Server in configurazione High Availability

L'ESPERIENZA AD ARCETRI

Francesco Tribioli

ICT-INAF, Cefalù – 6 Ottobre 2015

Scopi del progetto

- Eliminare le interruzioni occasionali del servizio (es. per il riavvio di un server)
- Eliminare le interruzioni del servizio o almeno ridurre al minimo i tempi di ripristino in presenza di guasti gravi a uno dei server
- Garantire l'accessibilità ai dati degli utenti e la loro sicurezza anche in caso di fermo di un server
- Migliorare l'efficienza complessiva e modernizzare l'infrastruttura anche attraverso la migrazione da **Sendmail** a **Postfix**, da **NIS** a **LDAP**, da **Solaris** a **Linux**

Hardware

Prima

- Due server gemelli SUN SPARC Enterprise T-2000 (UltraSPARC T1 @ 1.2GHz - 4 core - 16GB RAM DDR2 ECC @ 200MHz)
- 4 x Ethernet 1Gbit/s
- 2 dischi da 70GB SAS 15k rpm per il sistema
- 2 dischi da 140GB SAS 15k rpm per i dati utente

Hardware

Dopo

- Due server gemelli SuperMicro (dual Xeon E5-2630 v3 @ 2.40GHz - 16 core - 64GB RAM DDR4 @ 2133MHz)
- 2 x Ethernet 10Gbit/s
- 1 disco SSD da 500GB per il sistema + 1 disco da 1TB SATA III per il backup
- 2 dischi da 2TB SATA III per i dati utente

Software

Prima

Solaris 10 su entrambi i server

Un server era dedicato alla posta, con **Sendmail** (formato *mbox*), **Dovecot** e **Mailman**, l'altro era dedicato al sito web, con **Apache** e **MySQL**

Un server **NIS** era attivo su entrambi i computer, configurati come master/slave

Le configurazioni e il software addizionale e locale erano aggiornati con rsync, per permettere lo switch manuale ma rapido dei servizi da un server all'altro (alla prova dei fatti non ha funzionato bene)

Software

Dopo

Linux CentOS 7 su entrambi i server

Su entrambi i server sono attivi i servizi di interesse: **OpenLdap**, **Apache**, **MariaDB** (alias MySQL), **Postfix** (formato *Maildir*), **Dovecot**, **Samba**, **Bind**, **SNTP**

Alcuni servizi secondari, come ad esempio **Nagios** e **Mailman**, possono essere attivi su un solo server ma sono configurati e pronti a partire sull'altro server con gli stessi dati

Le configurazioni dei servizi sono identiche e su entrambi i server, giornalmente, il disco di backup viene sincronizzato con il disco di sistema. In caso di guasto del disco SSD è possibile fare il boot dal disco di backup

Problema n. 1

Configurazione delle interfacce di rete

Quando uno dei server non è raggiungibile l'altro server deve subentrare nel giro di qualche secondo

La soluzione più semplice per il nostro caso, con due server e due interfacce fisiche ciascuno, sembrerebbe quella di attivare o disattivare una delle due assegnando su entrambi i server gli stessi numeri IP

È possibile far qualcosa di molto simile per mezzo di keepalived, un software disponibile con la distribuzione standard di CentOS che implementa il protocollo **VRRP** (Virtual Router Redundancy Protocol)

Keepalived (1)

Il protocollo **VRRP** prevede di definire delle «istanze». Per ciascun server assegnato a una istanza si definisce l'interfaccia da usare e una priorità. Ognuno può essere nello stato di **MASTER**, con i numeri IP dell'istanza assegnati all'interfaccia fisica specificata (IP aliasing se più di uno), oppure nello stato di **BACKUP**, con l'interfaccia configurata e attiva ma senza i numeri IP assegnati. I server possono comunicare via multicast oppure si possono specificare i numeri IP dei server nella configurazione di ciascuno di essi

Il processo *keepalived* su ciascun server assegnato all'istanza informa a intervalli regolari gli altri server del proprio stato e priorità e di volta in volta il server che ha la priorità più alta assume lo stato di master. L'intervallo di handshaking si può definire nella configurazione. Nel nostro caso abbiamo scelto 5 secondi

