Infrastruttura di autenticazione distribuita per servizi Cloud

Stato dell'arte, attività, proposte

Cristina Aiftimiei - cristina.aiftimiei@pd.infn.it
Stefano Stalio - stefano.stalio@lngs.infn.it

Mini workshop CCR
Laboratori Nazionali del Gran Sasso

February 28, 2014



Obiettivi

Alcuni degli obiettivi, legati alle problematiche di autenticazione ed autorizzazione, del **Cloud Working Group** (Cloud-wg) INFN:

- Consentire agli utenti INFN l'**accesso a risorse cloud distribuite nelle sedi INFN** con un livello di trasparenza ancora da definire, dipendente anche dall'applicazione, ed attraverso un **sistema Single Sign On** (SSO) agganciato ad **INFN AAI**;
- Consentire agli utenti INFN l'**utilizzo di risorse cloud esterne alle sedi INFN** - cloud pubbliche o cloud private di altre istituzioni - con un livello di trasparenza ancora da definire, attraverso lo stesso meccanismo di autenticazione ed autorizzazione;
- Consentire l'**accesso, ad utenti appartenenti ad enti esterni all'INFN**, a risorse cloud distribuite nelle sedi INFN, attraverso il sistema SSO utilizzato dall'ente di afferenza.

Integrazione Keystone/AAI

La prima necessità è stata quella di trovare il modo di integrare l'*Identity Service* di OpenStack (**Keystone**) con **INFN AAI**.

- **LNGS** - Proof Of Concept: si potrebbe utilizzare un ramo ldap di AAI come back-end Keystone, ma difficoltà di integrazione con AAI di altre istituzioni;
- **Bari** - Un layer SAML permette la comunicazione tra Horizon e l'IdP di INFN AAI, quando un utente INFN accede per la prima volta un account Keystone per quel *username* viene creato automaticamente in un *tenant* dedicato;
- **Padova** - Approccio abbastanza simile al precedente, prevista integrazione con gli IdP IDEM e con OpenID. Questi meccanismi sono usati esclusivamente per l'autenticazione, non per l'autorizzazione;

Integrazione Keystone/AAI

- **Catania** - All'inizio un approccio simile a Bari e Padova ma:

This approach was temporary discarded because of a relevant drawback: other tools in OpenStack need to authenticate against Keystone and do not support SAML, therefore using SAML with apache would require to modify all of them
...
Considering, the effort to maintain all the changes in future versions we decided to stop and contacted the developers to understand if it is possible to integrate a better support for SAML even if this would require some effort from the INFN to help with SAML integration. From the developer team several approaches showing some integration between SAML and keystone were highlighted and suggested to test.

