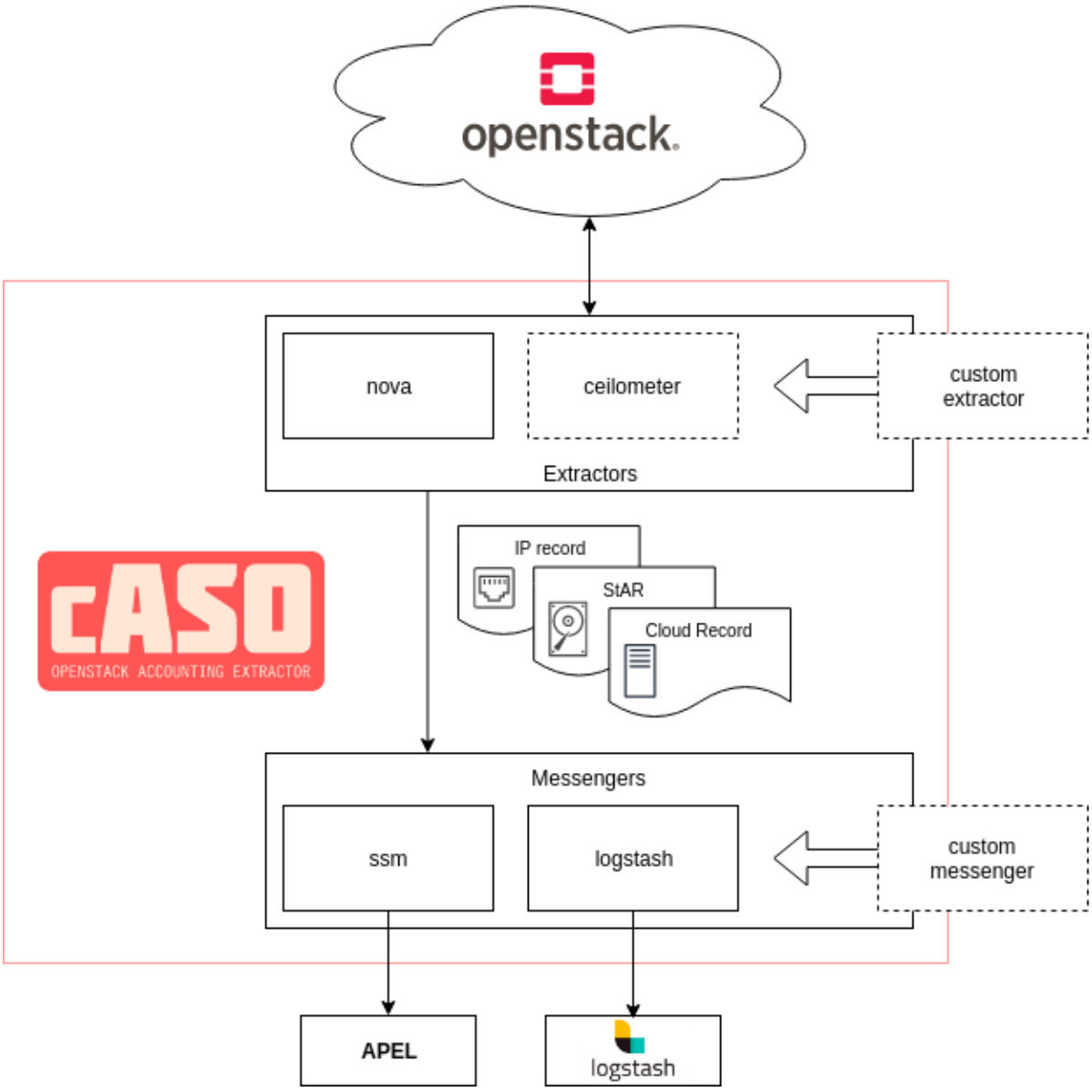
caso Documentation

Spanish National Research Council (CSIC)

Jun 07, 2023

CONTENTS

cASO is an accounting reporter (currently supports Cloud Accounting Usage Records) for OpenStack deployments. cASO gets usage information from OpenStack public APIs (no access to DB is required) and can generate valid output for Apel SSM or logstash.



Contents:

CHAPTER

ONE

CASO RELEASE NOTES

1.1 Current Release Notes

1.1.1 4.1.1-17

Bug Fixes

- Fixes issue 115, where timestamps where incorrecly rendered. This fix requires a republish of records.

1.1.2 4.1.1

Upgrade Notes

- The `reader` role with a system scope of `all` must be granted to the `cASO` user, please check the documentation.

Bug Fixes

- – *#108 <https://github.com/IFCA/caso/issues/108>*. Failure getting accounting data when using different domains.

