README.md

rds-backup.py

RDS backup (rds_backup.py) is a **Python 3.6** AWS lambda function to take rds snapshot and set tags with retention and deletion info.

Setup RDS backup AWS lambda function.

Create a rds-backup role

You must create a role with sufficient privileges on RDS to ensure the lambda function execution.
Be cautious and do not assign permissions that are not necessary.

Add new tags on RDS instances

It is necessary to add at least one new tag to each RDS to backup, the only required tag is “backup”, the tags “retentionsdays” and “deletesnapshot” are not necessary.

```
backups: True
deletesnapshot: False
retentiondays: 123
```

The **backup** and, **deletesnapshot** tags can take the values "**True**" or

"**False**", as indicated here (**literally**)

The **retentiondays** tag must have an integer value greater than one. The objective of this tag is to indicate the number of days the snapshot should be retained.

Create lambda function

1. Browse to AWS services Lambda and click on **Create fuction** button.
2. Add function name
3. Select Python 3.6 on Runtime list.
4. Select **Choose an existing role** on Role*.
5. On **Existing role*** choose the previously created role
6. Click on **Create function** botton to continue.
7. Remove the example code from the edition window and paste the python function code and click on **Save** botton.
8. Move down to the page and in the **Basic Settings** box add the **description** and set the **Time out** in 1 minute zero seconds and, click **Save** botton again.

Add trigger to lambda function.

1. Scroll down on **Add triggers** choices and click on **CloudWatch Events**.
2. Scroll down to the bottom page and click on **Rule** choices list and select **Create new rule**.
3. Add Rule name and, rule description.
4. Choose the **Schedule expression** option button.

5. Enter a **Schedule expression***. Enter a **Schedulle expression** that fits with the execution needs. You can find a good examples on Schedule Expressions Using Rate or Cron
6. Click on **Adde** botton.
7. Click 0n **Save** botton

Now your lambda function works! it will be ready to take action or if you prefer, you can add a test and validate its operation.