
OpsCommon Documentation

Release 1.1

Henry Huang

January 08, 2014

1	Getting Started	1
2	Currently Supported API	3
2.1	Getting Started with OpsCommon	3
2.2	IcsEc2 Common Library	5
2.3	IcsELB Common Library	11
2.4	IcsAS Common Library	13
2.5	IcsS3 Common Library	18
2.6	IcsSNS Common Library	19
2.7	IcsR53 Common Library	20
2.8	IcsMeta Common Library	23
2.9	IcsLog Common Library	26
2.10	IcsException Common Library	28
2.11	IcsSecurityGroup Common Library	29
2.12	IcsSQS Common Library	32
2.13	IcsUtils.IcsAlert Common Library	33
2.14	IcsUtils.Misc Common Library	34
2.15	IcsUtils.Daemon Common Library	38
2.16	IcsUtils.CLI Common Library	39
2.17	IcsUtils.JsonCli Common Library	40
2.18	IcsUtils.JsonSubs Common Library	40
2.19	IcsUtils.JsonDiff Common Library	44
3	Indices and tables	47
	Python Module Index	49

Getting Started

If you've never used `OpsLib` before, you should read the *Getting Started with `OpsLib`* guide to get familiar with `OpsLib` & its usage.

Currently Supported API

- **Common Library for Amazon Web Services**

- *IcsEc2 API Reference*
- *IcsELB API Reference*
- *IcsAS API Reference*
- *IcsS3 API Reference*
- *IcsR53 API Reference*
- *IcsSNS API Reference*
- *IcsMeta API Reference*
- *IcsSGroup API Reference*
- *IcsSQS API Reference*

- **Common Library for ICS Utilities**

- *IcsAlert API Reference*
- *Misc API Reference*
- *Daemon API Reference*
- *CLI based on DOC Reference*
- *CLI based on JSON Reference*
- *JSON Template Substitution Reference*
- *JSON Diff API Reference*

- **Common Library for ICS Logging**

- *IcsLog API Reference*

- **Common Library for ICS Exception**

- *IcsException API Reference*

2.1 Getting Started with OpsCommon

This tutorial will walk you through installing and configuring OpsCommon, as well how to use it to make API calls.

This tutorial assumes you are familiar with Python & that you have registered for an [Amazon Web Services](#) account. You'll need retrieve your Access Key ID and Secret Access Key from the web-based console.

2.1.1 Installing OpsCommon

You can use `yum` to install the latest released version of OpsCommon:

```
yum install -y ICS-OpsCommon
```

2.1.2 Importing This Module

You need to add the path of `opslib` module into `sys.path`, and then import it:

```
import sys
sys.path.append("/opt/icsops")
```

2.1.3 Configuring OpsCommon Credentials

You have a few options for configuring OpsCommon. For this tutorial, we'll be using a configuration file.

Default Configuration

There is a `/opt/icsops/opslib/opslib.ini` file with these contents:

```
[Credentials]
aws_access_key_id =
aws_secret_access_key =

[Boto]
num_retries = 5
http_socket_timeout = 70
metadata_service_num_attempts = 5
metadata_service_timeout = 70
```

Please fill in `Credentials` section with your AWS credentials. If you would like to use `IAM Role`, just let it be **empty** here.

Customized Configuration

For your own customized configuration, follow the format with the default configuration above show the path to `opslib` module when you try to import it:

```
import opslib
opslib.init_boto_config("/path/your_own_config_file")
```

2.1.4 Logging Configuration

By default, when you import `opslib` module, it will not enable console output or write logs to log file. So you have to enable it by yourself:


```
import opslib
opslib.init_logging(name="log_handler", logfile="/path/your_log_file",
                   console=1, loglevel="info")
```

2.1.5 Next Steps

For many of the services that OpsCommon supports, there are tutorials as well as detailed API documentation. If you are interested in a specific service, the tutorial for the service is a good starting point. For instance, if you'd like more information on IcsEc2, check out the [IcsEc2 API reference](#).

Here comes an example - we need to fetch all the tags of instance "i-123456" in "us-east-1" region:

```
import opslib
from opslib.icsec2 import IcsEc2
ec2 = IcsEc2("us-east-1")
tags = ec2.get_instance_tags("i-12345678")
```

2.2 IcsEc2 Common Library

2.2.1 IcsEc2: Library for EC2

This is the IcsEc2 common library.

```
class opslib.icsec2.IcsEc2(region, **kwargs)
    ICS Library for EC2

    __init__(region, **kwargs)

    __module__ = 'opslib.icsec2'

    add_instance_tags(instance_id, tags)
        Add tags to the instance

        Parameters instance_id (string) – EC2 instance id startwith 'i-xxxxxxx'

    bind_eip(eip, instance_id)
        Bind EIP address to the instance

        Parameters instance_id (string) – EC2 instance id startwith 'i-xxxxxxx'

        Return type bool

        Returns success or raise IcsEc2Exception

    clean_snapshots(tags, duration)
        Clean up snapshots by specific tags and duration

        Parameters tags (dict) – snapshot tags like

        {
            "Name": "XXX"
        }

        Parameters duration (int) – seconds

        Return type list

        Returns list of cleaned snapshot ids
```

del_instance_tags (*instance_id*, *tags*)

Remove tags of the instance

Parameters **instance_id** (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

del_snapshot (*snapshot_id*)

Delete snapshots by snapshot_id

Parameters **snapshot_id** (*string*) – snapshot Id like ‘snap-xxxxxx’

Return type boolean

Returns true, false, exception

fetch_imageid_by_tags (***tags*)

Fetch the Image Id by specific tags

Parameters **tags** (*dict*) – AMI tags like { ‘Name’: ‘XXX’ }

Return type string

Returns Image Id

fetch_latest_snapshot (*snapshots*)

Find the latest Snapshot

fetch_snapid_by_tags (***tags*)

Find the Snapshot Id by specific tags

Parameters **tags** (*dict*) – snapshot tags like { ‘Name’: ‘XXX’ }

Return type string

Returns Snapshot Id

find_ami_by_tags (*tags*)

Find AMI by specific tags

Parameters **tags** (*dict*) – AMI tags like { ‘Name’: ‘XXX’ }

Return type list

Returns list of boto image objects

find_snapshot_by_tags (*tags*)

Find a snapshot by specific tags

Parameters **tags** (*dict*) – snapshot tags like { ‘Name’: ‘XXX’ }

Return type list

Returns list of boto snapshot objects

static format_tags (*tags*)

Convert { “Name”: “XXX” } to { “tag:Name”: “XXX” }

free_eip (*eip*, *instance_id*)

Free EIP address to the instance

Parameters **instance_id** (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

Return type bool

Returns success or raise IcsEc2Exception

get_all_zones ()

Get all Availability Zones under this region

Return type list

Returns list of availability zones in this region

get_az_from_subnet_id (*subnet_id=None*)

Get the name of Availability Zone by its Subnet Id

Parameters **subnet_id** (*string*) – subnet id

Return type string

Returns availability zone name

get_eips_from_addr (*eip_list*)

Get EIP objects via the list of EIP addresses

Parameters **eip_list** (*list*) – the list of EIP addresses

Return type class

Returns EIP objects in boto

get_eips_from_instance (*instance_id*)

Get EIP objects via the instance id

Parameters **instance_id** (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

Return type class

Returns EIP objects in boto

get_instance_attribute (*instance_id, attr_name*)

Get the attribute value of an instance.

Parameters

- **instance_id** (*string*) – EC2 instance id startwith ‘i-xxxxxxx’
- **attr_name** (*string*) – the name of the instance attribute, details shown as below:

Variables

- **id** – The unique ID of the Instance.
- **groups** – A list of Group objects representing the security groups associated with the instance.
- **public_dns_name** – The public dns name of the instance.
- **private_dns_name** – The private dns name of the instance.
- **state** – The string representation of the instance’s current state.
- **state_code** – An integer representation of the instance’s current state.
- **previous_state** – The string representation of the instance’s previous state.
- **previous_state_code** – An integer representation of the instance’s current state.
- **key_name** – The name of the SSH key associated with the instance.
- **instance_type** – The type of instance (e.g. m1.small).
- **launch_time** – The time the instance was launched.
- **image_id** – The ID of the AMI used to launch this instance.
- **placement** – The availability zone in which the instance is running.