1.1.3 4.1.0

New Features

- Allow loading VO mapping directly from Keystone. Instead of mainitaining a separate file, we now allow users to load the VO directly from a configurable project property.
- Allow to use Keystone project tags to get usages from projects.

Upgrade Notes

- Now cASO defaults to all the installed default extractors, that are the OpenStack Compute (nova), Volume (cinder) and Network (neutron), publishing Cloud, Storage and IP records.
- It is reccomended that you stop using the Keystone voms mapping file as soon as possible, please migrate your old JSON mapping to the new method using Keystone project's properties. In order to do so, you can use the `caso-mapping-migrate` command line option, that will provide you with the correct commands to run to perform the migration.
- If using tags to gather the list of projects you need to ensure that the correct roles and system scopes are applied. Please ensure that the correct roles are defined, as indicated in the configuration documentation.

Deprecation Notes

- The VO mapping file is now deprecated, in favour of project properties.

1.1.4 4.0.0

New Features

- Allow to load more than one extractor in the configuration, making possible to get more than one type of record.
- Add support for the EMI STaR records for storage, and support extracting records from OpenStack Cinder.

1.1.5 3.0.0

New Features

- Add support for GPU accounting, using the GPU 0.1 record agreed with APEL.

Upgrade Notes

- Please ensure that you have the correct configuration in the policy files, as a new policy rule must be modified. The accounting user does not need to have access to the "identity:list_users" action, but to the "identity:get_user" action instead.

Deprecation Notes

- All the *benchmark_** and *accelerator_* option definitions in the *[DEFAULT]* section of the configuration file have been marked as deprecated, and are now included in the individual *[benchmark]* and *[acelerator]* sections, with the corresponding prefix (i.e. *benchmark_* and *accelerator_*) removed. Check the sample configuration file for more details.

Bug Fixes

- Fix an issue when getting the usernames, that caused configuration errors to be unnoticed.

Other Notes

- Keystone versions from Ussuri onwards (>= 17.0.0) implement a new policy. Please check the documentation so as to ensure that you are applying the correct changes.

1.1.6 2.1.0

Prelude

This version includes a refactoring of the base extractors, dropping support for the ceilometer extractor that was unmaintained for a long period of time.

Upgrade Notes

- Ceilometer extractor is no longer supported.

1.1.7 2.0.0

Prelude

Starting with this version cASO release notes are published within the documentation. This version is a major release that implements IP accounting record, as well as several bugfixes. There are no upgrade notes to take into account.

New Features

- Add multi-region support in order to extract information from several regions through different configuration files.
- New IP accounting record is implemented. Now cASO is able to extract IP accounting and publish it using its JSON rendering. No new configuration needs to be done, but the cASO user needs to have access to the Neutron endpoints.
- cASO now allows to specity the projects to extract records from as project IDs, rather than names. When dealing with different identity domains this is troublesome, therefore we need to allow users to specify project IDs rather than names.

Bug Fixes

- Define the correct entrypoints for the V2 and V4 messengers.
- Generate LOG warnings when mappings cannot be found.

CHAPTER

TWO

INSTALLATION

2.1 Pre-requisites

If you are planning to use `cASO` for generating accounting records for EGI, you will need a valid APEL/SSM configuration. Follow the documentation available at the EGI.eu Federated Cloud documentation in order to set it up.

2.2 Installation

The best way to install cASO and have the most up to date version is using the repositories and packages provided in the EGI AppDB:

https://appdb.egi.eu/store/software/caso

2.2.1 Manual installation

Even the reccomended method is to use a package from the EGI AppDB, it is possible to install it from the Python Pacakge Index as follows:

```
$ pip install caso
```

Or you can install it on a virtualenv:

```
$ virtualenv --python python3 caso
$ source caso/bin/activate
(caso) $ pip install caso
```

CHAPTER

THREE

CONFIGURATION

3.1 OpenStack Configuration

Apart from configuring cASO, several actions need to be performed in your OpenStack installation in order to be able to extract accounting records.

3.1.1 User credentials (required)

In the next section you will configure an OpenStack Keystone credentials in order to extract the records. The cASO user has to be a member of each of the projects (another option is to convert that user in an administrator, but the former option is a safer approach) for which it is extracting the accounting, with the `reader` role (this is a default OpenStack Keystone role). Otherwise, `cASO` will not be able to get the usages and will fail:

```
openstack user create --password <password> accounting
# For each of the projects, add the user with the accounting role
openstack role add --user accounting --project <project> reader
```

Moreover, you need to grant the user the role reader with a system scope of `all` in order to get all the project tags, as well as the other user's information:

```
openstack role add --system all --user accounting reader
```

3.1.2 Policy modifications

Important: No policy modifications are needed

The following policy modifications are just shown here for reference, if you wish to use a different role. You do not need to use them.