Proposta di integrazione Keystone/VOMS

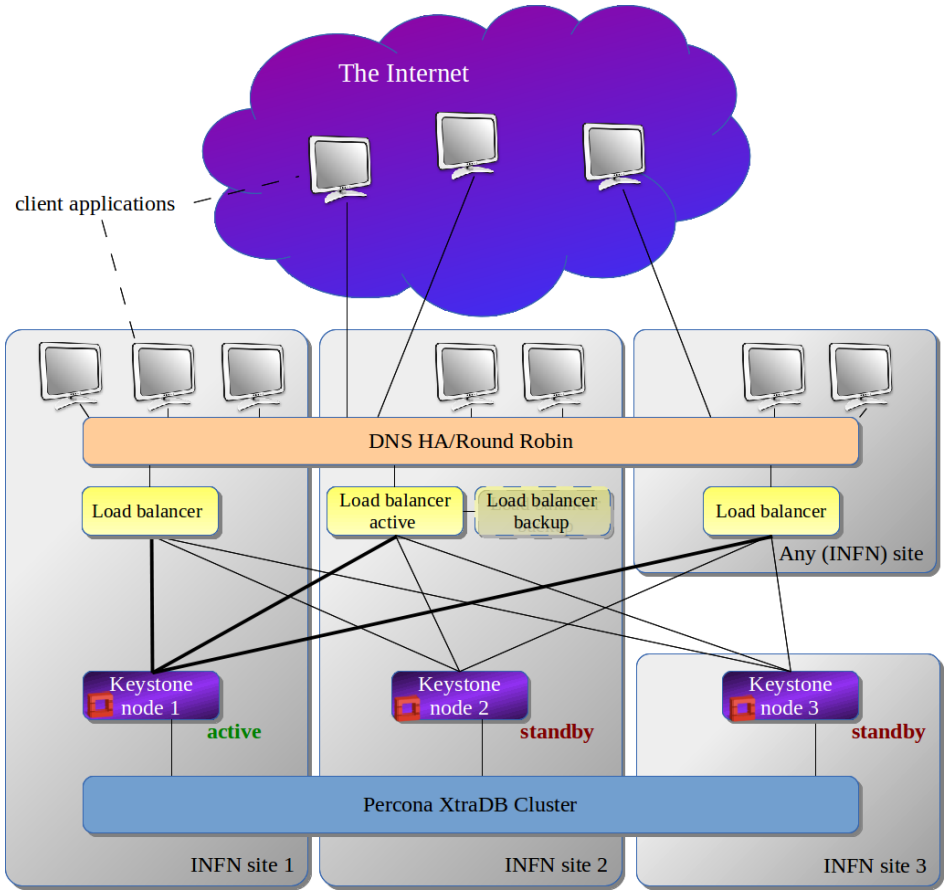
- Riportare molti concetti grid in ambito cloud:
  - **uso di VOMS per l'autorizzazione federata**;
  - distribuzione ed applicazione di policy definite dalle VO;
  - definizione a livello di servizio di profili per l'applicazione delle policy in base agli attributi.
- Il servizio VOMS deve:
  - diventare un Attribute Provider SAML2 (molte parti mancanti nell'architettura);
  - consentire una mappatura configurabile fra attributi di una federazione e attributi di una VO.
- A partire da Keystone, e nel caso altri servizi a seguire, si deve:
  - identificare un profilo per l'attuazione delle policy distribuite dalle VO;
  - implementare le corrette azioni previste dal profilo (es: mapping su utenti cloud locali).

Servizio Keystone distribuito

Il servizio di identità di una infrastruttura cloud distribuita deve essere **sempre disponibile**, anche in caso di interruzione del collegamento ad un intero sito, e deve perciò essere dislocato in più sedi.

- Questo obiettivo è stato raggiunto realizzando un cluster di tre server Keystone in tre sedi diverse sedi INFN: **Padova** per il Nord Italia, i **LNGS** per il Centro, e **Bari** per il Sud Italia.
- Il database SQL usato come back-end è replicato sui tre siti usando **Percona XtraDB Cluster**, una soluzione di alta disponibilità per cluster MySQL.
- Per ottimizzare le performance, poiché la replica è sincrona, e garantire la consistenza delle transazioni, si è preferito fare in modo che uno solo dei server Keystone sia attivo in ogni istante, mentre un secondo server è pronto a prenderne il posto.
- Il terzo server ha lo scopo di contribuire a stabilire il quorum del cluster e può diventare il server attivo nel caso improbabile di failure dei primi due.