- **placement_group** – The name of the placement group the instance is in (for cluster compute instances).
- **placement_tenancy** – The tenancy of the instance, if the instance is running within a VPC. An instance with a tenancy of dedicated runs on a single-tenant hardware.
- **kernel** – The kernel associated with the instance.
- **ramdisk** – The ramdisk associated with the instance.
- **architecture** – The architecture of the image (i386|x86_64).
- **hypervisor** – The hypervisor used.
- **virtualization_type** – The type of virtualization used.
- **product_codes** – A list of product codes associated with this instance.
- **ami_launch_index** – This instances position within it's launch group.
- **monitored** – A boolean indicating whether monitoring is enabled or not.
- **monitoring_state** – A string value that contains the actual value of the monitoring element returned by EC2.
- **spot_instance_request_id** – The ID of the spot instance request if this is a spot instance.
- **subnet_id** – The VPC Subnet ID, if running in VPC.
- **vpc_id** – The VPC ID, if running in VPC.
- **private_ip_address** – The private IP address of the instance.
- **ip_address** – The public IP address of the instance.
- **platform** – Platform of the instance (e.g. Windows)
- **root_device_name** – The name of the root device.
- **root_device_type** – The root device type (ebsinstance-store).
- **block_device_mapping** – The Block Device Mapping for the instance.
- **state_reason** – The reason for the most recent state transition.
- **groups** – List of security Groups associated with the instance.
- **interfaces** – List of Elastic Network Interfaces associated with this instance.
- **ebs_optimized** – Whether instance is using optimized EBS volumes or not.
- **instance_profile** – A Python dict containing the instance profile id and arn associated with this instance.

get_instance_event (*instance_id*)

Get the event of the specified instance

Parameters *instance_id* (*string*) – EC2 instance id startwith 'i-xxxxxxx'

get_instance_status (*instance_id*)

Get the instance status and system status of the specified instance

Parameters *instance_id* (*string*) – EC2 instance id startwith 'i-xxxxxxx'

Return type tuple

Returns a tuple contains (instance_status, system_status)

get_instance_tags (*instance_id*)

Get tags of the instance

Parameters *instance_id* (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

Return type dict

Returns a dictionary containing the tags of this instance

get_private_address (*instance_id*)

Get the private IPv4 address of the instance

Parameters *instance_id* (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

Return type string

Returns a string containing the private IPv4 address

get_private_dns (*instance_id*)

Get the private dns address of the instance

Parameters *instance_id* (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

Return type string

Returns a string containing the private IPv4 address

get_public_address (*instance_id*)

Get the public IPv4 address of the instance

Parameters *instance_id* (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

Return type string

Returns a string containing the public IPv4 address

get_public_dns (*instance_id*)

Get the public dns address of the instance

Parameters *instance_id* (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

Return type string

Returns a string containing the public dns address

get_security_group_id (*name*, *vpc_id=None*)

Get security group id

Parameters

- *name* (*string*) – security group name
- *vpc_id* (*string*) – vpc id

Return type string

Returns security group id

get_sgroup (*name*, *vpc_id=None*)

Get Security Group Name (if Ec2) / Id (if Vpc)

Parameters

- *name* (*string*) – security group name
- *vpc_id* (*string*) – vpc id

Return type string

Returns security group id

get_volumes_by_instance (*instance_id*)

Get boto Volume Objects by instance Id

Parameters *instance_id* (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

Return type list

Returns list of boto volume objects

get_zone_index_for_cassandra (*index*)

Get the index of Availability Zone for Cassandra

Parameters *index* (*int*) – the index of cassandra instance

Return type string

Returns zone index like “1”

get_zone_name_for_cassandra (*index*)

Get the name of Availability Zone for Cassandra

Parameters *index* (*int*) – the index of cassandra instance

Return type string

Returns zone name like “us-west-2a”

get_zone_suffix_for_cassandra (*index*)

Get the suffix of Availability Zone for Cassandra

Parameters *index* (*int*) – the index of cassandra instance

Return type string

Returns zone suffix like “a-1”

is_eip_free (*eip*)

check the availability of the specified EIP address: free or not

Parameters *eip* (*string*) – one EIP address

Return type tuple

Returns (True/False, EIP object/None)

is_instance_healthy (*instance_id*)

check the health of the specified instance

Parameters *instance_id* (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

Return type boolean

Returns True/False

size_of_all_zones ()

Get the number of all Availability Zones under this region

Return type int

Returns number of availability zones in this region

take_snapshot (*volume_id*, *description=None*, *tags=None*)

Take a snapshot to existing volume with specific tags

Parameters

- *volume_id* (*string*) – EC2 volume id startwith ‘vol-xxxxxxx’

- **description** (*string*) – words to describe the usage of this snapshot
- **tags** (*dict*) – snapshot tags like { 'Name': 'XXX' }

Return type class

Returns boto snapshot object

2.2.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.3 IcsELB Common Library

2.3.1 IcsELB: Library for ELB

This is the IcsELB common library.

class `opslib.icselb.IcsELB` (*region*, ***kwargs*)
ICS Library for ELB

`__dict__` = `dict_proxy({'get_elb_dns_name': <function get_elb_dns_name at 0x2f8c5f0>, 'remove_elb_listeners': <funct`

`__init__` (*region*, ***kwargs*)

`__module__` = 'opslib.icselb'

`__weakref__`

list of weak references to the object (if defined)

create_elb (*name*, *zones*, *listeners=None*, *subnets=None*, *groups=None*)

Create an ELB named <name>

Parameters

- **name** (*string*) – The mnemonic name associated with the new load balancer
- **zones** (*list of strings*) – The names of the availability zone(s) to add
- **listeners** – Each tuple contains three or four values:
 - LoadBalancerPortNumber and InstancePortNumber are integer values between 1 and 65535;
 - Protocol is a string containing 'TCP', 'SSL', 'HTTP', 'HTTPS';
 - SSLCertificateID is the ARN of a AWS AIM certificate, and must be specified when doing HTTPS

Parameters

- **subnets** (*list of strings*) – A list of subnet IDs in your VPC to attach to your LoadBalancer
- **groups** (*list of strings*) – The security groups assigned to your LoadBalancer within your VPC

delete_elb (*name*)

Remove an load balancers from your account

Parameters **name** (*string*) – The name of the Load Balancer to delete

get_all_elbs (*load_balancer_names=None*)

Get all load balancers in this region

Parameters **load_balancer_names** (*list*) – An optional list of load balancer names

Return type boto.resultset.ResultSet

Returns A ResultSet containing instances of boto.ec2.elb.loadbalancer.LoadBalancer

get_elb_dns_name (*name*)

Get Load Balancer DNS name. :type name: str :param name: The load balances name

Return type str

Retrun load balancer hosted zone id

get_elb_health (*name, instance_id*)

check the health of the specified instance in the specified elb

Parameters

- **name** (*string*) – The load balances name
- **instance_id** (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

Return type boolean or string

Returns False or out of service reason

get_elb_id (*name*)

Get Load Balancer Hosted Zone ID

Parameters **name** (*str*) – The load balances name

Return type str

Retrun load balancer hosted zone id

parse_listeners (*listeners*)

Parse elb listeners form list of string to list of tuple

Parameters **listeners** (*list of string*) – Listeners of this elb

Returns The list of listeners tuple

Type list of tuple

remove_elb_listeners (*name, listeners*)

Remove a Listener (or group of listeners) for an existing Load Balancer

Parameters

- **name** (*string*) – The name of the load balancer to create the listeners for
- **listeners** (*List int*) – Each int represents the port on the ELB to be removed

set_elb_listeners (*name, listeners*)

Create a Listener (or group of listeners) for an existing Load Balancer

Parameters

- **name** (*string*) – The name of the load balancer to create the listeners for
- **listeners** – Listener to be setted

set_health_check (*name, health_check*)

Configures the health check behavior for the instances behind this load balancer.

Parameters

- **name** (*string*) – The mnemonic name associated with the load balancer
- **health_check** (*boto.ec2.elb.healthcheck.HealthCheck*) – A HealthCheck instance that tells the load balancer how to check its instances for health.

