rENIAC Data Engine

Aug 13, 2021

Version: 2021-rsds-master-docv2

Contact

rENIAC Support - for technical help or discounts on pricing
support@reniac.com

Thomas Jorgensen, VP of Customer Success
thomas@reniac.com

Table of Contents

Document Revision History

Version	Date	Changes
2021-rsds-master-docv1	Aug 13, 2021	- Support for Amazon Keyspaces
2021-rsds-master-docv1	Jan 7, 2021	- N/A

Overview

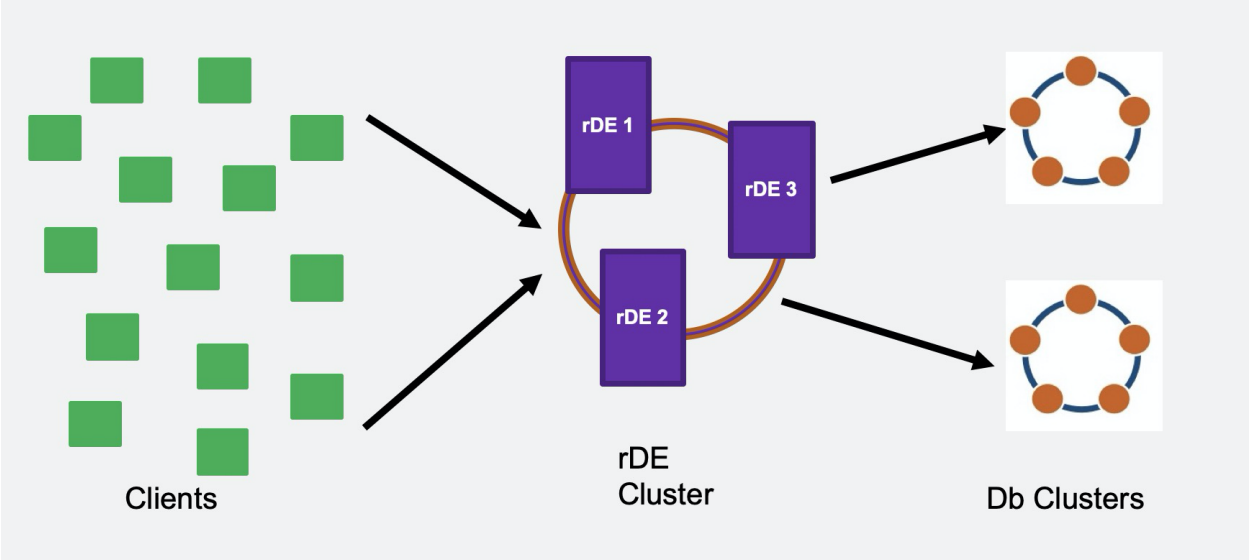
rENIAC Data Engine (rDE) sits between the Cassandra client and Cassandra database nodes or Amazon Keyspaces. It caches data in local storage and responds to queries by serving data either from its local storage or fetching it from the backend database when the data does not exist in the local storage.

rDE is implemented entirely in C++ using libraries optimized for fast data access. It provides consistently low latency and higher throughput as it is not subject to delays due to garbage collection and compaction.

rDE is designed to work in on-premise and cloud deployments like GCP, Azure and AWS.

rDE Architecture

rENIAC Data Engine has been designed as a managed cache to work without requiring any changes to the client code or the database, and with minimal configuration (as explained in the next section).



The rDE nodes are listening for incoming queries on the configured port. For read queries, rDE parses the query and looks for the data in the local storage. If found, it returns the result to the client. If not found, it obtains the data from the database cluster, stores a copy in the local storage and returns the result to the client.

For insert, update and delete operations, rDE forwards the query to the database cluster, invalidating the data stored in its own cache. When the database has successfully processed the query, rDE forwards the response to the client.

rDE generates metrics and log messages that are useful in understanding its performance. rDE sends metrics and logs to a metrics service which is responsible for displaying in a console window, in a browser and also for storing the data in a database.