Servizio Keystone distribuito



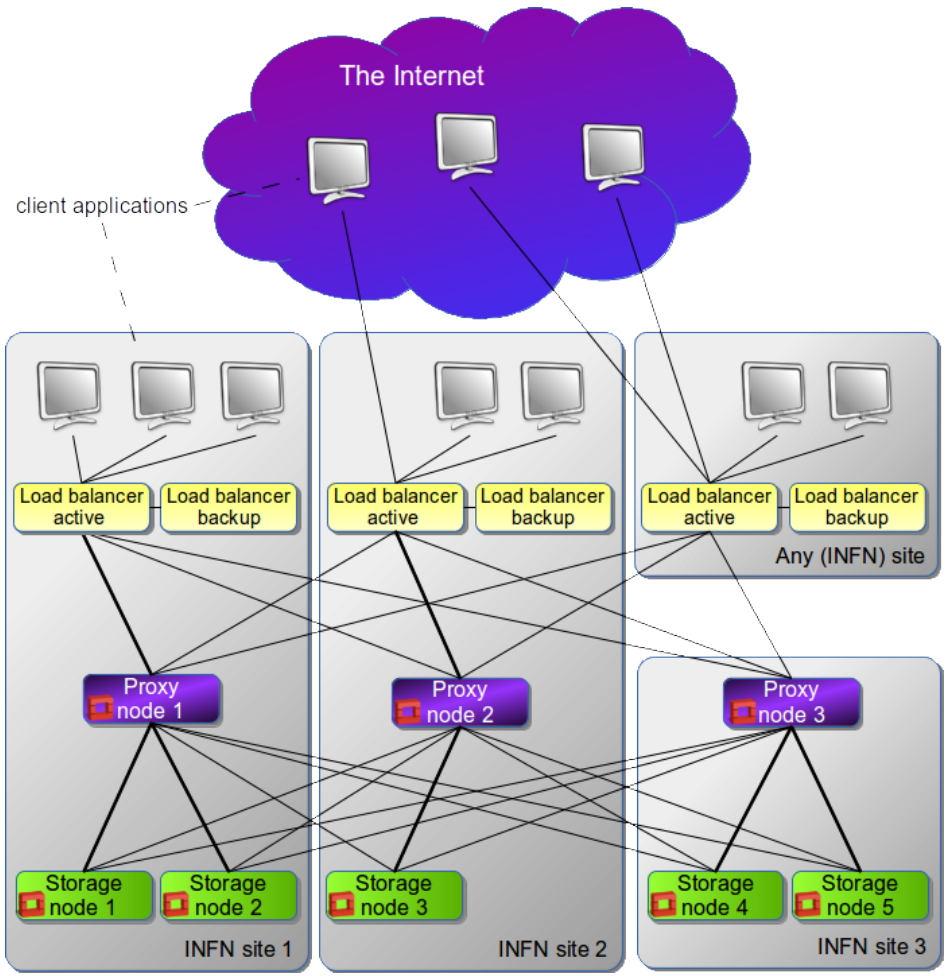
- **HAProxy** è usato come *load balancer* e come *HA provider* per il servizio Keystone, mentre il **servizio DNS riconfigurabile dinamicamente** offerto da **ha.infn.it** garantisce *fault protection* per i server proxy.
- Per l'integrazione con AAI, e non solo, la comunicazione tra i client ed i proxy server, tra i proxy server e i server Keystone e tra i server Keystone è criptata (SSL).

- Questa architettura replicata è completamente trasparente agli utenti e non ha singoli punti critici. I problemi sui singoli componenti sono notificati agli amministratori di sistema da un **sistema di monitor della rete**.

Servizio Keystone distribuito
Monitoring dei componenti

- I server Keystone di Bari, Padova e LNGS, ed i server HAproxy presenti nelle stesse sedi **sono costantemente monitorati** dal servizio Nagios attivo presso i LNGS grazie ad un controllo sulla raggiungibilità delle porte tcp 5000 e 35357;
- in più lo stesso **Nagios richiede periodicamente un token** al servizio Keystone distribuito, per verificare l'effettivo funzionamento di tutta la catena autenticativa;
- il servizio **Nagios di ha.infn.it** invece verifica periodicamente la raggiungibilità dei proxy server e **modifica le tabelle DNS** quando uno dei proxy server diventa irraggiungibile o torna attivo dopo un periodo di indisponibilità;
- i proxy server stessi verificano la raggiungibilità dei server Keystone e dirigono le richieste di autenticazione verso i soli server attivi.