If you use the role `reader` as configured above, you do not need to configure anything else in the policy. However, if you wish to use a different role mapping, the accounting user needs access to Keystone so as to extract the users information. Depending on your configuration, you need to modify the JSON policy file (`/etc/keystone/policy.json`) or the YAML policy file (`/etc/keystone/policy-yaml`). The modifications in the policy depend on the Keystone version, please ensure that you are applying the correct changes as listed in the following table. In the example show, we are using a dedicated role `accounting`.

OpenStack Version	Policy contents	
From Stein (>= 15.0.0)	Original	"identity:get_user": "(role:reader and system_scope:all) or (role:reader and token.domain.id:%(target.user.domain_id)s) or user_id:%(target.user.id)s"
	Modified	"identity:get_user": "(role:reader and system_scope:all) or (role:reader and token.domain.id:%(target.user.domain_id)s) or user_id:%(target.user.id)s or role:accounting"
Up to Rocky (<= 14.0.0)	Original	"identity:get_user": "rule:admin_or_owner"
	Modified	"identity:get_user": "rule:admin_or_owner or role:accounting"

3.2 Selecting projects to get usages

cASO will extract project usages for those projects that have been explictly marked by the operator by either of the ways explained below. The final project list will result from the merge of both methods, so thay are not mutually exclusive.

- Tagging the project with the configured `caso_tag` in OpenStack Keystone. By default this option is set to `caso`, so in order to mark a project to get extracted you should use the following command for each of the projects:

  ```
  openstack project set --tag caso <project id>
  ```

  You can check the list of projects to get usages by using:

  ```
  openstack project list --tags caso
  ```

- Using the `projects` list in the `[DEFAULT]` section of your configuration file (see below).

3.2.1 Setting VO mapping

In order to publish correct accounting records, cASO needs to know the VO that should be used to publish the records from a given project. In order to do so, you need to specify the correct mapping in each of the projects properties. The name of the property that will be used is defined in the `vo_property` configuration option, and defaults to `accounting:VO`, therefore you can configure it as follows:

```
openstack project set --property accounting:VO=<VO FQAN> <project id>
```

3.3 cASO configuration

cASO uses a config file (default at /etc/caso/caso.conf) with several sections. A sample file is available at etc/caso/caso.conf.sample.

3.3.1 [DEFAULT] section

The [DEFAULT] section configures the basic behavior of cASO. The sample config file (/etc/caso/caso.conf.sample) includes a description of every option. You should check at least the following options:

- extractor (default value: nova), specifies which extractor to use for getting the data. The following APIs are supported: ceilomenter and nova. Both should generate equivalent information.
- site_name (default value: <None>). Name of the site as defined in GOCDB.
- service_name (default value: $site_name). Name of the service within a site. This is used if you have several endpoints within your site.
- projects (list value, default empty). List of the projects to extract records from. You can use either the project ID or the project name. We recommend that you use the project ID, especially if you are using domain-based authentication, as otherwise gathering the information might fail. This option, and the usage of caso_tag below will set up the final project list.
- caso_tag (default value: caso), specified the tag to be used filter projects to extract their usage. The projects that are listed with this tag, as well as the projects list set above will set up the final project list. If you only use tags, and want to remove a project from being published, you just need to remove the tag from the project.
- messengers (list, default: noop). List of the messengers to publish data to. Records will be pushed to all these messengers, in order. Valid messengers shipped with cASO are:
  - ssm for publishing APEL records.
  - logstash for publishing to Logstash.
  - noop do nothing at all.

  Note that there might be other messengers available in the system if they are registered into the caso.messenger entry point namespace. Please also note that versioning of the SSM messenger is deprecated.
- vo_property (default: accounting:VO). The project that will be set in the OpenStack Keystone project to map a given project to a specific VO.
- **DEPRECATED** mapping_file (default: /etc/caso/voms.json). File containing the mapping from VOs to local projects as configured in Keystone-VOMS, in the following form:

```
{
    "VO": {
        "projects": ["foo", "bar"],
    }
}
```

  Note that you have to use either the project ID or project name for the mapping, as configured in the projects configuration variable.

3.3.2 [keystone_auth] section

This section is used to specify the authentication credentials to be used to connect to the OpenStack APIs. cASO leverages the OpenStack keystoneauth library for authentication, so that it is possible to use any authentication plugin that is available there (so starting on version 1.0 of cASO it is possible to use the Keystone V3 API).