2.3.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.4 IcsAS Common Library

2.4.1 IcsAS: Library for Auto Scaling

This is the IcsAS common library.

class `opslib.icsas.IcsAS` (*region*, ***kwargs*)

ICS Library for AutoScale

__dict__ = `dict_proxy({'__module__': 'opslib.icsas', 'resume_scaling_group': <function resume_scaling_group at 0x3b>}`

__init__ (*region*, ***kwargs*)

__module__ = 'opslib.icsas'

__weakref__

list of weak references to the object (if defined)

create_launch_config (*launch_config*)

Create the Launch Configuration

Parameters **launch_config** (*class*) – boto launch_config object

Return type string

Returns AWS request Id

delete_launch_config_from_name (*name*)

Delete the Launch Configuration from its name

Parameters **name** (*string*) – the name of launch configuration

Return type string

Returns AWS request Id

get_group_from_name (*name*)

Get the ASG from its name

Parameters **name** (*string*) – the ASG name

Return type list

Returns a list represents the specific ASG(s)

get_group_name_from_instance (*instance_id*)

Get the ASG name from the specific instance id

Parameters `instance_id` (*string*) – EC2 instance id startwith ‘i-xxxxxxx’

Return type `string`

Returns name of the ASG, this instance belongs to

get_instances_from_group_name (*name*)

Get the instance from the specific ASG name

Parameters `name` (*string*) – the specific ASG name

Return type `list`

Returns a list contains all the instances

get_launch_config_from_name (*name*)

Get the Launch Configuration from its name

Parameters `name` (*string*) – the Launch Configuration name

Return type `list`

Returns a list represents the specific Launch Configuration(s)

resume_scaling_group (*name*, *scaling_processes=None*)

Resumes Auto Scaling processes for an Auto Scaling group.

Parameters

- `name` (*string*) – the ASG name
- `scaling_processes` (*string or list*) – scaling process names
 - Launch
 - Terminate
 - HealthCheck
 - ReplaceUnhealthy
 - AZRebalance
 - AlarmNotification
 - ScheduledActions
 - AddToLoadBalancer

suspend_scaling_group (*name*, *scaling_processes=None*)

Suspends Auto Scaling processes for an Auto Scaling group.

Parameters

- `name` (*string*) – the ASG name
- `scaling_processes` (*string or list*) – scaling process names
 - Launch
 - Terminate
 - HealthCheck
 - ReplaceUnhealthy
 - AZRebalance
 - AlarmNotification

- ScheduledActions
- AddToLoadBalancer

```
terminate_group_instance(instance_id, decrement_capacity=True)
```

Terminates the specified instance. The desired group size can also be adjusted, if desired.

Parameters

- **instance_id** (*str*) – The ID of the instance to be terminated.
- **decrement_capacity** – Whether to decrement the size of the autoscaling group or not.

```
to_list (input)
```

Validate input, if not list, but string, make it as a list

update_instance_health (*instance_id*, *health_status*, *grace_period=False*)

Explicitly set the health status of an instance.

Parameters

- **instance_id** (*str*) – The identifier of the EC2 instance
 - **health_status** (*str*) – The health status of the instance.
- Healthy: the instance is healthy and should remain in service.
 - Unhealthy: the instance is unhealthy. Auto Scaling should terminate and replace it.

Parameters `grace_period` (*bool*) – If True, this call should respect the grace period associated with the group.

update_launch_config(*name*, *launch_config*)

Update the Launch Configuration for specific ASG

Parameters

- **name** (*string*) – the name of Auto-Scaling Group
- **launch_config** (*class*) – boto launch_config object

Return type string

Returns AWS request Id

```
class opslib.icsas.RawAS (region)
```

Raw Library for AutoScale, based on Botocore

```
__dict__ = dict_proxy({'__module__': 'opslib.icasas', 'create_group': <function create_group at 0x3b5e668>, 'kill_grou
```

__init__ (*region*)

Initialize the proper botocore service

```
__module__ = 'opslib.icsas'
```

__weakref__

list of weak references to the object (if defined)

```
create_group (group_config)
```

Create a new Auto-Scaling Group

Parameters `group_config` (*dict*) – auto-scaling group configuration

```
create_launch_config(launch_config)
```

Create a new Launch Configuration

Parameters `launch_config` (*dict*) – launch configuration

delete_group (*name*)

Delete an existing Auto-Scaling Group

Parameters `name` (*string*) – auto-scaling group name

delete_launch_config (*name*)

Delete an existing Launch Configuration

Parameters `name` (*string*) – launch configuration name

delete_scaling_policy (*scaling_policy, metric_alarm*)

Delete an existing Scaling Policy

Parameters

- `scaling_policy` (*dict*) – scaling policy configuration
- `metric_alarm` (*dict*) – metric alarm configuration

fetch_all_groups ()

Fetch all the Auto-Scaling Groups

Return type dict

Returns JSON object for all the Auto-Scaling Groups

fetch_group (*name*)

Fetch an existing Auto-Scaling Group

Parameters `name` (*string*) – auto-scaling group name

fetch_launch_config (*name*)

Fetch an existing Launch Configuration

Parameters `name` (*string*) – launch configuration name

find_groups (*filter={}*)

Find the names of Auto-Scaling Groups in the filters

Parameters `filter` (*dict*) – a dictionary to used for resource filtering The format should be consistent with botocore JSON output

```
{
  "Tags": [
    {
      "Key": "Owner",
      "Value": "Production"
    }
  ]
}
```

Return type list

Returns a list containing all the names of filtered groups

handle_response (*response*)

Handle the botocore response

kill_group (*group_name, force=False*)

Delete a new Auto-Scaling Group

Parameters

- **name** (*string*) – launch configuration name
- **force** (*boolean*) – whether to delete the auto-scaling group forcibly

launch_group (*group_config*, *launch_config*)

Launch a new Auto-Scaling Group

Parameters

- **group_config** (*dict*) – auto-scaling group configuration
- **launch_config** (*dict*) – launch configuration

modify_group (*group_config*)

Modify the Auto-Scaling Group

Parameters **group_config** (*dict*) – auto-scaling group configuration

modify_launch_config (*launch_config*, *delimiter*='_U_')

Modify the Launch Configuration

Parameters **launch_config** (*dict*) – launch configuration

new_scaling_policy (*scaling_policy*, *metric_alarm*)

Create a new Scaling Policy

Parameters

- **scaling_policy** (*dict*) – scaling policy configuration
- **metric_alarm** (*dict*) – metric alarm configuration

update_group (*group_config*, *launch_config*)

Update the Auto-Scaling Group

Parameters

- **group_config** (*dict*) – auto-scaling group configuration
- **launch_config** (*dict*) – launch configuration

`opslib.icsas.get_region(region_name, **kw_params)`

Find and return a `boto.ec2.autoscale.RegionInfo` object given a region name.

Parameters **region_name** (*str*) – The name of the region.

Return type `boto.ec2.autoscale.RegionInfo`

Returns The `RegionInfo` object for the given region or `None` if an invalid region name is provided.

2.4.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.5 IcsS3 Common Library

2.5.1 IcsS3: Library for S3

This is the IcsS3 common library.

class `opslib.icss3.IcsS3` (***kwargs*)
ICS Library for S3

`__init__` (***kwargs*)

`__module__` = 'opslib.icss3'

add_rule (*id=None, prefix=None, status=None, expiration=None, transition=None*)

Add a rule to this Lifecycle configuration. This only adds the rule to the local copy. To install the new rule(s) on the bucket, you need to pass this Lifecycle config object to the `configure_lifecycle` method of the Bucket object.

Parameters

- **id** (*str*) – Unique identifier for the rule. The value cannot be longer than 255 characters.
- **status** (*str*) – If 'Enabled', the rule is currently being applied. If 'Disabled', the rule is not currently being applied.
- **expiration** (*int*) – Indicates the lifetime, in days, of the objects that are subject to the rule. The value must be a non-zero positive integer. A Expiration object instance is also perfect.
- **transition** (*Transition*) – Indicates when an object transitions to a different storage class.