AWS Installation

> ▲ **Note 1:**
> You can deploy rDE as a single node or as a cluster of nodes fronting your database cluster. The recommended deployment model is deploying the solution as a cluster. When working with a cluster of nodes, install the rpm on all the servers that will be a part of the cluster. Identify one node to work as a controller node (purely from a configuration standpoint). It will be used to generate configuration files and sync to the other rENIAC nodes in the cluster. In terms of functionality, it is equivalent to other nodes. If using a single node, that node is also the controller.

> ▲ **Note 2:**
> rENIAC requires password authentication to be enabled for ssh access. Please refer to Appendix D for more details.

> ▲ **Note 3:**
> Ensure that you can access the Cassandra databases from the rDE servers.

Follow these steps to configure rDE and its required components.

1. Make changes to the following files as explained in the Configuration section:

File name and location	Description
/opt/reniac/swp/conf/rde.ini	rENIAC Data Engine configuration. See the Configuration section below for details.

2. Create the cache folder (specified in the swproxy.yaml) and make sure rDE has write permissions to this folder.
   rENIAC recommends the use of NVMe storage for caching the data. Please make sure the NVMe drive is mounted correctly and the cache folder is on this mounted drive.

3. Configure the rDE cluster with these commands (and follow the instructions on the screen):
   ```
   cd /opt/reniac/swp/bin
   ./rDE_deploy -s
   ```

   The above step enables passwordless access to the remote hosts as well as for the controller node.

   To check that the configuration succeeded, run
   ```
   ./rDE_deploy -c
   ```

4. Run rDE as follows:
   ```
   cd /opt/reniac/swp/bin
   ./rDE_deploy --load
   ```

   Note: Before starting rDE cluster, you must make sure the Cassandra database is running on the nodes specified in the <cassandra-hosts> parameter in the configuration file (see below).

5. To stop the running services, issue the following:
   ```
   cd /opt/reniac/swp/bin
   ./rDE_deploy --terminate
   ```

   Note 1: The rDE_deploy script will start the required services.
   Note 2: The rENIAC Data Engine will accept queries only on the “host-ip-address” and “port”. You will need to point your applications to use this IP address and port.

   For example:
   ```
   ./cqlsh <Host IP address> 19393 --ssl
   ```

   Note that the ‘--ssl’ option is required when rDE is in front of Amazon Keyspaces. For the above command to work, the <CQLSHRC-DIR>/cqlshrc file should point to the public key file of rDE. Sample content of cqlshrc file is shown here:
   ```
   [ssl]
   version = TLSv1_2
   certfile =  /home/centos/Test_CLIENT_selfsigned.cer.pem
   ```

Cassandra-stress command (with Amazon Keyspaces)

```
./cassandra-stress write n=100000 -rate threads=16 -node <host IP address> -port native=19393 -transport ssl-protocol=TLSv1.2 truststore="{PATH to trustore dir}/cassandra.truststore" truststore-password=<password>
```

Cassandra-stress command (with a Cassandra db cluster)

```
./cassandra-stress write n=100000 -rate threads=16 -node <host IP address> -port native=19393
```