Important: You need to specify the `auth_type` that you want to use (normally `v3password` is a good choice.

For an exhaustive list of available plugins please refer to the keystoneauth documentation.

3.3.3 [ssm] section

Options defined here configure the SSM messenger. There is only one option at the moment:

- `output_path` (default: `/var/spool/apel/outgoing/openstack`), directory to put the generated SSM records. APEL/SSM should be configured to take records from that directory.

3.3.4 [logstash] section

Options defined here configure the logstash messenger. Available options:

- `host` (default: `localhost`), host of Logstash server.
- `port` (default: `5000`), Logstash server port.

3.3.5 Other cASO configuration options

For an exhaustive list of the defined options, please check the following page:

cASO configuration file

caso: DEFAULT

`messengers`

> **Type**
> list
>
> **Default**
> `['noop']`

List of messengers that will dispatch records. valid values are logstash,noop,ssm,ssmv4. You can specify more than one messenger.

`spooldir`

> **Type**
> string
>
> **Default**
> `/var/spool/caso`

Spool directory.

`lock_path`

Type
string

Default
`$spooldir`

Directory to use for lock files. For security, the specified directory should only be writable by the user running the processes that need locking. Defaults to environment variable CASO_LOCK_PATH or $spooldir

`dry_run`

Type
boolean

Default
`False`

Extract records but do not push records to SSM. This will not update the last run date.

`site_name`

Type
string

Default
`<None>`

Site name as in GOCDB.

`service_name`

Type
string

Default
`$site_name`

Service name within the site

`projects`

Type
list

Default
`[]`

List of projects to extract accounting records from. You can use this option, or add 'caso' tag to the project in Keystone. Please refer to the documentation for more details.

`caso_tag`

Type
string

Default
`caso`

Tag used to mark a project in Keystone to be extracted by cASO

`vo_property`

Type
string

Default
`VO`

Property key used to get the VO name from the project properties.

`mapping_file`

Type
string

Default
`/etc/caso/voms.json`

File containing the VO <-> project mapping as used in Keystone-VOMS.

Table 1: Deprecated Variations

Group	Name
extractor	mapping_file

Warning: This option is deprecated for removal. Its value may be silently ignored in the future.

Reason
This option is marked for removal in the next release. Please see the release notes, and migrate your current configuration to use the new project mapping as soon as possible. If you already migrated your configuration, please remove the JSON file to get rid of this message.

`extract_to`

Type
string

Default
<None>

Extract record changes until this date. If it is not set, we use now. If a server has ended after this date, it will be included, but the consuption reported will end on this date. If no time zone is specified, UTC will be used.

`extract_from`

Type
string

Default
<None>

Extract records that have changed after this date. This means that if a record has started before this date, and it has changed after this date (i.e. it is still running or it has ended) it will be reported. If it is not set, extract records from last run. If it is set to None and last run file is not present, it will extract records from the beginning of time. If no time zone is specified, UTC will be used.

`extractor`

Type
list

Default
`['nova', 'cinder', 'neutron']`

Which extractor to use for getting the data. If you do not specify anything, nova will be used. Available choices are ['cinder', 'neutron', 'nova']

caso: accelerator

`type_key`

Type
string

Default
`Accelerator:Type`

Metadata key used to retrieve the accelerator type from the flavor properties.

`vendor_key`

Type
string

Default
`Accelerator:Vendor`

Metadata key used to retrieve the accelerator vendor from the flavor properties.

`model_key`

Type
string

Default
`Accelerator:Model`

Metadata key used to retrieve the accelerator model from the flavor properties.

`number_key`

Type
string

Default
`Accelerator:Number`

Metadata key used to retrieve the accelerator number from the flavor properties.

caso: benchmark

name_key

Type
string

Default
`accounting:benchmark_type`

Metadata key used to retrieve the benchmark type from the flavor properties.

value_key

Type
string

Default
`accounting:benchmark_value`

Metadata key used to retrieve the benchmark value from the flavor properties.

caso: keystone_auth

auth_type

Type
unknown type

Default
<None>

Authentication type to load

Table 2: Deprecated Variations

Group	Name
keystone_auth	auth_plugin

auth_section

Type
unknown type

Default
<None>

Config Section from which to load plugin specific options

cafile

Type
string

Default
<None>

PEM encoded Certificate Authority to use when verifying HTTPs connections.

`certfile`

Type
string

Default
<None>

PEM encoded client certificate cert file

`keyfile`

Type
string

Default
<None>

PEM encoded client certificate key file

`insecure`

Type
boolean

Default
False

Verify HTTPS connections.

