



Servizi di Backup dell'infrastruttura IaaS PRISMA



Stato del deliverable

Ver.	Data	Autore della modifica	Note	Validazione
1.0	30-01-2015	INFN-BARI	Prima Stesura	INFN



INDICE DEGLI ARGOMENTI

1. INTRODUZIONE	4
2. BACKUP DELLE MACCHINE VIRTUALI	4
3. BACKUP DELLE MACCHINE VIRTUALI	6
4. BACKUP DI DATABASE MONGODB E MYSQL	8
4.1 Gestione delle chiavi per la cifratura del backup	9
5. PROCEDURE DI RIPRISTINO	15
5.1 Ripristino della macchina virtuale	15
5.2 Ripristino del volume	16
5.3 Ripristino del database	17
5.3.1 MongoDB	19
5.3.2 MySQL	19
5.4 Considerazioni finali	19



1. Introduzione

L'infrastruttura PRISMA-iaaS fornisce le seguenti funzionalità di backup:

- Backup delle macchine virtuali (VM: virtual machine)
- Backup dei volumi (block device)
- Backup dei database (supportati: mongodb, mysql)

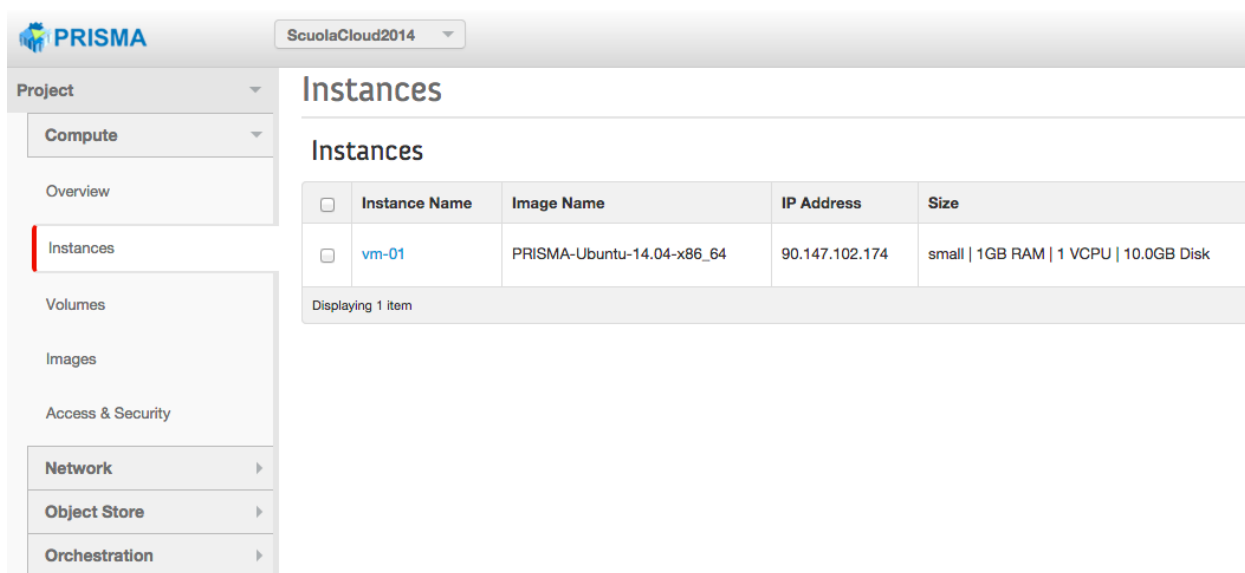
Nel seguito, le tre tipologie di backup saranno descritte in dettaglio.

2. Backup delle macchine virtuali

Esistono due modalità di backup della macchina virtuale:

- 1) modalità manuale
- 2) modalità automatica

La modalità **manuale** consiste nell'esecuzione di uno snapshot dell'istanza virtuale e può essere effettuata in qualsiasi momento dall'utente attraverso la dashboard (<https://prisma-cloud.ba.infn.it>) accedendo alla pagina delle istanze dal menù Project>Compute>Instances e poi cliccando sul bottone "Create snapshot", come mostrato nella figura seguente.