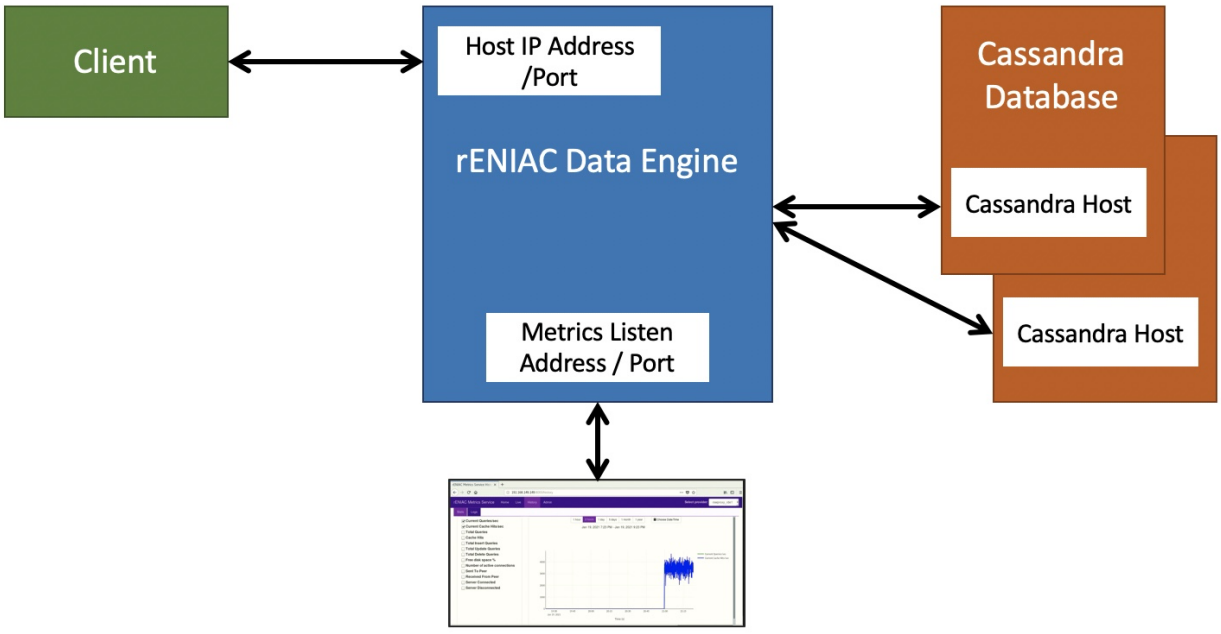
Configuration

Server configuration

The following configuration changes need to be made.

rENIAC Data Engine Configuration

The configuration file uses a hierarchical yaml file format familiar to Cassandra users.



Configuration parameter	Description

Environment	Information of rDE nodes
List of nodes	Name of the rDE nodes that will be part of the cluster. Example: list_of_nodes:rde1,rde2
dhcp	To enable DHCP (dhcp:1) and disable (dhcp:0)
<rDE node name>	
Host IP Address	The IP address of the host running the rENIAC Data Engine services. rENIAC recommends using a private IP address. Example: host-ip-address ：10.1.10.76
Host User	An existing, valid and active user (with sudo privileges) on the rDE server. Example: host-user: centos
Host Listen Port	This port is used for internal rDE communication. This setting should not be changed.
Software data dir	Local folder used by rDE to cache the data. The path must exist and have write permission. Example: software-data-dir : /home/centos/rde/cache
Cassandra Hosts	A comma-separated list of the IPs or hostnames of Cassandra hosts in the database cluster. Example: (for Amazon Keyspaces) cassandra-hosts: [cassandra.us-west-2.amazonaws.com] (for Cassandra db cluster) cassandra-hosts: [11.11.1.100, 11.11.1.101]
Cassandra Port	The port on which Cassandra service is listening. Example: cassandra-port: 9142
Cluster Name	Name of the Cassandra Cluster. Example: cluster-name: Amazon Keyspaces

Metrics Listen Address	IP address where the metrics service is running. rENIAC recommends using a private IP address. Example: metrics-listen-address: 10.1.6.11
Metrics Listen Port	Metrics listen port Example: metrics-listen-port : 8000
CQL Listen port	This is the port where the rDE listens for incoming queries. Example: host-listen-port : 19393
Cql SSL Enable	This is a boolean flag to enable/disable TLS encryption between client and rDE. Communication between rDE and Amazon Keyspaces is always TLS encrypted. Example: cql-ssl : true
Amazon Keyspaces Public Key	This is the public key provided by Amazon Keyspaces to access the service. It should be in openssl format. Example: cassandra-public-key : /opt/reniac/reniac_ThirdPartyTools/certificates/awsks.pem
Cql Listen Public Key	This is the rDE's public key provided with the package by default. It can be replaced with a user-generated public key in openssl format. This field can be blank if CQL traffic is not encrypted (see above). Example: cqllisten-public-key : /opt/reniac/reniac_ThirdPartyTools/certificates/Test_CLIENT_selfsigned.cer.pem
Cql Listen Private Key	This is the rDE's private key provided with the installation by default. It can be replaced with a user-generated private key in openssl format. This field can be blank if CQL traffic is not encrypted (see above).