`timeout`

Type
integer

Default
<None>

Timeout value for http requests

`collect_timing`

Type
boolean

Default
False

Collect per-API call timing information.

`split_loggers`

Type
boolean

Default
False

Log requests to multiple loggers.

`auth_url`

Type
unknown type

Default
<None>

Authentication URL

`system_scope`

Type
unknown type

Default
<None>

Scope for system operations

`domain_id`

Type
unknown type

Default
<None>

Domain ID to scope to

`domain_name`

Type
unknown type

Default
<None>

Domain name to scope to

`project_id`

Type
unknown type

Default
<None>

Project ID to scope to

Table 3: Deprecated Variations

Group	Name
keystone_auth	tenant-id
keystone_auth	tenant_id

`project_name`

Type
unknown type

Default
<None>

Project name to scope to

Table 4: Deprecated Variations

Group	Name
keystone_auth	tenant-name
keystone_auth	tenant_name

`project_domain_id`

Type
unknown type

Default
<None>

Domain ID containing project

`project_domain_name`

Type
unknown type

Default
<None>

Domain name containing project

`trust_id`

Type
unknown type

Default
<None>

ID of the trust to use as a trustee use

`default_domain_id`

Type
unknown type

Default
<None>

Optional domain ID to use with v3 and v2 parameters. It will be used for both the user and project domain in v3 and ignored in v2 authentication.

`default_domain_name`

Type
unknown type

Default
<None>

Optional domain name to use with v3 API and v2 parameters. It will be used for both the user and project domain in v3 and ignored in v2 authentication.

`user_id`

Type
unknown type

Default
<None>

User id

`username`

Type
unknown type

Default
<None>

Username

Table 5: Deprecated Variations

Group	Name
keystone_auth	user-name
keystone_auth	user_name

`user_domain_id`

Type
unknown type

Default
<None>

User's domain id

`user_domain_name`

Type
unknown type

Default
<None>

User's domain name

`password`

Type
unknown type

Default
<None>

User's password

caso: logstash

host

Type
string

Default
`localhost`

Logstash host to send records to.

port

Type
integer

Default
`5000`

Logstash server port.

caso: ssm

output_path

Type
string

Default
`/var/spool/apel/outgoing/openstack`

Directory to put the generated SSM records.

max_size

Type
integer

Default
`100`

Maximum number of records to send per message

oslo.config: DEFAULT

config_file

Type
list of filenames

Default
`['~/.project/project.conf', '~/project.conf', '/etc/project/project.conf', '/etc/project.conf']`

Path to a config file to use. Multiple config files can be specified, with values in later files taking precedence. Defaults to the value above. This option must be set from the command-line.

`config_dir`

Type
list of directory names

Default
`['~/.project/project.conf.d/', '~/project.conf.d/', '/etc/project/project.conf.d/', '/etc/project.conf.d/']`

Path to a config directory to pull **.conf* files from. This file set is sorted, so as to provide a predictable parse order if individual options are over-ridden. The set is parsed after the file(s) specified via previous –config-file, arguments hence over-ridden options in the directory take precedence. This option must be set from the command-line.

`config_source`

Type
list

Default
`[]`

Lists configuration groups that provide more details for accessing configuration settings from locations other than local files.

`driver`

Type
string

Default
`remote_file`

This option has a sample default set, which means that its actual default value may vary from the one documented above.

The name of the driver that can load this configuration source.

`uri`

Type
URI

Default
`https://example.com/my-configuration.ini`

This option has a sample default set, which means that its actual default value may vary from the one documented above.

Required option with the URI of the extra configuration file's location.

`ca_path`

Type
string

Default
`/etc/ca-certificates`

This option has a sample default set, which means that its actual default value may vary from the one documented above.

The path to a CA_BUNDLE file or directory with certificates of trusted CAs.

`client_cert`

Type
string

Default
`/etc/ca-certificates/service-client-keystore`

This option has a sample default set, which means that its actual default value may vary from the one documented above.

Client side certificate, as a single file path containing either the certificate only or the private key and the certificate.

`client_key`

Type
string

Default
`<None>`

Client side private key, in case client_cert is specified but does not includes the private key.

oslo.log: DEFAULT

`debug`

Type
boolean

Default
`False`

Mutable
This option can be changed without restarting.

If set to true, the logging level will be set to DEBUG instead of the default INFO level.

`log_config_append`

Type
string

Default
`<None>`

Mutable
This option can be changed without restarting.

The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation. Note that when logging configuration files are used then all logging configuration is set in the configuration file and other logging configuration options are ignored (for example, log-date-format).