The screenshot shows the PRISMA dashboard interface. At the top, there's a header with the PRISMA logo and a dropdown menu set to 'ScuolaCloud2014'. Below the header, a sidebar on the left contains a navigation menu with options: Project, Compute, Overview, Instances (highlighted with a red bar), Volumes, Images, Access & Security, Network, Object Store, and Orchestration. The main content area is titled 'Instances' and displays a table with the following data:

	Instance Name	Image Name	IP Address	Size
<input type="checkbox"/>	vm-01	PRISMA-Ubuntu-14.04-x86_64	90.147.102.174	small 1GB RAM 1 VCPU 10.0GB Disk

Below the table, it says 'Displaying 1 item'.



Instances

Instances

<input type="checkbox"/>	Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Uptime	Actions
<input type="checkbox"/>	vm-01	PRISMA-Ubuntu-14.04-x86_64	90.147.102.174	small 1GB RAM 1 VCPU 10.0GB Disk	marica-cloudify-key	Active	nova	None	Running	0 minutes	<input type="button" value="Create Snapshot"/> <input type="button" value="More"/>

Displaying 1 item

Gli snapshot così creati sono visibili accedendo al menù Project>Images

PRISMA ScuolaCloud2014

Project

Images

<input type="checkbox"/>	Image Name	Type	Status	Public
<input type="checkbox"/>	vm-01-snap29122014	Snapshot	Active	No
<input type="checkbox"/>	vm-01-snap30012015	Snapshot	Active	No

Displaying 2 items

Il backup **automatico** delle macchine virtuali è un servizio fornito su richiesta agli utenti dell'infrastruttura e prevede la creazione degli snapshot secondo uno schema di salvataggio e rotazione predefinito:

- il backup è giornaliero ed avviene in una finestra temporale intorno alle 04:00;
- la rotazione degli snapshot è implementata in modo tale che siano sempre disponibili gli snapshot relativi agli ultimi 7 giorni e due snapshot quindicinali.

Anche gli snapshot automatici sono visibili nella pagina Project>Images della dashboard e riconoscibili dalla presenza della stringa “auto” nel nome dello snapshot.

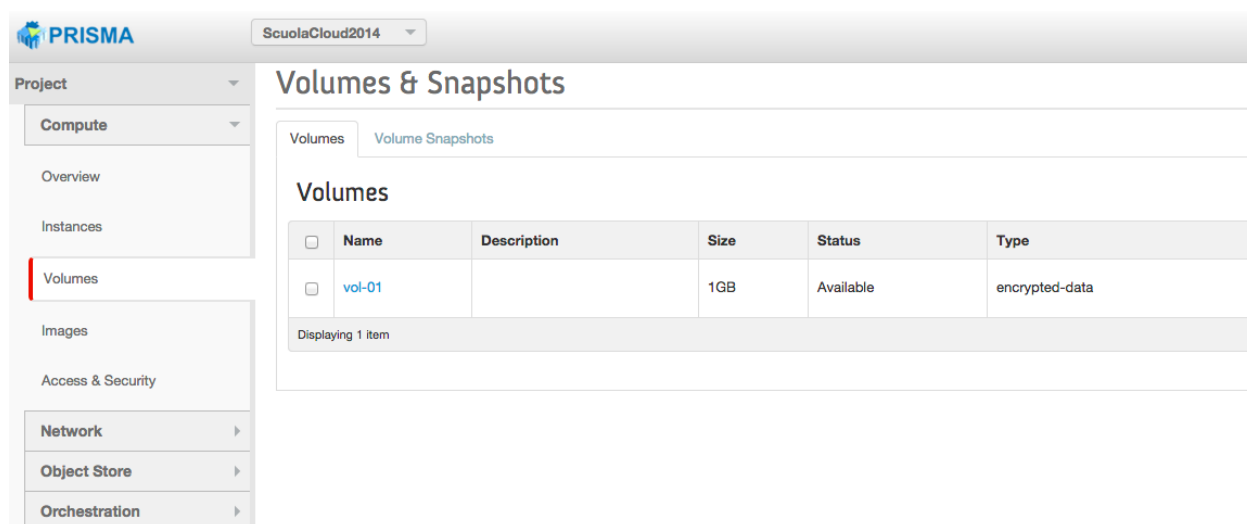


3. Backup delle macchine virtuali

Analogamente al backup delle VM, esistono due modalità di backup dei volumi (block device) agganciati alle VM:

- 3) modalità manuale
- 4) modalità automatica

La modalità **manuale** consiste nell'esecuzione di uno snapshot del volume e può essere effettuato in qualsiasi momento dall'utente attraverso la dashboard (<https://prisma-cloud.ba.infn.it>) accedendo alla pagina delle istanze dal menù Project>Compute>Volumes e poi cliccando sul bottone “More” e scegliendo dal menu a tendina “Create Snapshot”, come mostrato nella figura seguente.



PRISMA ScuolaCloud2014

Project: Compute

Overview
Instances
Volumes
Images
Access & Security
Network
Object Store
Orchestration

Volumes & Snapshots

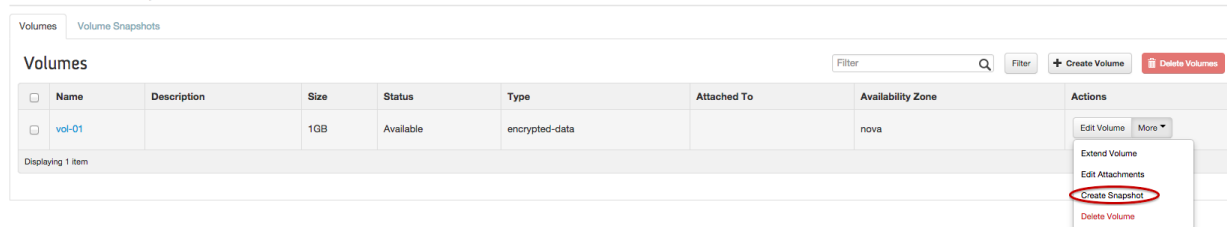
Volumes Volume Snapshots

Volumes

<input type="checkbox"/>	Name	Description	Size	Status	Type
<input type="checkbox"/>	vol-01		1GB	Available	encrypted-data

Displaying 1 item

Volumes & Snapshots



Volumes Volume Snapshots

Volumes

Filter Filter

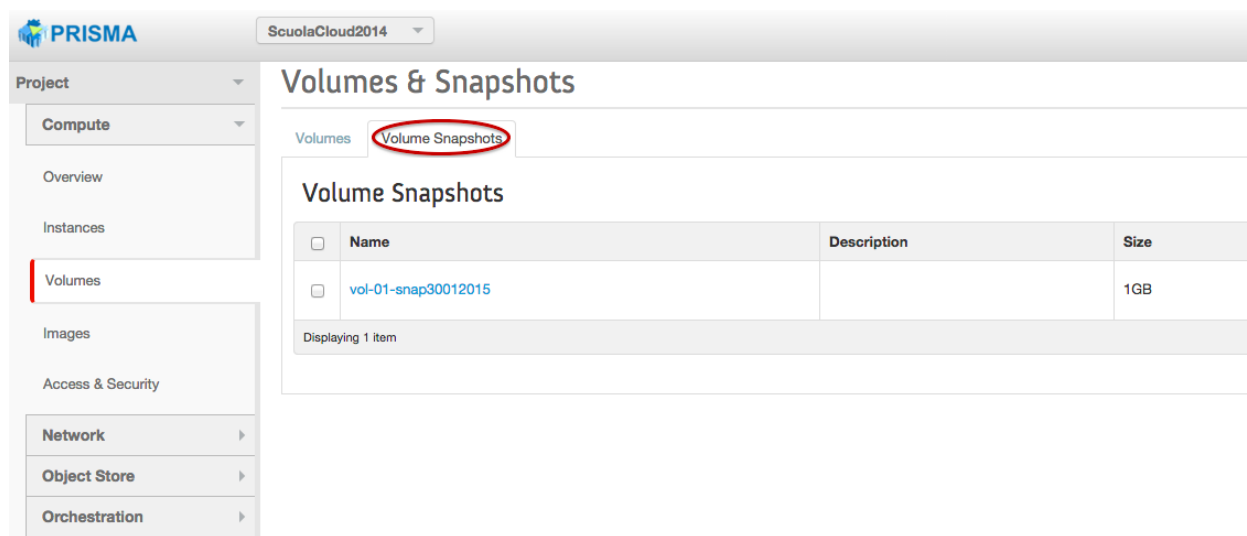
<input type="checkbox"/>	Name	Description	Size	Status	Type	Attached To	Availability Zone	Actions
<input type="checkbox"/>	vol-01		1GB	Available	encrypted-data		nova	<input type="button" value="Edit Volume"/> <input type="button" value="More"/>