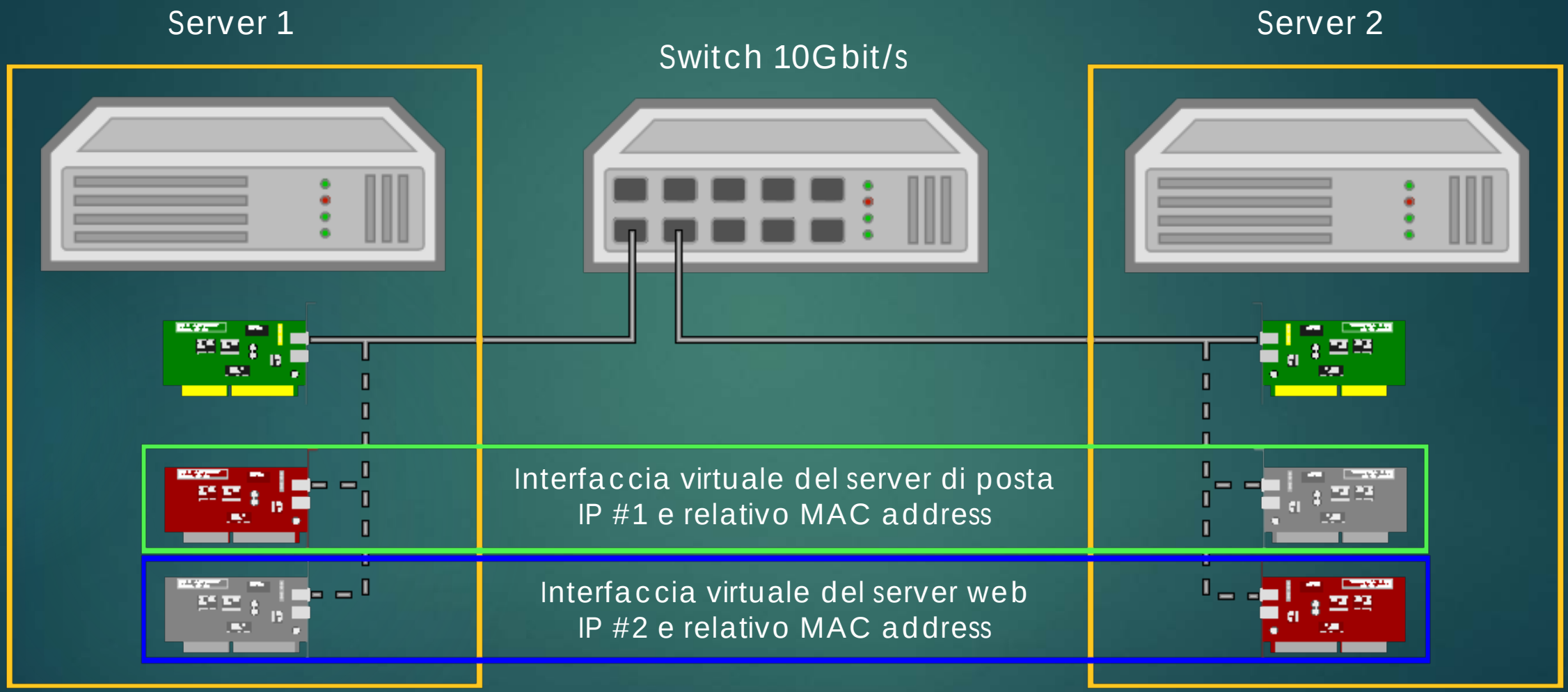
Keepalived (2)

L'utilizzo diretto delle interfacce fisiche pone il problema del loro MAC address che è differente. Keepalived può mandare un «gratuitous ARP» quando avviene una transizione da backup a master ma la cosa non è del tutto trasparente per il software cliente e non è immediata.