Table 6: Deprecated Variations

Group	Name
DEFAULT	log-config
DEFAULT	log_config

`log_date_format`

Type
string

Default
`%Y-%m-%d %H:%M:%S`

Defines the format string for %(asctime)s in log records. Default: the value above . This option is ignored if log_config_append is set.

`log_file`

Type
string

Default
<None>

(Optional) Name of log file to send logging output to. If no default is set, logging will go to stderr as defined by use_stderr. This option is ignored if log_config_append is set.

Table 7: Deprecated Variations

Group	Name
DEFAULT	logfile

`log_dir`

Type
string

Default
<None>

(Optional) The base directory used for relative log_file paths. This option is ignored if log_config_append is set.

Table 8: Deprecated Variations

Group	Name
DEFAULT	logdir

`watch_log_file`

Type
boolean

Default
`False`

Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if log_file option is specified and Linux platform is used. This option is ignored if log_config_append is set.

`use_syslog`

Type
boolean

Default
`False`

Use syslog for logging. Existing syslog format is DEPRECATED and will be changed later to honor RFC5424. This option is ignored if log_config_append is set.

`use_journal`

Type
boolean

Default
`False`

Enable journald for logging. If running in a systemd environment you may wish to enable journal support. Doing so will use the journal native protocol which includes structured metadata in addition to log messages.This option is ignored if log_config_append is set.

`syslog_log_facility`

Type
string

Default
`LOG_USER`

Syslog facility to receive log lines. This option is ignored if log_config_append is set.

`use_json`

Type
boolean

Default
`False`

Use JSON formatting for logging. This option is ignored if log_config_append is set.

`use_stderr`

Type
boolean

Default
`False`

Log output to standard error. This option is ignored if log_config_append is set.

`use_eventlog`

Type
boolean

Default
`False`

Log output to Windows Event Log.

`log_rotate_interval`

Type
integer

Default
`1`

The amount of time before the log files are rotated. This option is ignored unless log_rotation_type is set to "interval".

`log_rotate_interval_type`

Type
string

Default
`days`

Valid Values
Seconds, Minutes, Hours, Days, Weekday, Midnight

Rotation interval type. The time of the last file change (or the time when the service was started) is used when scheduling the next rotation.

`max_logfile_count`

Type
integer

Default
`30`

Maximum number of rotated log files.

`max_logfile_size_mb`

Type
integer

Default
`200`

Log file maximum size in MB. This option is ignored if "log_rotation_type" is not set to "size".

`log_rotation_type`

Type
string

Default
none

Valid Values
interval, size, none

Log rotation type.

Possible values

interval
Rotate logs at predefined time intervals.

size
Rotate logs once they reach a predefined size.

none
Do not rotate log files.

`logging_context_format_string`

Type
string

Default

```
%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s
[%(global_request_id)s %(request_id)s %(user_identity)s]
%(instance)s%(message)s
```

Format string to use for log messages with context. Used by oslo_log.formatters.ContextFormatter

`logging_default_format_string`

Type
string

Default

```
%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [-]
%(instance)s%(message)s
```

Format string to use for log messages when context is undefined. Used by oslo_log.formatters.ContextFormatter

`logging_debug_format_suffix`

Type
string

Default

```
%(funcName)s %(pathname)s:%(lineno)d
```

Additional data to append to log message when logging level for the message is DEBUG. Used by oslo_log.formatters.ContextFormatter

`logging_exception_prefix`

Type
string

Default

```
%(asctime)s.%(msecs)03d %(process)d ERROR %(name)s %(instance)s
```

Prefix each line of exception output with this format. Used by oslo_log.formatters.ContextFormatter

`logging_user_identity_format`

Type
string

Default

```
%(user)s %(project)s %(domain)s %(system_scope)s %(user_domain)s
%(project_domain)s
```

Defines the format string for %(user_identity)s that is used in logging_context_format_string. Used by oslo_log.formatters.ContextFormatter

`default_log_levels`

Type
list

Default

```
['amqp=WARN', 'amqplib=WARN', 'boto=WARN', 'qpid=WARN',
'sqlalchemy=WARN', 'suds=INFO', 'oslo.messaging=INFO',
'oslo_messaging=INFO', 'iso8601=WARN', 'requests.packages.urllib3.
connectionpool=WARN', 'urllib3.connectionpool=WARN', 'websocket=WARN',
'requests.packages.urllib3.util.retry=WARN', 'urllib3.util.
retry=WARN', 'keystonemiddleware=WARN', 'routes.middleware=WARN',
```

```
'stevedore=WARN', 'taskflow=WARN', 'keystoneauth=WARN', 'oslo.
cache=INFO', 'oslo_policy=INFO', 'dogpile.core.dogpile=INFO']
```