Displaying 1 item

Actions dropdown menu:

- Edit Volume
- Extend Volume
- Edit Attachments
- Create Snapshot**
- Delete Volume

Gli snapshot sono visibili cliccando sul tab “Volume Snapshots” nella stessa pagina:



PRISMA ScuolaCloud2014

Project

Compute

Overview

Instances

Volumes

Images

Access & Security

Network

Object Store

Orchestration

Volumes & Snapshots

Volumes Volume Snapshots

Volume Snapshots

<input type="checkbox"/>	Name	Description	Size
<input type="checkbox"/>	vol-01-snap30012015		1GB

Displaying 1 item

Il backup **automatico** dei volumi è un servizio fornito su richiesta agli utenti dell'infrastruttura e prevede la creazione degli snapshot secondo uno schema di salvataggio e rotazione predefinito:

- il backup avviene ogni 15 giorni durante la notte;
- la rotazione degli snapshot è implementata in modo tale che siano sempre disponibili gli ultimi due snapshot quindicinali.

Anche gli snapshot automatici sono visibili nel tab "Volume Snapshots" e riconoscibili dalla presenza della stringa "*auto*" nel nome dello snapshot.



4. Backup di database MongoDB e MySQL

Per gli utenti del servizio DBaaS (DataBase as a Service) dell'infrastruttura PRISMA-IAAS è possibile attivare su richiesta il backup automatico dei dati memorizzati con uno schedule predefinito e dettagliato in questo paragrafo.

Al momento la funzionalità è disponibile per i DBMS MongoDB e MySQL.

Il backup del database viene effettuato ogni giorno, e viene creato e salvato un nuovo archivio compresso nella cartella `/var/backups/mongodb` (nel caso di backup di mongodb) o `/var/backups/mysql` (nel caso di backup di mysql).

In particolare, all'interno di queste cartelle, il nuovo backup viene salvato in una sottocartella a scelta fra `/var/backups/mongodb/daily`, `/var/backups/mongodb/weekly` o `/var/backups/mongodb/monthly` (analogamente per mysql). La sottocartella viene scelta secondo questo criterio:

- se è il giorno 1 del mese, il nuovo backup viene salvato in `/var/backups/mongodb/monthly` e `/var/backups/mysql/monthly`;
- se è lunedì, il nuovo backup viene salvato in `/var/backups/mongodb/weekly` e `/var/backups/mysql/weekly`;
- altrimenti, se nessuna delle precedenti condizioni è vera, il file viene salvato in `/var/backups/mongodb/daily` e `/var/backups/mysql/daily`.

Questa distinzione serve perché ciascuna cartella ha una diversa regola di rotazione dei file, cioè di eliminazione dei file vecchi. La rotazione avviene secondo questa procedura:

- i file salvati nelle cartelle *monthly* non vengono mai cancellati;
- le cartelle *weekly* conservano lo storico dei backup relativi agli ultimi quattro lunedì. Questo vuol dire che in queste cartelle si troveranno, a regime, sempre quattro file e, quando il quinto viene aggiunto, il più vecchio di questi quattro file viene cancellato;
- i file salvati nelle cartelle *daily* conservano lo storico dell'ultima settimana. A regime, ciascuna cartella *daily* conterrà sei file e, quando il settimo viene aggiunto, il più vecchio di questi sei file viene cancellato.

Si ha quindi sempre a disposizione uno storico dei database di un file al mese, che non viene mai cancellato. Questo assicura che backup vecchi siano comunque conservati, senza però gravare troppo sullo spazio disco occupato. Ogni backup è completo, e contiene l'intero database. Un link all'ultimo backup effettuato viene salvato nelle cartelle `/var/backups/mongodb/latest` e `/var/backups/mysql/latest`, in modo da sapere sempre qual è l'ultimo file salvato e, di conseguenza, in quale cartella si trova.