VRRP prevede di poter definire per ciascuna istanza una interfaccia virtuale, con uno o più numeri IP assegnati e con il proprio MAC address uguale per tutti i server. Il funzionamento per il resto è identico. Noi abbiamo definita una istanza riservata al servizio di posta elettronica e una per il server web

Restano alcuni problemi legati alla temporizzazione all'avvio. Inoltre i processi, identici nei due server, devono poter aprire socket sui numeri IP virtuali anche quando essi non siano assegnati (backup). Allo scopo, occorre modificare alcuni parametri di sistema (IPv4) e utilizzare qualche «trucco» (IPv6)

Keepalived (3)



Keepalived (4)

```
enp4s0f0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 193.206.154.110  netmask 255.255.254.0  broadcast 193.206.155.255
        inet6 2001:760:2a01::c1ce:9a6e  prefixlen 64  scopeid 0x0<global>
        inet6 fe80::225:90ff:fef9:6f3e  prefixlen 64  scopeid 0x20<link>
        ether 00:25:90:f9:6f:3e  txqueuelen 1000  (Ethernet)
        RX packets 2415868052  bytes 1177368072780 (1.0 TiB)
        RX errors 85  dropped 1431533  overruns 0  frame 85
        TX packets 2553166477  bytes 1373549542944 (1.2 TiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        inet6 2001:760:2a01::c1ce:9aaa  prefixlen 128  scopeid 0x0<global>
        inet6 2001:760:2a01::c1ce:9a21  prefixlen 128  scopeid 0x0<global>
        loop  txqueuelen 0  (Local Loopback)
        RX packets 2812861796  bytes 1809183846045 (1.6 TiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2812861796  bytes 1809183846045 (1.6 TiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
vrrp.51: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet6 fe80::200:5eff:fe00:133  prefixlen 64  scopeid 0x20<link>
        ether 00:00:5e:00:01:33  txqueuelen 0  (Ethernet)
        RX packets 9126121  bytes 1077762725 (1.0 GiB)
        RX errors 0  dropped 301312  overruns 0  frame 0
        TX packets 3  bytes 258 (258.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

vrrp.52: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 193.206.154.170  netmask 255.255.255.255  broadcast 0.0.0.0
        inet6 2001:760:2a01::c1ce:9aaa  prefixlen 128  scopeid 0x0<global>
        inet6 fe80::200:5eff:fe00:134  prefixlen 64  scopeid 0x20<link>
        ether 00:00:5e:00:01:34  txqueuelen 0  (Ethernet)
        RX packets 46451541  bytes 10755586379 (10.0 GiB)
        RX errors 0  dropped 301312  overruns 0  frame 0
        TX packets 437154  bytes 18547686 (17.6 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Problema n. 2

Configurazione dello spazio disco

Il nostro scopo era di creare uno spazio dati replicato sui due server. Soluzioni del tipo di rsync non sono l'ideale perché non garantiscono che i dati vengano aggiornati in tempo reale

La soluzione è utilizzare un file system distribuito che consenta anche una configurazione di replica. Abbiamo scelto GlusterFS, perché disponibile nella distribuzione di CentOS oppure direttamente dal sito **http://www.glusterfs.org**. GlusterFS è manutenuto e sviluppato da RedHat che ha acquistato la compagnia che lo ha creato

GlusterFS (1)

GlusterFS è un filesystem distribuito che gira nello user space. Non gestisce direttamente i dischi ma i dati vengono salvati in directory, chiamate *brick*, su dischi formattati con uno dei filesystem standard di Linux. Non usa un database centralizzato per i metadata che sono invece salvati negli attributi estesi dei file