Iparam prefix Prefix identifying one or more objects to which the rule applies.

batch_download (*uri, pattern='', dirname=None*)

Batch download files from S3 (only for current folder)

Parameters

- **url** (*str*) – File URL in S3, like `s3://bucket/path`
- **pattern** (*string*) – regex expression to match
- **dirname** (*string*) – local path to save, 'None' by default

Return type list

Returns a list containing dowloaded file path

configure_s3rule (*bucket, rules=None*)

Parameters

- **bucket** (*object*) – the boto object of S3 bucket
- **rules** (*dict*) – describes the lifecycle rules

Return type list

Returns a list of results

create_bucket (*bucket_name, headers=None, location='us-west-2', policy=None*)

Creates a new located bucket. By default it's in the USA.

Parameters

- **bucket_name** (*string*) – The name of the new bucket

- **headers** (*dict*) – Additional headers to pass along with the request to AWS.
- **location** (*str*) – The location of the new bucket, like “us-west-2”
- **policy** (`boto.s3.acl.CannedACLStrings`) – A canned ACL policy that will be applied to the new key in S3.

download_file (*uri*, *fp*)
Download a file from S3

Parameters

- **url** (*str*) – File URL in S3, like `s3://bucket/path`
- **fp** (*file*) – file descriptor from local file

Return type string

Returns a string containing downloaded file name

get_file_as_string (*uri*)
Get a file as string from S3

Parameters **url** (*str*) – File URL in S3, like `s3://bucket/path`

Return type string

Returns a string containing downloaded file content

recursive_download (*uri*, *pattern*='', *dirname*=None)
Recursive download files from S3

Parameters

- **url** (*str*) – File URL in S3, like `s3://bucket/path`
- **pattern** (*string*) – regex expression to match
- **dirname** (*string*) – local path to save, ‘None’ by default

Return type list

Returns a list containing dowloaded file path

2.5.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.6 IcsSNS Common Library

2.6.1 IcsSNS: Library for SNS

This is the IcsSNS common library.

class `opslib.icssns.IcsSNS` (*region*, ***kwargs*)
ICS Libraray for SNS

__dict__ = `dict_proxy({'__module__': 'opslib.icssns', 'getTopicARN': <function getTopicARN at 0x4b29050>, 'publish`

```
__init__(region, **kwargs)
__module__ = 'opslib.icssns'
__weakref__
    list of weak references to the object (if defined)
getTopicARN(name)
    Get the Amazon Resource Name of specified SNS Topic
    Parameters name (str) – SNS Topic Name
    Return type string
    Returns a string containing Amazon Resource Name
publish(topic_arn, msg, subj=None)
    Publish messages to SNS Topic
    Parameters
    • topic_arn (string) – Amazon Resource Name for SNS
    • msg (string) – message contents
    • subj (string) – subject contents
```

2.6.2 Indices and tables

- [genindex](#)
- [modindex](#)
- [search](#)

2.7 IcsR53 Common Library

2.7.1 IcsR53: Library for Route53

This is the IcsR53 common library.

```
class opslib.icsr53.IcsR53(dns_name=None, **kwargs)
    ICS Library for R53
    __dict__ = dict_proxy({'add_a': <function add_a at 0x4c83c80>, '__module__': 'opslib.icsr53', 'find_records': <function find_records at 0x4c83c80>, 'get_records': <function get_records at 0x4c83c80>, 'new_alias_record': <function new_alias_record at 0x4c83c80>, 'new_record': <function new_record at 0x4c83c80>, 'update_record': <function update_record at 0x4c83c80>, 'update_alias_record': <function update_alias_record at 0x4c83c80>})
    __init__(dns_name=None, **kwargs)
    __module__ = 'opslib.icsr53'
    __weakref__
        list of weak references to the object (if defined)
    add_a(name, value, ttl=None, identifier=None)
        Add a new A record to this Zone. See _new_record for parameter documentation. Returns a Status object.
    add_alias(name, type, alias_hosted_zone_id, alias_dns_name, identifier=None)
        Add a new alias record to this Zone. See _new_alias_record for parameter documentation. Returns a Status object.
```


add_cname (*name, value, ttl=None, identifier=None*)

Add a new CNAME record to this Zone. See `_new_record` for parameter documentation. Returns a Status object.

add_record (*resource_type, name, value, ttl=60, identifier=None*)

Add a new record to this Zone. See `_new_record` for parameter documentation. Returns a Status object.

delete_a (*name, identifier=None, all=False*)

Delete an A record matching name and identifier from this Zone. Returns a Status object.

If there is more than one match delete all matching records if `all` is `True`, otherwise throws `TooManyRecordsException`.

delete_cname (*name, identifier=None, all=False*)

Delete a CNAME record matching name and identifier from this Zone. Returns a Status object.

If there is more than one match delete all matching records if `all` is `True`, otherwise throws `TooManyRecordsException`.

delete_record (*record*)

Delete one or more records from this Zone. Returns a Status object.

Parameters **record** – A ResourceRecord (e.g. returned by `find_records`) or list, tuple, or set of ResourceRecords.

find_all_records ()

Search all records in this zone.

find_records (*name, type, desired=1, all=False, identifier=None*)

Search this Zone for records that match given parameters. Returns `None` if no results, a ResourceRecord if one result, or a ResourceRecordSets if more than one result.

Parameters

- **name** (*str*) – The name of the records should match this parameter
- **type** (*str*) – The type of the records should match this parameter
- **desired** (*int*) – The number of desired results. If the number of matching records in the Zone exceeds the value of this parameter, throw `TooManyRecordsException`
- **all** (*Boolean*) – If true return all records that match name, type, and identifier parameters
- **identifier** (*Tuple*) – A tuple specifying WRR or LBR attributes. Valid forms are:
 - (str, str): WRR record [e.g. ('foo', '10')]
 - (str, str): LBR record [e.g. ('foo', 'us-east-1')]

get_a (*name, identifier=None, all=False*)

Search this Zone for A records that match name.

Returns a ResourceRecord.

If there is more than one match return all as a ResourceRecordSets if `all` is `True`, otherwise throws `TooManyRecordsException`.

get_cname (*name, identifier=None, all=False*)

Search this Zone for CNAME records that match name.

Returns a ResourceRecord.

If there is more than one match return all as a ResourceRecordSets if `all` is `True`, otherwise throws `TooManyRecordsException`.

get_records ()

Return a ResourceRecordsSets for all of the records in this zone.

get_zone (*name*)

Get the hosted zone for the specified domain name

Parameters *name* (*string*) – the specified domain name

Return type class

Returns a class containing the specified hosted zone

get_zone_dict (*name*)

Get the hosted zone info for the specified domain name

Parameters *name* (*string*) – the specified domain name

Return type dict

Returns a dict containing the specified hosted zone info

get_zone_id ()

Get the hosted zone ID for the specified domain name

Return type string

Returns a string containing the ID of the specified hosted zone

static parse_dns_name (*name*)

Parse the value of Tag “DnsName”

Parameters *name* (*string*) – the value of Instance Tag “DnsName” for example, “test.example.com:A:Public:1”

Return type tuple

Returns a tuple containing (DnsName, DnsType, Public/Private, Weight) for example, (“test.example.com”, “A”, True, “10”)

set_zone (*name*)

Set the hosted zone for the specified domain name

Parameters *name* (*string*) – the specified domain name

update_a (*name, value, ttl=None, identifier=None*)

Update the given A record in this Zone to a new value, ttl, and identifier. Returns a Status object.

Will throw TooManyRecordsException is name, value does not match a single record.

update_alias (*name, type, identifier=None, alias_dns_name=None*)

Update the given alias record in this Zone to a new routing policy Returns a Status object.

Will throw TooManyRecordsException is name, value does not match a single record.

update_cname (*name, value, ttl=None, identifier=None*)

Update the given CNAME record in this Zone to a new value, ttl, and identifier. Returns a Status object.

Will throw TooManyRecordsException is name, value does not match a single record.

update_record (*old_record, new_value, new_ttl=None, new_identifier=None*)

Update an existing record in this Zone. Returns a Status object.