Gli archivi così generati vengono cifrati usando la suite GnuPG¹ (già installato nella maggior parte dei sistemi operativi GNU/Linux) e poi caricati sull'Object Store dell'infrastruttura PRISMA-IaaS.

Per usare il sistema GnuPG c'è bisogno di una chiave pubblica e una chiave privata. La cifratura avviene con la chiave pubblica, mentre la decodifica avviene con la chiave privata.

L'utente che richiede il servizio di backup automatico del database dovrà pertanto

1. generare una coppia di chiavi (pubblica e privata)
2. esportare la chiave pubblica e inviarla al team di supporto dell'infrastruttura PRISMA-IaaS

4.1 Gestione delle chiavi per la cifratura del backup

Step1: creazione delle due chiavi

Per generare una coppia di chiavi usare il comando `gpg` con l'opzione `--gen-keys`.

Verrà richiesto di rispondere ad una serie di domande:

- a. il tipo di chiave,
- b. la lunghezza della chiave,
- c. il tempo di scadenza della chiave,
- d. un identificativo utente (utilizzato per associare la chiave che si sta creando ad una persona reale)
- e. una passphrase per proteggere la chiave primaria.

Per le opzioni *a,b,c* il default suggerito va bene (a meno che l'utente non abbia particolari esigenze); mentre occorre prestare attenzione alle opzioni *d* ed *f*.

Qui di seguito un esempio di esecuzione del comando (in grassetto gli input utente):

```
# gpg --gen-key

gpg (GnuPG) 1.4.11; Copyright (C) 2010 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Please select what kind of key you want:
```

¹ <https://www.gnupg.org/gph/it/introduzione.html>



(1) RSA and RSA (default)

(2) DSA and Elgamal

(3) DSA (sign only)

(4) RSA (sign only)

Your selection? **1**

RSA keys may be between 1024 and 4096 bits long.

What keysize do you want? (2048) **2048**

Requested keysize is 2048 bits

Please specify how long the key should be valid.

0 = key does not expire

<n> = key expires in n days

<n>w = key expires in n weeks

<n>m = key expires in n months

<n>y = key expires in n years

Key is valid for? (0) **0**

Key does not expire at all

Is this correct? (y/N) **y**

You need a user ID to identify your key; the software constructs the user ID

from the Real Name, Comment and Email Address in this form:

"Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Real name: **Mario Rossi**

Email address: **mario.rossi@example.com**

Comment: **M. Rossi**



You selected this USER-ID:

"Mario Rossi (M. Rossi) <mario.rossi@example.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? O

You need a Passphrase to protect your secret key.

<ENTER A PASSWORD>

L'output è simile al seguente:

```
gpg: key 443EE359 marked as ultimately trusted
public and secret key created and signed.

gpg: checking the trustdb
gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
gpg: depth: 0  valid:   2  signed:   0  trust: 0-, 0q, 0n, 0m, 0f, 2u
pub   2048R/443EE359 2015-02-02
      Key fingerprint = B9A9 700E E017 084B 5340  C71A BDA2 EE49 443E
      E359
uid           Mario Rossi (M. Rossi) <mario.rossi@example.com>
sub   2048R/1B82FDB5 2015-02-02
```

Step2: esportare la chiave pubblica

Per elencare le chiavi presenti nel proprio portachiavi pubblico utilizzare l'opzione a linea di comando --list-keys.

```
# gpg --list-keys
/root/.gnupg/pubring.gpg
-----
```



```
pub 2048R/EA20E47D 2015-02-02
uid Mario Rossi (M. Rossi) <mario.rossi@example.com>
sub 2048R/27ADC454 2015-02-02

pub 2048R/443EE359 2015-02-02
uid Marica Antonacci (M. Antonacci)
<marica.antonacci@gmail.com>
sub 2048R/1B82FDB5 2015-02-02
```

Per esportare una chiave pubblica si usa l'opzione a linea di comando `--export` specificando come argomento l'identificativo della chiave da esportare:

```
# gpg --armor --output mrossi.gpg --export mario.rossi@example.com
```

