti come un unico utente LSF)
- *DomainB\user1*
- *DomainC\user1*

Nell'eventualità non si desideri eseguire dei job *cross-platform*, si può disabilitare l'*user mapping* predefinito semplicemente non specificando un *LSF user domain*. In tal caso LSF riconoscerà 4 differenti utenti

- *user1* (l'account UNIX)
- *DomainA\user1*
- *DomainB\user1*
- *DomainC\user1*

Sarà anche possibile specificare più domini windows come *LSF user domain* in modo da poter sottomettere dei job da un sistema UNIX verso un ambiente Windows caratterizzato da più domini.

In uno scenario del genere, caratterizzato dalla presenza di più domini, poiché ci potrebbero essere *username* uguali nei diversi domini, LSF impiegherà la coppia *nomedominio nomeutente* per individuare univocamente l'utente.

La specifica del solo *username* individuerà l'utente appartenente al dominio puntato da *LSF user domain*. Gli utenti degli altri domini saranno riferiti specificando anche il nome del dominio mediante la sintassi *nomedominio\nomeutente*.

L'analisi fin qui svolta si è focalizzata su come LSF stabilisca quale utente impiegare per l'esecuzione di un job sull'*execution host* tralasciando il tema della validazione dell'utente. In effetti LSF riconosce gli ambienti di autenticazione sia di UNIX che di

Windows includendo sia la validazione di dominio che quella di workgroup. Ovviamente nel caso di dominio gli account sono validati a livello centrale e l'account stesso sarà valido su tutti i sistemi appartenenti al dominio mentre in ambienti workgroup l'autenticazione è svolta dal singolo sistema e l'account è valido sul sistema specifico.

Ciò premesso gli utenti che desiderano eseguire dei job in ambiente Windows, impiegando il proprio account, devono registrare il proprio *username* e la propria password in LSF utilizzando il comando **lspasswd**.

Una delle cause che potrebbero determinare la mancata esecuzione di un job in ambiente Windows è dunque costituita dal fatto che LSF non riconosce la password e ciò sia perché l'account utente non è mai stato registrato in LSF mediante il comando **lspasswd**, sia perché la password è stata variata in Windows ma non è mai stata aggiornata in LSF.

4. Mixed cluster LSF in ENEA-GRID

Per poter eseguire il test delle componenti software per il passaggio del token AFS di cui si è ampiamente discusso in [1], [2] e [3] è necessario che nella GRID ENEA si realizzi un mixed-cluster LSF.

In effetti come trattato specificatamente in [2] l'impiego congiunto dei metodi LSF `esub` ed `eexec` consente di passare i dati all'ambiente di esecuzione. In effetti, se `esub` deve passare dei dati a `eexec`, può scrivere tali dati sul suo standard output e gli stessi verranno letti sullo standard input di `eexec` in quanto LSF agisce da *pipe* fra `esub` ed `eexec`. (`esub | eexec`). Quindi lo `stdout` di ogni metodo `esub` è automaticamente inviato a `eexec`. Poiché `eexec` non può trattare più di un *output stream* sarà possibile usare un solo `esub` e quindi un solo standard output per generare i dati sullo standard input di `eexec`. Tale meccanismo è alla base dell'inoltro del token per AFS. In tal caso avremo un metodo `esub` specifico per AFS (ad esempio `esub.afs`) che invia il suo token di autenticazione tramite lo standard output a `eexec` il quale provvederà ad attivarlo sull'host di sottomissione.

Tale meccanismo dunque sembra prescindere dalla fase di autenticazione realizzata da LSF ed esaminata nel paragrafo 3 *Autenticazione nel mixed cluster di LSF* ed esserle successivo.

Quindi per poter procedere con il test delle componenti software sviluppate nella GRID-ENEA risulta indispensabile inserire ed integrare dei sistemi Windows nel cluster LSF.

Durante tale attività peraltro, il personale ENEA ha riscontrato una serie di problematiche per la cui risoluzione sono stati attivati contatti sia con la società che in Italia supporta LSF, ossia la NICE, sia con la casa produttrice, la Platform.

Dai risultati finora disponibili i problemi sarebbero dovuti alla particolare struttura del Dominio Windows realizzato che, per essere integrato nella GRID ENEA, fa riferimento ad un server Kerberos 5. Tale situazione determinerebbe delle anomalie di funzionamento nella fase di autenticazione di LSF.

5. Conclusioni

Le attività di test delle componenti software sviluppate potranno essere attivate solo dopo la risoluzione dei problemi che al momento si riscontrano nella fase di autenticazione di LSF.

Nel mixed-cluster LSF dell'ENEA-GRID nel momento in cui si cercano di eseguire dei job *cross-platform*, la fase di autenticazione prevista da LSF per stabilire, fra l'altro, sotto quale utente dovrà eseguire l'attività sull'*execution host*, esibisce dei malfunzionamenti che al momento sembrano da addebitarsi alla particolare struttura del Dominio Windows che vede la partecipazione di un server Kerberos 5 esterno.

Risolti tali problemi si potrà procedere con l'articolato piano di test e più in generale con le attività descritte nel paragrafo 5 di [3].