	Example: cqllisten-private-key : /opt/reniac/reniac_ThirdPartyTools/certificates/Test_CLIENT_selfsigned.key.pem
Amazon Keyspaces Username	Username to access the Amazon Keyspaces service. This field can be blank if CQL traffic is not encrypted (see above). Example: aws-keyspaces-username : <username>
Amazon Keyspaces Password	Password to access the Amazon Keyspaces service. This field can be blank if CQL traffic is not encrypted (see above). Example: aws-keyspaces-password : <password>
Garbage Collection Time	Time at which the rENIAC service performs maintenance tasks in the background Example: garbage-collection : 3:00:00
Page Size	The block size (in KB) used by rDE for storing cache data. If your partitions are large, use a larger page size. This will reduce the number of disk reads and reduce the latency. For smaller partitions, use a page size of 4KB or less. If you change this setting, you will have to delete the cache contents (specified in the software-data-dir setting). Possible values are 4, 8, 16, 32, 64, 128. Example: page-size: 4

▲ Note

The following ports need to be opened up on the rDE servers:

- Host listen port (default: 18383)
- CQL listen port (default: 19393)
- Metrics listen port (default: 8000)

Client configuration

The client must be changed to send the queries to rDE's <host-ip-address> and <host-listen-port>. No other changes are required on the client side.

Metrics Service

The Metrics Console is a web UI for viewing the metrics and logs provided by the rDE nodes. The Metrics Console has a live view and a historic view.

To open the Metrics Console, point your web browser to:
http://<Metrics service IP address>:8000

This is the IP address you specified in the configuration file (see above). When you point your browser here, it opens the Live page of the Metrics Console.

These are some of the metrics rENIAC Data engine reports.

Queries
Current Queries/sec
Current Cache Hits/sec
Total Queries
Cache Hits
Total Insert Queries
Total Update Queries
Total Delete Queries

Workloads

We expect the data engine to provide the best performance improvement (over Cassandra) with read-heavy workloads (80% read or higher).

Known Limitations

- Version
    - rENIAC Data Engine has been tested extensively with Cassandra db version 3.11.4 and DSE version 5.1.17
    - The data engine requires the Cassandra client and database to support Cassandra native protocol version 4 or version 5
- Known issues
    - Authentication/authorization is not supported
    - Wire encryption is not supported
    - Database clusters with nodes spread across multiple data centers or cloud provider zones are not supported
    - PAGING is not supported
    - Queries on tables with UDT are not optimized
    - BATCH statements are not optimized
    - IN queries are only supported for partition key columns and not for cluster key columns
    - Indexes are not optimized
    - Updating column types is not supported
    - The internode traffic between peer rDE nodes is not encrypted.
    - Amazon Keyspaces does not support TTL or WRITETIME, like Cassandra does. If TTL is used in a select query for a partition key which is in rDE's cache, an empty row will be returned. In case of a non-cached partition key, the Amazon Keyspaces provided error string "TTL is not yet supported" is returned.

Customer Support

In case of any problem in using the rENIAC Data Engine, you can contact rENIAC support using the following email address or our Help Center. Please mention the product version in the communication.

Email: support@reniac.com
rENIAC Help Center: https://reniac.atlassian.net/servicedesk/customer/portal/1

Appendix A: Licensing

If you are using rENIAC as an AMI through AWS Marketplace, the trial period will automatically convert to a paid subscription. For discounted pricing, please contact us at hello@reniac.com.