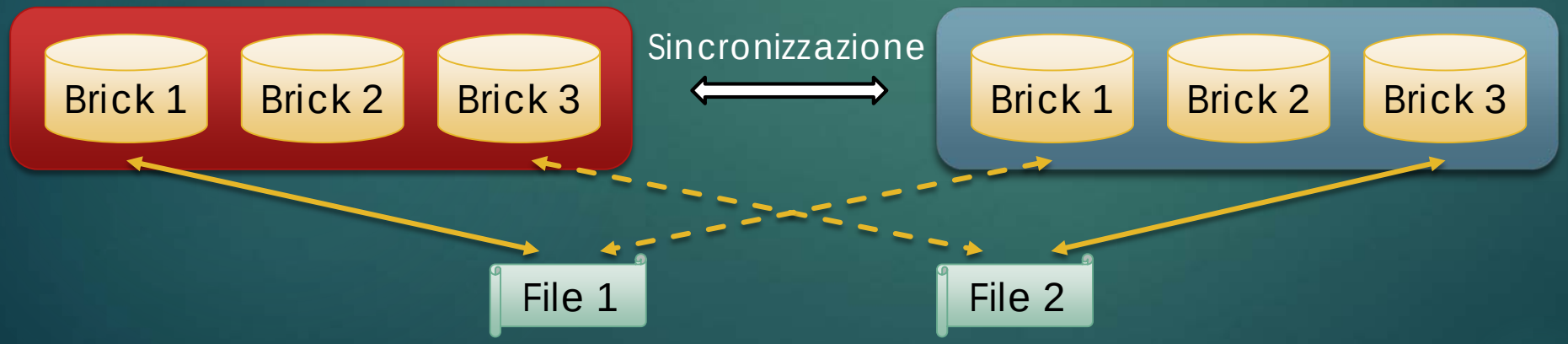
I brick sono assemblati a formare un unico filesystem logico che può essere montato via Filesystem in USErspace. FUSE è un modulo del kernel che va aggiunto al boot

GlusterFS offre anche la funzionalità di server NFS con il vantaggio di un accesso diretto al demone

GlusterFS (2)

Sono possibili varie configurazioni per i volumi GlusterFS: striped, replicated, distributed e combinazioni delle stesse. È possibile aggiungere, togliere o sostituire brick con il volume attivo. È possibile replicare geograficamente i volumi, attraverso Internet

Nel nostro caso, per lo spazio utenti, abbiamo scelto una configurazione distributed/replicated. Ognuno dei server ha copia di tutti i brick del volume. Le partizioni vengono assemblate in un solo volume e replicate tra i due server. I file vengono salvati atomicamente (non striped)



GlusterFS (3)

Limiti di GlusterFS

- GlusterFS non è estremamente performante, soprattutto con file di piccole dimensioni. La velocità della rete è fondamentale per le prestazioni del sistema
- Il server NFS integrato non supporta NFSv4, non supporta il controllo di accesso via netgroup e ha dato alcuni problemi di compatibilità
- Il fenomeno dei file in *split-brain* può essere fastidioso. Non esiste un modo automatico per risolvere i conflitti ma si deve procedere a mano, direttamente sui file nei brick

GlusterFS (4)

Esempio di situazione che porta a file in split-brain

- Server1 fa un reboot
- Server2 subentra e comincia a scrivere sui propri brick. Alcuni file vengono marcati come «dirty», in attesa di essere inviati a Server1
- Server1 è ancora offline quando anche Server2 fa un reboot
- Server1 torna online prima di Server2 e comincia a scrivere sui propri brick
- Alcuni file, modificati da Server2 quando Server1 era offline e pertanto non sincronizzati, sono modificati anche da Server1
- Quando Server2 torna online, alcuni file risultano «dirty» su entrambi i server e si ha la condizione di split-brain

La situazione va corretta a mano, cercando i file in split-brain direttamente nei brick e scegliendo quale delle due copie tenere. Fatta la scelta, si può azzerare uno specifico attributo del file da scartare o semplicemente cancellarlo e aspettare che GlusterFS sincronizzi i dischi

Problema n. 3

Replica delle informazioni di sistema

È necessario mantenere aggiornati in tempo reale alcuni database fondamentali. In particolare **LDAP** e **MySQL**

Il primo contiene le informazioni di login ma anche informazioni necessarie a una serie di altri servizi

MySQL nel nostro caso serve il database sul quale si basa il sito web e, tra le altre cose, i database del nostro sistema interno di ticket di supporto

LDAP

LDAP è un sistema molto articolato e può contenere una notevole mole di informazioni specifiche. Nel nostro caso, tra le altre cose:

- informazioni circa Samba
- il formato di archiviazione della posta e la sua collocazione
- gli alias personali di posta e informazioni di qualifica e logistiche (telefono, stanza ecc.)
- informazioni di inventario delle apparecchiature informatiche

Abbiamo ottenuto da **IANA** un **OID** (**O**bject **ID**) specifico per Arcetri che ci permette di creare schemi LDAP privati senza entrare in conflitto con ciò che già esiste. Nel nostro caso abbiamo creato uno schema con alcune informazioni aggiuntive per il nostro inventario interno.

Pur non essendo semplicissimo da configurare, LDAP possiede un motore di replica nativo che funziona perfettamente e in tempo reale