List of package logging levels in logger=LEVEL pairs. This option is ignored if log_config_append is set.

publish_errors

Type
boolean

Default
`False`

Enables or disables publication of error events.

instance_format

Type
string

Default
`"[instance: %(uuid)s] "`

The format for an instance that is passed with the log message.

instance_uuid_format

Type
string

Default
`"[instance: %(uuid)s] "`

The format for an instance UUID that is passed with the log message.

rate_limit_interval

Type
integer

Default
`0`

Interval, number of seconds, of log rate limiting.

rate_limit_burst

Type
integer

Default
`0`

Maximum number of logged messages per rate_limit_interval.

rate_limit_except_level

Type
string

Default
`CRITICAL`

Log level name used by rate limiting: CRITICAL, ERROR, INFO, WARNING, DEBUG or empty string. Logs with level greater or equal to rate_limit_except_level are not filtered. An empty string means that all levels are filtered.

`fatal_deprecations`

Type
boolean

Default
`False`

Enables or disables fatal status of deprecations.

3.4 Additional (optional) configurations

3.4.1 Publishing benchmark information for OpenStack flavors (optional)

cASO is able to publish benchmark information included in the accounting recors, in order to do CPU normalization at the accounting level.

In order to do so, you need to add this information to the flavor properties and configure caso to retrieve this information. There are two different values that need to be added to the flavor

Table 9: Default flavor properties used by cASO to publish benchmark information

Property	Value
`accounting:benchmark_name`	Benchmark name (e.g. HEPSPEC06)
`accounting:benchmark_value`	Benchmark value (e.g. 99)

For example, if you are using HEPSPEC06 and the benchmark value is `99` for the flavor `m1.foo`, the benchmark information is configured as follows:

```
openstack flavor set --property accounting:benchmark_name="HEPSPEC06" --property␣
↪accounting:benchmark_value=99 m1.foo
```

Using different keys to specify bechmark information

If you do not want to use cASO's default flavor properties `accounting:benchmark_name` and `accounting:benchmark_value` (for example because you are using different benchmark types and values) you can specify which properties `cASO` should look for by using the `name_key` and `value_key` in the `[benchkmark]` section of the configuration file.

Important: Please note that there is an OpenStack scheduler filter that removes hosts based on flavor properties. In order to not interfere with the behaviour of this filter you must prefix the property with a `scope:` so that cASO's properties are not taken into account for this filtering. When adding these properties in cASO's configuration file, please include the complete name (i.e. `scope:property`).

Important: Option deprecation

Please bear in mind that the old options `benchmark_name_key` and `benchmark_value_key` in the `[DEFAULT]` configuration option are marked as deprecated. Please update your configuration file as soon as possible to avoid warnings.

3.4.2 Publishing accelerator information for OpenStack accelerators (optional)

Starting with cASO >= 3.0.0 it is possible to publish accelerator information using a new accounting record.

In order to do so, you need to add this information to the flavor properties and configure caso to retrieve this information. There are different values that need to be added to the flavor:

Table 10: Default flavor properties used by cASO to publish accelerator information

Flavor Property	Value
Accelerator:Type	The accelerator type (e.g. GPU))
Accelerator:Vendor	Name of the accelerator vendor (e.g. NVIDIA)
Accelerator:Model	Accelerator model (e.g. V100)
Accelerator:Number	Hoy many accelerators are available for that flavor (e.g. 2)

Using different keys to specify bechmark information

If you do not want to use cASO's default flavor properties to publish the existing accelerators, you can specify which properties `cASO` should look for by using the `type_key`, `vendor_key`, `model_key` and `number_key` in the `[acelerator]` section of the configuration file.

Important: Please note that there is an OpenStack scheduler filter that removes hosts based on flavor properties. In order to not interfere with the behaviour of this filter you must prefix the property with a `scope:` so that cASO's properties are not taken into account for this filtering. When adding these properties in cASO's configuration file, please include the complete name (i.e. `scope:property`).

CHAPTER

FOUR

CASO MULTI-REGION SUPPORT

- In case the monitored projects rely on a specific region, define the following variable in the /etc/caso/caso.conf

```
[DEFAULT]
region_name = <REGION>
```

- In case the monitored Project(s) rely on different regions, prepare different files /etc/caso/caso-<REGION>.conf

```
[DEFAULT]
region_name = <REGION>
```

- List the Project(s) in the /etc/caso/voms.json as from the documentation

```
{
  "Project1": {
    "projects": ["Project1"]
  },
  "Project2": {
    "projects": ["Project2"]
  }
}
```

- Execute caso-extract for each Project (and related REGION) to be monitored (Project1-REGION1, Project2-REGION2)

```
/usr/bin/caso-extract --projects "Project1" --config-file /etc/caso/caso-<REGION1>.conf
/usr/bin/caso-extract --projects "Project2" --config-file /etc/caso/caso-<REGION2>.conf
```

CHAPTER

FIVE

USAGE

5.1 command line

`cASO` provides the `caso-extract` command to generate new records from your OpenStack deployment. `caso-extract -h` will show a complete list of available arguments.