Appendix B: Sample INI File

```
[Environment]
# mention here the rDE names you want to start the setup.
list_of_nodes:rde1
dhcp:0

[rde1]
### RDE Host details - ip address
host-ip-address : 11.11.1.2

# An existing, valid and active user name on the machine with sudo access
host-user : centos

host-listen-port    : 18383

# software data directory -  The path should be existing and it should have the write permission
software-data-dir       : /home/centos/perf_demo/cache

# Lists of the IPs or hostnames of cassandra hosts from the cluster
cassandra-hosts: [ 11.11.1.4 ]

# Port on which cassandra hosts are listening
cassandra-port : 9042

# Name of the Cassandra Cluster
cluster-name: Amazon Keyspaces

### Metrics Services Details
# ip of the machine where we are running  the metrics services
metrics-listen-address  : 10.1.8.16

#Enable/disable tls
cql-ssl : true

#Amazon Keyspaces public key. Can be empty if cql-ssl is false
cassandra-public-key: /opt/reniac/reniac_ThirdPartyTools/certificates/awsks.pem

#Public key of rDE. Can be empty if cql-ssl is false
cqllisten-public-key:
/opt/reniac/reniac_ThirdPartyTools/certificates/Test_CLIENT_selfsigned.cer.pem
```

#Private key of rDE. Can be empty if cql-ssl is false
cqllisten-private-key:
/opt/reniac/reniac_ThirdPartyTools/certificates/Test_CLIENT_selfsigned.key.pem

#Username to access Amazon Keyspaces service. Can be empty if cql-ssl is false
aws-keyspaces-username: <username>

#Password to access Amazon Keyspaces service. Can be empty if cql-ssl is false
aws-keyspaces-password: <password>

Port number of Metrics Service
metrics-listen-port : 8000

cqllisten-port : 19393

garbage-collection : 3:00:00

page-size: 4

Appendix C - Software Dependencies

Required software not installed by rENIAC installer

- Python 2.7

The following Python modules are required for proper operation of the Metrics Console. They are installed as part of the installation process.

- flask-socketio
- socketIO-client
- eventlet
- beautifultable
- coloredlogs
- prettytable
- flask
- Flask-SQLAlchemy
- configparser
- sshpass

rENIAC appreciates and acknowledges the use of third-party software in the rENIAC Data Engine. Please see the accompanying file titled “Third Party Software” for details.

Appendix D - Authentication

rENIAC Data Engine is designed to work in a distributed environment, with coordination happening between instances installed across multiple nodes in a clustered setup. So the installation and configuration steps need to be done as a user with elevated privileges and using password authentication.

Password Authentication

rDE requires password authentication to be enabled for ssh access. Follow these steps to enable password authentication on all the rDE servers.

- On the rDE host, edit the sshd_config file
  `sudo vi /etc/ssh/sshd_config`

- Search for 'PasswordAuthentication' in this file (around line 63). Uncomment this entry by removing the '#' at the beginning of the line. Change 'no' to 'yes'. Make sure that there are no other 'PasswordAuthentication' entries in this file that might override this change. Save and close this file.

- Restart the SSH daemon for the above change to take effect
  `sudo systemctl restart sshd`

- After enabling password authentication, set the password (if not done already) using the following command
  `sudo passwd <<username>>`

Elevated Privileges

If you do not use a default user (ec2-user, centos) account, please ensure that the user account is part of the operating system's wheel group and is able to run sudo commands without being prompted for a password.

- To add the user to the wheel group, run the following command
  `sudo usermod -aG wheel <<username>>`

- To enable passwordless sudo, do the following
  - Run the following command
    `sudo visudo`

  - Uncomment the following line (by removing the '#' at the beginning of the line)
    %wheel ALL=(ALL) NOPASSWD: ALL