Parameters *old_record* (*ResourceRecord*) – A ResourceRecord (e.g. returned by find_records)

See *_new_record* for additional parameter documentation.

wait_to_complete (*status=None, timeout=60*)
Wait for the Route53 commit change to complete

Parameters **status** (*class*) – the instance initializing `boto.route53.status.Status`

2.7.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.8 IcsMeta Common Library

2.8.1 IcsMeta: Library for Meta

This is the IcsMeta common library.

class `opslib.icsmeta.IcsMeta`

__dict__ = `dict_proxy({'get_role_name': <function get_role_name at 0x4b29aa0>, 'init_alert': <function init_alert at 0`

__init__ ()
Initialize Ics Meta (meta-data, user-data, credentials, tags)

__module__ = `'opslib.icsmeta'`

__weakref__
list of weak references to the object (if defined)

download_cfg (*pattern*)
Download configuration files from S3
Parameters **pattern** (*string*) – regex expression to match

Return type `string`

Returns the local path where downloaded files stored

download_script (*pattern*)
Download scripts from S3
Parameters **pattern** (*string*) – regex expression to match

Return type `string`

Returns the local path where downloaded files stored

generate_hostname ()
Generate the hostname

Return type `string`

Returns the hostname

get_cfg_bucket ()
Get the Config Bucket from instance user-data

Return type `string`

Returns the Config Bucket name

get_credentials()

Get AWS credentials from instance user-data

Return type dict

Returns AWS credentials as a dictionary

get_dns_from_tag()

Get the DnsName from instance tags

Return type string

Returns the DnsName

get_eips_from_tag()

Get the EIP list from instance tags

Return type list

Returns the list contains EIP addresses

get_instance_id()

Get the instance id from instance meta-data

Return type string

Returns the instance id

get_instance_name()

Get the Instance name from instance tags

Return type string

Returns the Instance name

get_machine_tags(timeout=120)

Get the instance tags

Return type string

Returns the tags of this instance

static get_meta_data(timeout=None, url=None, num_retries=None)

Get instance meta data

Parameters

- **timeout** (*int*) – timeout for the request
- **url** (*string*) – metadata_service_url
- **num_retries** (*int*) – how many times to retry

Return type dict

Returns instance meta data as a dictionary

get_openssh_pubkey()

Get the openssh public key from instance meta-data

Return type string

Returns the contents of openssh public key

get_private_ip()

Get the private ip address from instance meta-data

Return type string

Returns the private ip address

get_public_hostname()

Get the public hostname from instance meta-data

Return type string

Returns the public hostname

get_public_ip()

Get the public ip address from instance meta-data

Return type string

Returns the public ip address

get_region()

Get the region from instance meta-data

Return type string

Returns the region name

get_role_name()

Get the Role name from instance tags

Return type string

Returns the Role name

get_script_url()

Get the S3 url for import scripts from instance user-data

Return type string

Returns the s3 url for scripts

get_sns_topic()

Get the SNS Topic from instance user-data

Return type string

Returns the SNS Topic name

static get_user_data (*timeout=None, url=None, num_retries=None*)

Get instance user data

Parameters

- **timeout** (*int*) – timeout for the request
- **url** (*string*) – metadata_service_url
- **num_retries** (*int*) – how many times to retry

Return type dict

Returns instance user data as a dictionary

get_zone()

Get the availability zone from instance meta-data

Return type string

Returns the availability zone name

init_alert (*prefix*=*'ICS'*)

Intialize ICS Alert

Parameters **prefix** (*str*) – prefix string to indicate which process

Return type class

Returns the instance of initialized IcsAlert class

init_config ()

Combine meta-data, user-data, tags into one json string

Return type dict

Returns json string contains meta-data, user-data, tags

is_eip_ready (*eip*)

check the readiness of the specified EIP address

Parameters **eip** (*string*) – one EIP address

Return type boolean

Returns True/False

2.8.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.9 IcsLog Common Library

2.9.1 IcsLog: Library for Log

This is the IcsLog common library.

class opslib.icslog.**IcsLog** (*name*, *console*=*True*, *logfile*=*None*, *level*=*'DEBUG'*)

ICS Log Library

__dict__ = dict_proxy({'__module__': 'opslib.icslog', 'set_debug_level': <function set_debug_level at 0x2332aa0>, 'wa

__init__ (*name*, *console*=*True*, *logfile*=*None*, *level*=*'DEBUG'*)

Initialize the Ics Log

Parameters

- **name** (*string*) – the logger name, this param should be different for different loggers
- **console** (*int*) – whether output the log to the console, value should be 0 or 1
- **logfile** (*string*) – the file to save the logs

Return type class object

Returns a log object

__module__ = 'opslib.icslog'

__weakref__

list of weak references to the object (if defined)

_lock()

Lock the file

_unlock()

Unlock the file

critical (*msg*, **args*, ***kwargs*)

Logs a message with level critical on this logger.

Parameters

- **msg** (*string*) – message format string
- **args** (*arguments*) – the arguments which are merged into msg using the string formatting operator
- **kwargs** (*not recommended to use*) – not recommended to use

debug (*msg*, **args*, ***kwargs*)

Logs a message with level DEBUG on this logger.

Parameters

- **msg** (*string*) – message format string
- **args** (*arguments*) – the arguments which are merged into msg using the string formatting operator
- **kwargs** (*not recommended to use*) – not recommended to use

error (*msg*, **args*, ***kwargs*)

Logs a message with level error on this logger.

Parameters

- **msg** (*string*) – message format string
- **args** (*arguments*) – the arguments which are merged into msg using the string formatting operator
- **kwargs** (*not recommended to use*) – not recommended to use

exception (*msg*, **args*, ***kwargs*)

Logs a message with level exception on this logger.

Parameters

- **msg** (*string*) – message format string
- **args** (*arguments*) – the arguments which are merged into msg using the string formatting operator
- **kwargs** (*not recommended to use*) – not recommended to use

info (*msg*, **args*, ***kwargs*)

Logs a message with level info on this logger.

Parameters

- **msg** (*string*) – message format string
- **args** (*arguments*) – the arguments which are merged into msg using the string formatting operator

- **kwargs** (*not recommended to use*) – not recommended to use

set_critical_level()

Sets the threshold for this logger to critical. Logging messages which are less severe than critical will be ignored.

set_debug_level()

Sets the threshold for this logger to debug. Logging messages will all be printed.

set_error_level()

Sets the threshold for this logger to error. Logging messages which are less severe than error will be ignored.

set_info_level()

Sets the threshold for this logger to info. Logging messages which are less severe than info will be ignored.

set_warning_level()

Sets the threshold for this logger to warning. Logging messages which are less severe than warning will be ignored.

warning (*msg, *args, **kwargs*)

Logs a message with level warning on this logger.

Parameters

- **msg** (*string*) – message format string
- **args** (*arguments*) – the arguments which are merged into msg using the string formatting operator
- **kwargs** (*not recommended to use*) – not recommended to use

class `opslib.icslog.NullHandler` (*level=0*)

`__module__` = 'opslib.icslog'

emit (*record*)

2.9.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.10 IcsException Common Library

2.10.1 IcsException: Library for Exception

This is the IcsException common library.

exception `opslib.icsexception.IcsAseException`

Error for Autoscale request

`__module__` = 'opslib.icsexception'

exception `opslib.icsexception.IcsAlertException`

Error generating or sending alerts


```

    __module__ = 'opslib.icsexception'
exception opslib.icsexception.IcsEc2Exception
    Error for EC2 request
    __module__ = 'opslib.icsexception'
exception opslib.icsexception.IcsException
    Error for ics exception
    __module__ = 'opslib.icsexception'
    __weakref__
        list of weak references to the object (if defined)
exception opslib.icsexception.IcsMetaException
    Error for ICS Meta process
    __module__ = 'opslib.icsexception'
exception opslib.icsexception.IcsR53Exception
    Errors processing ICS R53 request
    __module__ = 'opslib.icsexception'
exception opslib.icsexception.IcsS3Exception
    Errors processing ICS S3 request
    __module__ = 'opslib.icsexception'
exception opslib.icsexception.IcsSNSException
    Errors processing ICS SNS request
    __module__ = 'opslib.icsexception'
exception opslib.icsexception.IcsSysCfgException
    Errors for ICS System Configuration
    __module__ = 'opslib.icsexception'

```

2.10.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.11 IcsSecurityGroup Common Library

2.11.1 IcsSecurityGroup: Library for AWS Security group

This is the IcsSecurityGroup common library.

```

class opslib.icssgroup.IcsSGGroup(region, **kwargs)
    Library for AWS Security group
    __dict__ = dict_proxy({'create_rds_group': <function create_rds_group at 0x490d230>, 'add_egress_rules': <function
    __init__ (region, **kwargs)

```

__module__ = 'opslib.icssgroup'

__weakref__

list of weak references to the object (if defined)

add_egress_rules (*group_id*, *ip_protocol*, *from_port=None*, *to_port=None*, *cidr_ip=None*,
des_group_id=None)

The action adds one or more egress rules to a VPC security group

Parameters

- **group_id** (*str*) – ID of the VPC security group to modify
- **ip_protocol** (*string*) – Either tcp | udp | icmp
- **from_port** (*int*) – The beginning port number you are enabling
- **to_port** (*int*) – The ending port number you are enabling
- **cidr_ip** (*list of strings*) – The CIDR block you are providing access to
- **des_group_id** (*str*) – The ID of destination security groups in the same VPC

add_ingress_rules (*group_name*, *src_group=None*, *ip_protocol=None*, *from_port=None*,
to_port=None, *cidr_ip=None*, *group_id=None*, *src_group_id=None*)

Add a new rule to an existing security group. You need to pass in either *src_security_group_name* OR *ip_protocol*, *from_port*, *to_port*, and *cidr_ip*. In other words, either you are authorizing another group or you are authorizing some ip-based rule.