Use the `--extract-from` argument to specify the date from when the records should be extracted. If no value is set, then `cASO` will extract the records from the last run. If equal to "None", then extract records from the beggining of time. If not time zone is specified, UTC will be used.

Important: If you are running an OpenStack Nova version lower than Kilo there is a bug in its API, making impossible to paginate over deleted results.

Since nova is limiting the results to 1000 by default, if you are expecting more than 1000 results you will get just the last 1000. This is important if you are publishing data for the first time, or if you are republishing all your accounting). If this is your case, adjust the `osapi_max_limit` to a larger value in `/etc/nova/nova.conf`.

5.1.1 Available options

Apart from other options, the following ones are the ones that specify how to extract accountig records:

`--config-dir` `DIR`

Path to a config directory to pull **.conf* files from. This file set is sorted, so as to provide a predictable parse order if individual options are over-ridden. The set is parsed after the file(s) specified via previous –config-file, arguments hence over-ridden options in the directory take precedence. This option must be set from the command-line.

`--config-file` `PATH`

Path to a config file to use. Multiple config files can be specified, with values in later files taking precedence. Defaults to None. This option must be set from the command-line.

`--debug, -d`

If set to true, the logging level will be set to DEBUG
instead of the default INFO level.

`--dry-run, --dry_run`

Extract records but do not push records to SSM. This will not update the last run date.

--extract-from EXTRACT_FROM, **--extract_from** EXTRACT_FROM

Extract records that have changed after this date. This means that if a record has started before this date, and it has changed after this date (i.e. it is still running or it has ended) it will be reported. If it is not set, extract records from last run. If it is set to None and last run file is not present, it will extract records from the beginning of time. If no time zone is specified, UTC will be used.

--extract-to EXTRACT_TO, **--extract_to** EXTRACT_TO

Extract record changes until this date. If it is not set, we use now. If a server has ended after this date, it will be included, but the consuption reported will end on this date. If no time zone is specified, UTC will be used.

--extractor EXTRACTOR

Which extractor to use for getting the data. If you do not specify anything, nova will be used. Allowed values: nova

--projects PROJECTS

List of projects to extract accounting records from.

5.2 Running as a cron job

The best way of running cASO is via a cron job like the following:

```
10 * * * * caso-extract
```

5.3 Other commands

5.3.1 caso-projects

cASO provides the `caso-projects` command line tool, that will print the list of configured OpenStack Keystone projects and their mappings.

CHAPTER

SIX

TROUBLESHOOTING

6.1 Cannot-find-VM-in-API

Danger: There is not a single recipe to fix this issue, and this involves touching and modifying the DB directly. We reccomend that you ignore these messages, unless you know what you are doing.

In the logs you can see the following warnings (caso version < 1.4.4):

```
WARNING caso.extract.nova [-] Cannot get server '072e77c0-4295-4a83-9bdf-6afde796a00d'␣
↪from the Nova API, probably because it is an old VM that whose metadata is wrong in␣
↪the DB. There will be no record generated for this VM. : NotFound: Instance 072e77c0-
↪4295-4a83-9bdf-6afde796a00d could not be found. (HTTP 404) (Request-ID: req-8eabf5d8-
↪b722-4ee4-b211-aec36fc0499e)
```

Or the following one (caso version >= 1.4.4):

```
WARNING caso.extract.nova [-] Cannot get server '072e77c0-4295-4a83-9bdf-6afde796a00d'␣
↪from the Nova API, probably because it is an error in the DB. Please refer to the␣
↪following page for more details: https://caso.readthedocs.io/en/stable/troubleshooting.
↪html#Cannot-find-VM-in-API
```

These errors are caused by a VM that is in a bad state on the DB. The `os-simple-tenant-usage` API is returning instances that cannot be obtained from the API.

This may be caused by any of the following cases:

1. VMs that have changed their status on a date that enters into the extrating period.
2. VMs that are terminated and deleted, but their status is incorrect (i.e. no value for `terminated_at`). This can be fixed by setting a `terminated_at` value that is correct, directly in the DB.

INDEX

Symbols

C