La chiave pubblica esportata viene salvata nel file passato come argomento dell'opzione `--output`. Segue un esempio:

```
# cat mrossi.gpg
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1.4.11 (GNU/Linux)

mQENBFTPe6YBCACs4MEqT8c5DrMYjB6PgNZt0ww4mGdVmQDhuJwBZSRga6HYLfgs
eFZjUo/z3uv7iDlSqU1dY0WdD88kHVP/Eq3Yww/3PI5WKg9hmUdySfLTyWZAqxDU
e3u786ex1UkU0TBIfWHcAmWZMxfo6yde3jnZFE7DJGwZIsRmmGx8e2iIfixliZC3
vF4AMfFrGUHmfkQt41kR6glyDC0Zdpn+UD+dYANHjMzoM42soLesmwQMrKD2pj8C
FzhCG8geX0g8/P9ps4E2/Ld/ntgD4xvE9fsTW5FMv6XiBVqCx3qKg8gY2ioFxrkt
swW0vYMIFebwcZ1drjbFeiRl6jtwUuUtfoaZABEBAAG0ME1hcmlvIFJvc3NpICht
LiBSb3NzaSkqPG1hcmlvLnJvc3NpQGV4YW1wbGUuY29tPokBOAQTAAIAIgcUCVM97
pgIbAwYLCQgHAwIGFQgCCQoLBByCAwECHgECF4AACgkQYUJ51+og5H0WIAf/SPf/
9ACK102wa7C8cKEqYSj8MedtHXvKoH3JFp3vcqo+QLea9fICr2UGQ7q0oo09xPSP
APVgcQzOvx1thlF6dwrLs9KeeTFYFXGZ+MVJkaIhD0mGzvy0ahdjbKeoWv6lgX3M
```



```
ncA9E6OC/qzimueErU4/JcFJx9HiW0yhlI00RI7F2WGjlfqrrPotmpPs3VRsNJkj
P6X6GXnm3Eoymis0Mld7TOeIQ2lISe/IH9HKpebviN4j464dXfRvBY/ZxNAZzxSS
BFZDSI0837dv8+pJRXgrGMTUNWcPbNeU2FL2CGWy5ADTGWELb6xhI9LwljqDDsAW
lnyP6EspT+zFzmDml7kBDQRuz3umAQgAtLhHGkEb0Pldf3hsK+1HLtWqLunhIc4Z
6VPFKe3ElmNfwTmiHHT3+qkdShoPqlerGDRbGM1s6kianeeX+Kzn3DZSj+6OIG01
sMsWC0Kvb4xYscXD7Rd9aIbu6kuVHHuE8qYfQ294t2U0W+/ofn7GwysUCVXIWGac
1q9KhXyhhQKJEZwevX91lbqqhtMXSG92JbaaaY/DlpB3+n/sdMe/CmuoZVKS0z4E
didclwJqx0yec1tCSgPzksH9I2UeEZ4NpJNYe5t9GEIWyQ1RQ8QqDb4adNlqt81P
XKL402C4Q0aBBuKarOoc9NLHBZikhjNZ91F2ckXSLAy1vg5ZUL+KvQARAQABiQEf
BBgBAGAJBQJUz3umAhsMAAoJEGFCedfqiOR9h4QH/0Ij0QVQsi2oV0Np6+mOl5uI
8mimU9LdPqbyAj7qJNVFZ7xCHffJZXOzIewB57hZ+irnmlQn3BWvBMSltceBbEjK
qA6wd78k1DQAVX2JkdcLETunuDbnsSZbAPYyfeJeOtrf6/h6o+2awNYA+TAd0KV3L
VSyU2n3aBoMMoJFv9WpUQUnqWIKyn1Wv/cONBNWuwlyvXJI9CQgjfL1XTtE396D
OMhI5hTIDBsUVRPrp579EUuIymTyF8IR9OGcGraFDQNrw5NSmlFKXw8rwoMFCBmu
GSQAvAtLcvwxg/rtFrLfOq90/WLC9ejqorvQjL7mu05kTkyzfpjOPkZYhY1ZjUc=
=M4Fi
-----END PGP PUBLIC KEY BLOCK-----
```

Questo file va mandato al team di supporto di PRISMA-iaaS.