Parameters

- **group_name** (*string*) – The name of the security group you are adding the rule to
- **src_security_group_name** (*string*) – The name of the security group you are granting access to
- **ip_protocol** (*string*) – Either tcp | udp | icmp
- **from_port** (*int*) – The beginning port number you are enabling
- **to_port** (*int*) – The ending port number you are enabling
- **cidr** (*list of strings*) – The CIDR block you are providing access to
- **group_id** (*string*) – ID of the EC2 or VPC security group to modify. This is required for VPC security groups and can be used instead of *group_name* for EC2 security groups

Returns True if successful.

Type bool

create_group (*name*, *description*, *vpc_id=None*)

Create a new security group for your account. This will create the security group within the region you are currently connected to.

Parameters

- **name** (*string*) – The name of the new security group
- **description** (*string*) – The description of the new security group
- **vpc_id** (*string*) – The ID of the VPC to create the security group in.

Returns The newly created boto.ec2.securitygroup.SecurityGroup.

Type boto.ec2.securitygroup.SecurityGroup

create_rds_group (*name*, *description=None*)

Create a new security group for your account. This will create the security group within the region you are currently connected to.

Parameters

- **name** (*string*) – The name of the new security group
- **description** (*string*) – he description of the new security group

Returns The newly created DBSecurityGroup

Type boto.rds.dbsecuritygroup.DBSecurityGroup

get_security_groups (*groupnames=None*, *group_ids=None*, *filters=None*)

Get all security groups associated with your account in a region.

Parameters

- **groupnames** (*list*) – A list of the names of security groups to retrieve. If not provided, all security groups will be returned
- **group_ids** (*list*) – A list of IDs of security groups to retrieve for security groups within a VPC
- **filters** (*dict*) – Optional filters that can be used to limit the results returned. Filters are provided in the form of a dictionary consisting of filter names as the key and filter values as the value. The set of allowable filter names/values is dependent on the request being performed.

rds_authorize_group (*group_name*, *cidr_ip=None*, *src_group_name=None*,
src_group_owner_id=None)

Add a new rule to an existing security group. You need to pass in either *src_security_group_name* and *src_security_group_owner_id* OR a CIDR block but not both.

Parameters

- **group_name** (*string*) – The name of the security group adding the rule to.
- **cidr_ip** (*string*) – The CIDR block you are providing access to.
- **src_group_name** (*string*) – The name of the EC2 security group you are granting access to.
- **src_group_owner_id** (*string*) – The ID of the owner of the EC2 security group you are granting access to.

Returns True if successful.

Type bool

rds_revoke_rules (*group_name*, *src_group_name=None*, *src_group_owner_id=None*,
cidr_ip=None)

Remove an existing rule from an existing security group. You need to pass in either *ec2_security_group_name* and *ec2_security_group_owner_id* OR a CIDR block.

Parameters

- **group_name** (*string*) – The name of the security group you are removing the rule from.
- **src_group_name** (*string*) – The name of the EC2 security group from which you are removing access.
- **src_group_owner_id** (*string*) – The ID of the owner of the EC2 security from which you are removing access.

- **cidr_ip** (*string*) – The CIDR block from which you are removing access.

Returns True if successful.

Type bool

remove_ingress_rules (*group_name*, *src_group=None*, *ip_protocol=None*, *from_port=None*,
to_port=None, *cidr_ip=None*, *group_id=None*, *src_group_id=None*)

Remove an existing rule from an existing security group

Parameters

- **group_name** (*string*) – The name of the security group you are removing the rule from
- **src_security_group_name** (*string*) – The name of the security group you are revoking access to
- **ip_protocol** (*string*) – Either tcp | udp | icmp
- **from_port** (*int*) – The beginning port number you are disabling
- **to_port** (*int*) – The ending port number you are disabling
- **cidr** (*list of strings*) – The CIDR block you are revoking access to
- **group_id** (*string*) – ID of the EC2 or VPC security group to modify. This is required for VPC security groups and can be used instead of group_name for EC2 security groups

Returns True if successful.

Type bool

2.11.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.12 IcsSQS Common Library

2.12.1 IcsSqs: Library for SQS

This is the IcsSQS common library.

class opslib.icssqs.**IcsSqs** (*region*, ***kwargs*)
ICS Library for SQS

__dict__ = dict_proxy({'__dict__': <attribute '__dict__' of 'IcsSqs' objects>, '__module__': 'opslib.icssqs', '__doc__':

__init__ (*region*, ***kwargs*)

__module__ = 'opslib.icssqs'

__weakref__

list of weak references to the object (if defined)

create_queue (*name*, *visibility_timeout=None*)

Create an SQS Queue.

Parameters

- **name** (*string*) – The name of the new queue. Names are scoped to an account and need to be unique within that account.
- **visibility_timeout** (*int*) – The default visibility timeout for all messages written in the queue.

Returns The newly created queue

Type boto.sqs.queue.Queue

delete_queue (*name*)

Delete the queue

Parameters **name** (*string*) – According to the given name to delete the queue

Returns The result of this action

Type bool

get_queues (*name*='')

If name is empty, it will get all queues, else it retrieves the queue with the given name.

Parameters **name** (*string*) – The name of the queue to retrieve.

Returns The requested queue(list of queues), or None if no match was found

Type boto.sqs.queue.Queue or None or list of boto.sqs.queue.Queue instances

2.12.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.13 IcsUtils.IcsAlert Common Library

2.13.1 IcsAlert: Library for Alert

This is the IcsAlert common library.

class opslib.icsutils.icalert.**IcsAlert** (*region, topic, msg_prefix*='', **kwargs)

__dict__ = dict_proxy({'sendHealthAlert': <function sendHealthAlert at 0x4b29230>, '__module__': 'opslib.icsutils.ics'})

__init__ (*region, topic, msg_prefix*='', **kwargs)

Generate and send alerts when exceptions occur

Parameters

- **region** (*str*) – indicate region name you need to generate SNS alerts
- **topic** (*str*) – indicate topic name you have already setup in SNS
- **msg_prefix** (*str*) – prefix of alert message

__module__ = 'opslib.icsutils.icalert'

__weakref__

list of weak references to the object (if defined)

sendAlert (*msg*, *subj_result*)
Send SNS message as an Alert

Parameters

- **msg** (*str*) – message of the alert
- **subj_result** (*str*) – SUCCESS/ERROR described in subject content

sendHealthAlert (*msg*, *subj*)
Send SNS message as an Alert

Parameters

- **msg** (*str*) – message of the alert
- **subj** (*str*) – subject of message

2.13.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.14 IcsUtils.Misc Common Library

2.14.1 Misc: Library for Misc

This is the Misc common library.

class `opslib.icsutils.misc.IcsSysCfg`
ICS Library for System Configuration

`__del__` ()

`__dict__` = `dict_proxy`({'__module__': 'opslib.icsutils.misc', 'parse_sshpub': <function parse_sshpub at 0x3b58f50>, '_

`__init__` ()

`__module__` = 'opslib.icsutils.misc'

`__weakref__`

list of weak references to the object (if defined)

parse_sshpub (*sshpub=None*)
Parse SSH Public Key

Parameters **sshpub** (*string*) – SSH public key contents

recurmatch (*path*)
Get augeas output via recursive solution (low-level)