Servizio Object Storage distribuito

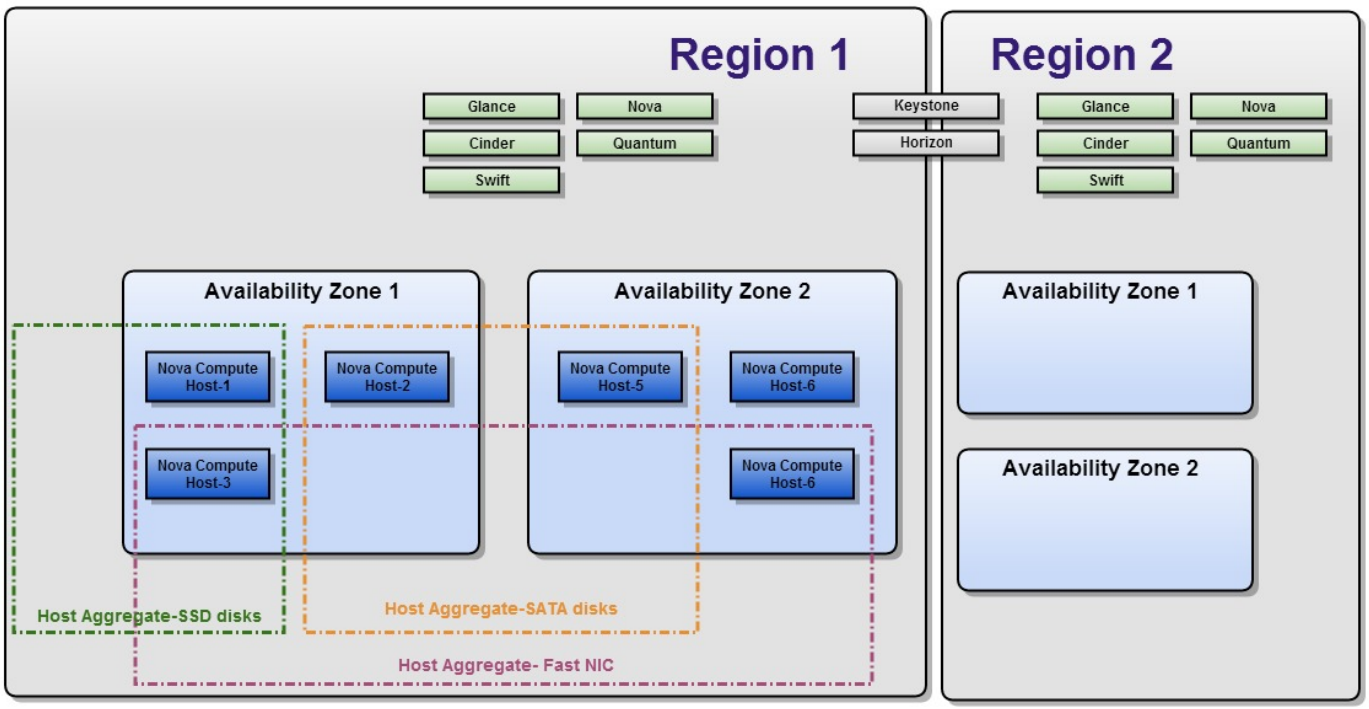


Il primo servizio ad appoggiarsi all'infrastruttura di autenticazione appena descritta è un **servizio di object storage** basato su OpenStack Swift, realizzato nelle stesse tre sedi. Anche questo setup è stato pensato per essere completamente ridondante e privo di singoli punti critici, ma deve essere ancora provato in condizioni di carico pesante.

Resta da risolvere il problema della criptazione del dei dati durante i trasferimenti tra storage node e proxy node e durante la replica tra storage node.

Unico dominio amministrativo (?)

La realizzazione di un servizio Keystone distribuito sul territorio ha mostrato che è possibile avere un'**infrastruttura di autenticazione ed autorizzazione centralizzata con ottime caratteristiche di fault tolerance** la quale può essere di supporto per la realizzazione di una Cloud INFN con risorse dislocate sul territorio ma sotto un unico dominio amministrativo.



Interazione tra e con risorse distribuite

In questo modello **utenti, gruppi e progetti sono condivisi** tra tutte le regioni, mentre **le risorse allocate per ogni progetto (quote) sono diverse in ogni regione**.

La gestione delle risorse (istanze, rete, block storage, immagini) è anch'essa propria di ogni regione.

Alcuni servizi (es. image service, object storage) possono essere condivisi tra più regioni mentre altri (compute, network, block storage) devono essere implementati in ognuna di esse.