Nota: export/import di una chiave privata

La chiave privata va custodita con attenzione e può eventualmente essere esportata nel caso sia necessario installarla nel portachiavi di un altro host.

Per completezza, qui di seguito si riporta anche la procedura da seguire nel caso sia necessario esportare ed importare una chiave privata.

Per esportare la chiave usare l'opzione `--export-secret-keys` seguito dallo User ID della chiave da esportare:

```
# gpg --armor --output secret.key --export-secret-keys
mario.rossi@example.com
```



Per importare la chiave privata esportata nel file secret.key usare l'opzione --import:

```
# gpg --import secret.key
```



5. Procedure di ripristino

5.1 Ripristino della macchina virtuale

Sia nel caso di backup manuale sia nel caso di backup automatico, l'utente può facilmente ripristinare la propria macchina virtuale a partire da uno degli snapshot salvati e visualizzati nella pagina "Images" del proprio progetto.

Cliccando sul bottone "Launch" è possibile istanziare una nuova macchina virtuale a partire dallo snapshot scelto.

Images

Images							
<div>Project (2) Shared with Me (0) Public (42) + Create Image Delete Images</div>							
<input type="checkbox"/>	Image Name	Type	Status	Public	Protected	Format	Actions
<input type="checkbox"/>	vm-01-snap29122014	Snapshot	Active	No	No	QCOW2	Launch More
<input type="checkbox"/>	vm-01-snap30012015	Snapshot	Active	No	No	QCOW2	Launch More
Displaying 2 items							

Al momento della creazione della nuova macchina virtuale, l'utente può eventualmente decidere anche di cambiare il flavor dell'istanza qualora si presentasse per esempio l'esigenza di maggiori risorse (cpu, ram).



Launch Instance

Details *

Access & Security *

Networking *

Post-Creation

Advanced Options

Availability Zone:

nova

Instance Name: *

restored-vm-01

Flavor: *

medium

Instance Count: *

1

Instance Boot Source: *

Boot from snapshot

Instance Snapshot:

vm-01-snap30012015

Specify the details for launching an instance.

The chart below shows the resources used by this project in relation to the project's quotas.

Flavor Details

Name	medium
VCPUs	2
Root Disk	20 GB
Ephemeral Disk	0 GB
Total Disk	20 GB
RAM	4,096 MB

Project Limits

Number of Instances

2 of 100 Used

Number of VCPUs

3 of 200 Used

Total RAM

5,120 of 512,000 MB Used

Cancel

Launch

5.2 Ripristino del volume

Sia nel caso di backup manuale sia in quello di backup automatico, l'utente può facilmente ripristinare lo stato del volume creandone uno nuovo a partire dagli snapshot visualizzati nella pagina "Volumes & Snapshots" del proprio progetto.

Cliccando sul bottone "Create Voume" è possibile creare un nuovo volume a partire dallo snapshot scelto.



Volumes & Snapshots

Volumes **Volume Snapshots**

Volume Snapshots Delete Volume Snapshots

<input type="checkbox"/>	Name	Description	Size	Status	Volume Name	Actions
<input type="checkbox"/>	vol-01-snap30012015		1GB	Available	vol-01	Create Volume More ▾

Displaying 1 item

Al momento della creazione del volume, l'utente può eventualmente richiedere anche il "resize" del volume specificando la nuova dimensione (maggiore di quella iniziale dello snapshot).

Create Volume

Volume Name: *

vol-01-snap30012015-restored

Description:

Type:

encrypted-data

Size (GB):

4

Use snapshot as a source:

vol-01-snap30012015 (1GB)

Description:

Volumes are block devices that can be attached to instances.