MySQL (1)

MySQL esiste attualmente in due versioni tra loro compatibili: **MySQL** e **MariaDB**

MariaDB, da noi adottata, è la versione Open Source non proprietaria di MySQL. La separazione in due rami è avvenuta dopo l'acquisto di MySQL da parte di SUN Microsystem, ora Oracle. Fino alla versione 5.5 c'è totale compatibilità a livello binario. La successiva versione 10 di MariaDB, invece, dichiaratamente diverge

MySQL e di conseguenza MariaDB ha un sistema di replica integrato in grado di sostenere una configurazione master/master. Può essere semplicemente gestito con tool del tipo di phpMyAdmin

Il sistema si è dimostrato poco affidabile. In caso di perdita di informazione o di dati in split-brain i due database divergono e non sono più sincronizzabili

MySQL (2)

Esistono alcune soluzioni più solide per la sincronizzazione in tempo reale. Alcune di esse non sono di facile configurazione e richiedono un processo esterno

Tra queste abbiamo scelta Galera Cluster, implementata in una release specifica ma ufficiale di MariaDB. Dal sito di MariaDB:

MariaDB/Galera è un cluster sincrono multi-master per i database MariaDB/InnoDB. Tra le altre funzionalità:

- Replica sincrona
- Topologia multi-master active-active
- Legge e scrive in tutti i nodi del cluster
- Controllo automatico dei membri, i nodi che falliscono vengono esclusi dal cluster
- Unione dei nodi automatica
- Vera replica parallela, a livello di riga
- Connessioni dirette dei client, look & feel nativo di MySQL

Per la soluzione automatica delle condizioni di split-brain occorrono tre copie del database, ma MariaDB distribuisce anche un demone arbitro che non conserva tutto il database ma tiene traccia delle modifiche

Conclusioni

Il sistema è attivo ad Arcetri da Marzo scorso. Da simulazioni effettuate sembra rispondere bene alle aspettative, ma i problemi si vedranno nelle vere situazioni di crisi

Alcune criticità:

1) Linux è un sistema in continua evoluzione e a volte vengono introdotti bugs. Questo vale anche per GlusterFS che inopinatamente distribuisce dal repository la versione di sviluppo invece della versione stabile. Attenzione agli upgrade automatici!

2) È fondamentale la tempistica e l'ordine di avvio dei servizi. Succedono talvolta cose strane e non prevedibili. Systemctl non sempre è affidabile da questo punto di vista

3) Occorre fare attenzione all'ordine e alla tempistica con cui vengono fatti gli upgrade di glusterfs sui due server, oppure i reboot, altrimenti si rischia di dover risolvere dozzine di situazioni di split-brain, in particolare sui file di dovecot

Questa configurazione è HA a livello di server ma non di processo: se il server ha keepealived attivo e tutto il resto non va non c'è failover dei servizi. Per un configurazione HA più puntuale si possono usare ad es. Pacemaker e Corosync che sono però più complessi da configurare e mantenere

Riferimenti

Keepalived e **V**irtual **R**outer **R**edundancy **P**rotocol:

http://www.keepalived.org
http://tools.ietf.org/html/rfc3768 (per i ***nerd***)
http://en.wikipedia.org/wiki/Virtual_Router_Redundancy_Protocol (per tutti gli altri)

GlusterFS

http://www.gluster.org
http://en.wikipedia.org/wiki/GlusterFS

MariaDB e **Galera Cluster**

http://mariadb.org
http://galeracluster.com

IANA - lista **OID** assegnati e modulo per la richiesta

http://www.iana.org/assignments/enterprise-numbers/enterprise-numbers
http://pen.iana.org/pen/PenApplication.page