Interazione tra e con risorse distribuite

ATTENZIONE!

Oggi non c'è modo di assegnare ad un utente un insieme di ruoli che ne facciano un *region administrator*, **chiunque abbia il ruolo di amministratore può di fatto amministrare tutti i servizi in ogni regione**.

Questo è un problema aperto e gli sviluppatori di OpenStack dovrebbero essere contattati per verificare la possibilità di avere, in futuro, la possibilità di definire il ruolo di **region administrator**.

Più in generale, il modo in cui Keystone gestisce oggi il ruolo di amministratore è considerato un bug (https://bugs.launchpad.net/keystone/+bug/968696) e ci possiamo aspettare che questo cambi in tempi brevi.

https://bugs.launchpad.net/keystone/+bug/968696

Fact: Keystone's rbac model grants roles to users on specific tenants, and post-keystone redux, there are no longer "global" roles.

Problem: Granting a user an "admin" role on ANY tenant grants them unlimited "admin"ness throughout the system because there is no differentiation between a scoped "admin"ness and a global "admin"ness.

I don't have a specific solution to advocate, but being an admin on *any* tenant simply *cannot* allow you to administer all of keystone.

Steps to reproduce (from Horizon, though you could do this with the CLI, too):

1. User A (existing admin) creates Project B and User B.
2. User A adds User B to Project B with the admin role on Project B.
3. User B logs in and now has unlimited admin rights not only to view things in the dashboard, but to take actions like creating new projects and users, managing existing projects and users, etc.

Interazione tra e con risorse distribuite - Domini

La divisione in **domini**, che sarà pienamente supportata nelle prossime versioni di OpenStack, permette di **delegare la gestione di utenti, gruppi e progetti**.
I domini possono essere di grande utilità, ma bisogna verificare che siano compatibili con una infrastruttura cloud multi-regione.

https://wiki.openstack.org/wiki/Domains

The intent of domain is to define the administrative boundaries for management of Keystone entities. A domain can represent an individual, company, or operator owned space.

Bisogna anche capire se con la API v3 di Keystone, oltre al supporto per i domini, sia possibile una gestione delle policy più adatta alle esigenze di una infrastruttura distribuita. Sui domini:

http://www.mirantis.com/blog/manage-openstack-projects-using-domains-havana

Conclusioni

- Ad oggi non abbiamo modo di autenticarci attraverso un IdP SAML direttamente sulle API OpenStack, senza utilizzare la dashboard Horizon, ma contiamo di poterlo fare in un futuro prossimo.
- Il modello di autorizzazione degli utenti siano essi appartenenti all'INFN o ad istituzioni esterne è ancora da stabilire.
- Abbiamo un servizio di identità distribuito funzionante con caratteristiche di alta disponibilità. Bisogna verificarne l'effettiva fault tolerance sul lungo periodo ed eventuali problemi di performance se vi si dovessero appoggiare più installazioni cloud distribuite nelle sedi INFN.

Conclusioni

- Abbiamo formulato un'ipotesi di architettura distribuita per la Cloud INFN (singolo dominio amministrativo, con possibilità di delega per la gestione di utenti, gruppi e progetti) basata sugli strumenti oggi a disposizione. In futuro, grazie agli sviluppi previsti – ed in parte già attuati - su Keystone, l'architettura ipotizzata sarà più flessibile ed avrà caratteristiche più adatte alle nostre esigenze. Tale ipotesi di architettura deve essere validata con **testbed coordinati tra più sedi**.
- La Cloud INFN deve potersi federare con installazioni cloud appartenenti ad altre istituzioni. Come questo si possa realizzare è ancora da definire.

Persone


Eric Frizziero
Alvise Dorigo
Riccardo Veraldi
CLOUD-WG
Stefano Stalio
Pasquale Notarangelo
Marco Fargetta
Cristina Aiftimiei
Marica Antonacci
Andrea Ceccanti
Paolo Andreetto
Matteo Panella
Sara Bertocco
Giacinto Donvito