Volume Limits

Total Gigabytes (7 GB)

1,000 <django.utils.functional.__proxy__ object at 0x7f79b0151890> Available

Number of Volumes (3)

10 Available

Volume size must be equal to or greater than the snapshot size (1GB)

Cancel

Create Volume

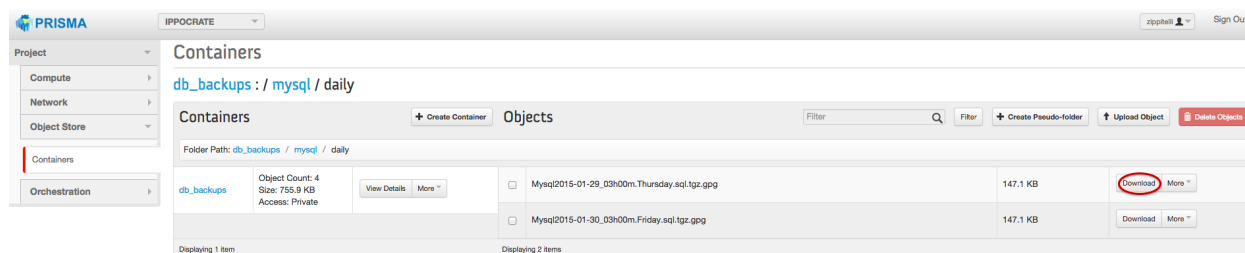
5.3 Ripristino del database

I backup dei database possono essere recuperati accedendo all'Object Storage dell'infrastruttura PRISMA-iaaS.

Attraverso la dashboard, dal menù Object Store>Containers, cliccando sul container "db_backups" sarà possibile navigare in una struttura simile a quella mostrata nella figura seguente.



Le foglie dell'albero sono i backup criptati del database ed hanno estensione “.gpg”.



L'utente può scaricare il backup scelto da ripristinare cliccando sul bottone “Download”.

L'archivio è criptato e va decodificato e scompattato prima di poter essere ripristinato.

Per decodificare il file .gpg è necessario assicurarsi che sull'host su cui è stato salvato l'archivio cifrato ci sia la chiave privata della coppia di chiavi usata per la cifratura ed in caso negativo importarla come descritto nel paragrafo **3.1 - Gestione delle chiavi per la cifratura del backup**.

Per la decodifica usare il seguente comando:

```
# gpg --output nomefile --decrypt nomefile.gpg
```

Con l'opzione --output il contenuto decriptato dell'archivio viene salvato nel file specificato come argomento dell'opzione.

Per esempio, se l'archivio che è stato scaricato dall'Object Store ha filename “Mysql2015-01-29_03h00m.Thursday.sql.tgz.gpg” procedere alla decodifica usando il seguente comando:



```
# gpg --output Mysql2015-01-29_03h00m.Thursday.sql.tgz --decrypt  
Mysql2015-01-29_03h00m.Thursday.sql.tgz.gpg
```

Una volta decodificato l'archivio è possibile estrarre il file di backup del database. Per esempio, nel caso del file precedente, trattandosi di un tar.gz è possibile usare il seguente comando:

```
# tar xvfz Mysql2015-01-29_03h00m.Thursday.sql.tgz
```

A questo punto il backup del database può essere ripristinato seguendo la procedura descritta sotto in base al tipo di database, MongoDB o MySQL.

5.3.1 MongoDB

Per ripristinare una particolare versione del database mongodb, bisogna prima estrarre la cartella interessata dall'archivio. Questa contiene una sottocartella per ogni database esistente, e ciascuna di queste sottocartelle contiene un file con estensione *bson* e uno con estensione *json* per ogni collection che fa parte di quel database.

Per importare tutta la cartella di backup, editare in una shell del terminale

```
$ mongorestore percorso_cartella/nome_cartella
```

Per caricare una particolare collection (*nome_collection*) di un particolare database (*nome_db*), editare

```
$ mongorestore --collection nome_collection --db nome_db  
percorso_file/nome_collection.bson
```

5.3.2 MySQL

Da un archivio di backup si estrae il file .sql che contiene il database di mysql salvato.

Per importarlo, si può usare questo comando in una shell del terminale

```
$ mysql -u root -p < percorso_file/nome_file.sql
```

5.4 Considerazioni finali

Lo script di backup creato consente di effettuare il backup giornaliero dei database mongodb e mysql, garantendo le seguenti caratteristiche:



- possibilità di avere sempre uno storico, almeno mensile, di tutto il database;
- ottimizzazione dello spazio disco occupato, grazie alla rotazione dei file e alla compressione dei file creati;
- versatilità, poiché si presta facilmente a implementare il backup di altri database di diverso tipo.