Parameters **path** (*string*) – augeas string

to_dict (*data*)
Convert augeas output as dict format (low-level)

Parameters **data** (*generator*) – augeas output data

Return type dict

Returns a dictionary contains augeas output data

update_cfg (*path*, *value*)

Update configuration via augeas library (low-level)

Parameters

- **path** (*string*) – augeas path
- **value** (*string*) – value

update_host_name (*hostname*)

Update hostname

Parameters **hostname** (*string*) – hostname need to update

update_hosts_file (*ipaddr*, *hostname*)

Update /etc/hosts with (ipaddr, hostname) pair

Parameters

- **ipaddr** (*string*) – IP address
- **hostname** (*string*) – hostname

update_sshpub (*user*='ics-user', *sshpub*=None)

Update SSH Public Key

Parameters

- **user** (*string*) – to determine which user to update
- **sshpub** (*string*) – SSH public key contents

class opslib.icsutils.misc.**Retry** (*tries*, *interval*=0, *exceptions*=None)

__call__ (*f*)

retry calling your function

__dict__ = dict_proxy({'__module__': 'opslib.icsutils.misc', 'default_exceptions': <type 'exceptions.Exception'>, '__doc__': 'Decorator for retrying function if exception occurs'})

__init__ (*tries*, *interval*=0, *exceptions*=None)

Decorator for retrying function if exception occurs

Parameters

- **tries** (*int*) – how many times to retry your function
- **exceptions** (*Exception*) – which exceptions you need to be caught to retry your function

__module__ = 'opslib.icsutils.misc'

__weakref__

list of weak references to the object (if defined)

default_exceptions

alias of Exception

opslib.icsutils.misc.**check_error** (*response_data*)

A helper function that prints out the error message recieved in the response_data and raises an error when there is an error.

opslib.icsutils.misc.**clean_empty_items** (*dict_json*)

Clean Empty Items in the Dictionary: None, {}, [], ""

opslib.icsutils.misc.**convert_keyname** (*obj*)

`opslib.icsutils.misc.dict_merge(a, b)`

recursively merges dict's. not just simple a['key'] = b['key'], if both a and b have a key whose value is a dict then `dict_merge` is called on both values and the result stored in the returned dictionary.

`opslib.icsutils.misc.drop_null_items(obj)`

`opslib.icsutils.misc.exec_shell(cmd)`

Execute Shell Commands

(not support for pipe in shell command)

Parameters `cmd` (*string*) – shell commands

Return type tuple

Returns a tuple containing (exitstatus, stdout, stderr)

`opslib.icsutils.misc.exec_shell_pipe(cmd)`

Execute Shell Commands

Parameters `cmd` (*string*) – shell commands

Return type string or bool

Returns False or error reason

`opslib.icsutils.misc.fetch_used_params(service_name, cmd_name, params)`

Fetch used parameters from the whole configuration

`opslib.icsutils.misc.filter_resource_from_json(names=None, filter=None, raw_data=None)`

Filter the resource with specified filter on JSON data

Parameters

- **names** (*list*) – specify the list of resource names to filter Ex: ["AutoScalingGroupName", "LaunchConfigurationName"]
- **filter** (*dict*) – describe the filter in details
- **raw_data** (*dict*) – resource data in JSON format

Return type list

Returns a list containing all the names for filtered resources

`opslib.icsutils.misc.format_aws_tags(resource_id, tags)`

Refine the format of tags under AWS tag syntax

Parameters

- **resource_id** (*string*) – AWS Resource Id
- **tags** (*list*) – a list of tags to refine

`opslib.icsutils.misc.gen_timestamp(format='%Y%m%d-%H%M%S')`

`opslib.icsutils.misc.get_data(search_path)`

Get the complete data paths under the path to search

Parameters `search_path` (*string*) – describes the folder path to search

`opslib.icsutils.misc.get_search_file(name, search_paths)`

Get the complete data path matched with the specified name

Parameters

- **name** (*string*) – specified name need to match

- **search_paths** (*list*) – a list of folder paths to search

`opslib.icsutils.misc.get_search_files(search_paths)`

Get the complete data paths under all the folder paths to search

Parameters **search_paths** (*list*) – a list of folder paths to search

`opslib.icsutils.misc.get_search_path(search_paths)`

Return the complete folder path used when searching for data files.

Parameters **search_paths** (*list*) – a list of folder paths to search

`opslib.icsutils.misc.get_userdata(version='latest', url='http://169.254.169.254', time-out=None, num_retries=5)`

Returns the instance userdata as a string by default.

If the timeout is specified, the connection to the specified url will time out after the specified number of seconds.

Parameters

- **version** (*str*) – API version from AWS
- **timeout** (*int*) – socket timeout
- **num_retries** (*int*) – how many times of retrying

Return type string

Returns a string containing user data

`opslib.icsutils.misc.init_botocore_service(name, region)`

Initialize the proper service with botocore

`opslib.icsutils.misc.is_valid_ip(ip_address)`

Check Validity of an IP address

`opslib.icsutils.misc.keyname_format(fp_json)`

Convert the key name of JSON data: from camel case to a “pythonic” name.

Parameters **fp_json** (*object*) – opened file object

`opslib.icsutils.misc.keyname_formatd(dict_json)`

Convert the key name of JSON data: from camel case to a “pythonic” name.

Parameters **dict_json** (*dict*) – JSON data

`opslib.icsutils.misc.keyname_formats(str_json)`

Convert the key name of JSON data: from camel case to a “pythonic” name.

Parameters **str_json** (*string*) – JSON data

`opslib.icsutils.misc.operate(service, cmd, kwargs)`

A helper function that universally calls any command by taking in the service, name of the command, and any additional parameters required in the call.

`opslib.icsutils.misc.traverse_json(data, delimiter='/', path='', output=None)`

Traverse all the items in JSON data

Parameters

- **data** (*dict, list, element*) – JSON data
- **delimiter** (*string*) – path delimiter for each JSON node
- **path** (*string*) – record the parent path on this JSON data
- **output** (*dict*) – record each item with full path

```
opslib.icsutils.misc.user_data_decode (user_data)
```

2.14.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.15 IcsUtils.Daemon Common Library

2.15.1 Daemon: Library for Daemon

This is the Daemon common library.

```
class opslib.icsutils.daemon.Daemon (pidfile,          stdin='/dev/null',          stdout='/dev/null',
                                     stderr='/dev/null')
```

A generic daemon class.

Usage: subclass the Daemon class and override the run() method

```
__init__ (pidfile, stdin='/dev/null', stdout='/dev/null', stderr='/dev/null')
```

```
__module__ = 'opslib.icsutils.daemon'
```

```
daemonize ()
```

do the UNIX double-fork magic, see Stevens' "Advanced Programming in the UNIX Environment" for details (ISBN 0201563177) http://www.erlenstar.demon.co.uk/unix/faq_2.html#SEC16

```
delpid ()
```

```
restart ()
```

Restart the daemon

```
run ()
```

You should override this method when you subclass Daemon. It will be called after the process has been daemonized by start() or restart().

```
start ()
```

Start the daemon

```
stop ()
```

Stop the daemon

2.15.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.16 IcsUtils.CLI Common Library

Pythonic command-line interface parser that will make you smile.

- <http://docopt.org>
- Repository and issue-tracker: <https://github.com/docopt/docopt>
- Licensed under terms of MIT license (see LICENSE-MIT)
- Copyright (c) 2013 Vladimir Keleshev, vladimir@keleshev.com

`opslib.icsutils.cli.docopt` (*doc*, *argv=None*, *help=True*, *version=None*, *options_first=False*)
Parse *argv* based on command-line interface described in *doc*.

docopt creates your command-line interface based on its description that you pass as *doc*. Such description can contain `–options`, `<positional-argument>`, `commands`, which could be [optional], (required), (mutually | exclusive) or repeated...

Parameters

doc [str] Description of your command-line interface.

argv [list of str, optional] Argument vector to be parsed. `sys.argv[1:]` is used if not provided.

help [bool (default: True)] Set to False to disable automatic help on `-h` or `–help` options.

version [any object] If passed, the object will be printed if `–version` is in *argv*.

options_first [bool (default: False)] Set to True to require options precede positional arguments, i.e. to forbid options and positional arguments intermix.

Returns

args [dict] A dictionary, where keys are names of command-line elements such as e.g. `“–verbose”` and `“<path>”`, and values are the parsed values of those elements.

Example

```
>>> from docopt import docopt
>>> doc = '''
... Usage:
...     my_program tcp <host> <port> [--timeout=<seconds>]
...     my_program serial <port> [--baud=<n>] [--timeout=<seconds>]
...     my_program (-h | --help | --version)
...
... Options:
...     -h, --help  Show this screen and exit.
...     --baud=<n>  Baudrate [default: 9600]
... '''
>>> argv = ['tcp', '127.0.0.1', '80', '--timeout', '30']
>>> docopt(doc, argv)
{'--baud': '9600',
 '--help': False,
 '--timeout': '30',
 '--version': False,
 '<host>': '127.0.0.1',
 '<port>': '80',
 'serial': False,
 'tcp': True}
```

See also

- For video introduction see <http://docopt.org>

•Full documentation is available in README.rst as well as online at <https://github.com/docopt/docopt#readme>

2.16.1 Indices and tables

- *genindex*
- *modindex*
- *search*

2.17 IcsUtils.JsonCli Common Library

2.17.1 JsonCli: Library for CLI based on JSON

This is the JsonCli common library.

```
class opslib.icsutils.jsoncli.OrderNamespace (**kwargs)
    Namespace with Order: from argparse.Namespace

    __init__ (**kwargs)
    __module__ = 'opslib.icsutils.jsoncli'
    __order__ = OrderedDict()
    __setattr__ (attr, value)

opslib.icsutils.jsoncli.add_arguments (group, args)
    Add Arguments to CLI

opslib.icsutils.jsoncli.parse_args (args)
    Create the Command Line Interface

    Parameters args (dict) – describes the command structure for the CLI

opslib.icsutils.jsoncli.recursive_parser (parser, args)
    Recursive CLI Parser
```

2.17.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.18 IcsUtils.JsonSubs Common Library

2.18.1 JsonSubs: Library for JSON Template Substitutions

This is the JsonSubs common library.

```
class opslib.icsutils.jsonsubs.DefaultFunc
    Default Functions for JsonSubs Library
```

```
__dict__ = dict_proxy({'func_join': <staticmethod object at 0x4c86b40>, '__module__': 'opslib.icsutils.jsonsubs', '__d
__module__ = 'opslib.icsutils.jsonsubs'
```

```
__weakref__
    list of weak references to the object (if defined)
```

static func_base64 (*params*)

Builtin Func: Base64

```
{
    "$<Base64>": "xxxxxxxx"
}
```

static func_circle_select (*index, params*)

Builtin Func: CSelect

```
{
    "$<CSelect>": [ 5, [ "apples", "grapes", "mangoes" ] ]
}
```

static func_include (**args*)

Builtin Func: Include

```
{
    "$<Include>": [ "a.json", "b.json" ]
}
```

static func_join (*delimiter, params*)

Builtin Func: Join

```
{
    "$<Join>": [ "-", [ "a", "b", "c" ] ]
}
```

static func_select (*index, params*)

Builtin Func: Select

```
{
    "$<Select>": [ 1, [ "apples", "grapes", "mangoes" ] ]
}
```

class opslib.icsutils.jsonsubs.JsonSubs

```
__dict__ = dict_proxy({'__module__': 'opslib.icsutils.jsonsubs', 'merge_dict': <function merge_dict at 0x4c95758>, '__
```

```
__init__ ()
    JSON Template Substitution Library
```

Detail Examples:

```
>>> from opslib.icsutils.jsonsubs import JsonSubs
>>> engine = JsonSubs()
>>> test_data = [ "-", [ "a", "b", "c" ] ]
>>> print engine.builtin["Join"](test_data)
a-b-c
```

```
>>> import opslib
>>> from opslib.icsutils.jsonsubs import JsonSubs
>>> engine = JsonSubs()
>>> from opslib.icsec2 import IcsEc2
```

```
>>> ec2 = IcsEc2("us-west-2")
>>> func = ec2.conn.get_all_instances
>>> engine.register_builtin({"GetAllInstances": func})
>>> print engine.builtin["GetAllInstances]()
[ Reservation:r-be459c8c,
  Reservation:r-e6822ed4,
  Reservation:r-e66dc3d4,
  Reservation:r-1608a124,
  Reservation:r-ce0da4fc ]

>>> import opslib
>>> from opslib.icsutils.jsonsubs import JsonSubs
>>> engine = JsonSubs()
>>> from opslib.icsec2 import IcsEc2
>>> ec2 = IcsEc2("us-west-2")
>>> func = ec2.get_instance_tags
>>> engine.register_builtin({"GetInstanceTags": func})
>>> default_json = {
...     "version": "1.2.0",
...     "Ids": [ "25d6d811", "12345678" ],
...     "RegionMaps": {
...         "us-east-1": { "32": "ami-xx", "64": "ami-yy" },
...         "us-west-2": { "32": "ami-xy", "64": "ami-yx" }
...     }
... }
>>> output_json = {
...     "Version": "${version}",
...     "Id": { "$<Select>": [0, "${Ids}"] },
...     "UserData": { "$<Base64>": "${Id}" },
...     "Maps": "${RegionMaps}",
...     "InstanceId": { "$<Join>": [ "-", [ "i", "${Id}" ] ] },
...     "Tags": { "$<GetInstanceTags>": "${InstanceId}" }
... }
>>> instance_json = output_json
>>> print engine.tplsub(output_json, instance_json, default_json)
{
  "Id": "25d6d811",
  "InstanceId": "i-25d6d811",
  "Maps": {
    "us-east-1": {
      "32": "ami-xx",
      "64": "ami-yy"
    },
    "us-west-2": {
      "32": "ami-xy",
      "64": "ami-yx"
    }
  },
  "Tags": "ec2-50-112-231-217.us-west-2.compute.amazonaws.com",
  "UserData": "MjVkJmQ4MTE=",
  "Version": "1.2.0"
}
```

Notes:

- Do NOT define Variables in *output_json*, otherwise, please merge *output_json* into *instance_json* or *default_json*
- Variables should be defined in *instance_json* or *default_json*

- If no such *instance_json* and *default_json*, take *output_json* also as *instance_json*
- If no such Variables found in *instance_json*, use one in *default_json*

__module__ = 'opslib.icsutils.jsonsubs'

__weakref__

list of weak references to the object (if defined)

do_sub (*value*, *instance_vars*=None, *default_vars*=None)

Execute JSON Template Substitutions

format_str (*str_value*, *new_str*)

static fp_to_json (*fp*, ***kwargs*)

func_mapping (*map_name*, **args*)

Builtin Func: Mapping

```
{
    "$<Mapping>": [ "MapName", "TopLevelKey", ... ]
}
```

merge_dict (*key*, *instance_vars*=None, *default_vars*=None)

merge_list (*key*, *instance_vars*=None, *default_vars*=None)

merge_map (*key*, *instance_vars*=None, *default_vars*=None)

merge_str (*key*, *instance_vars*=None, *default_vars*=None)

pattern (*esc*='\$')

register_builtin (*customized_func*)

Register Customized Functions

Parameters *customized_func* (*dict*) – describes customized functions to register

```
{
    "Join": DefaultFunc.func_join,
    "FindAMI": DefaultFunc.func_find_ami,
    "GetInstanceTags": IcsEc2.get_instance_tags
}
```

search (*value*, *esc*='\$')

static strip_comment (*data*)

tplsub (*value*, *instance_vars*=None, *default_vars*=None)

Entry for JSON Template Substitution (dict)

tplsub_dict (*dict_value*, *instance_vars*=None, *default_vars*=None)

tplsub_func (*func*, *value*, *instance_vars*=None, *default_vars*=None)

tplsub_list (*list_value*, *instance_vars*=None, *default_vars*=None)

tplsub_str (*str_value*, *instance_vars*=None, *default_vars*=None)

tplsubs (*output_fp*, *instance_fp*=None, *default_fp*=None)

Entry for JSON Template Substitution (fp)

type_of (*value*)

update_str (*str_value*, *new_str*, *instance_vars*=None, *default_vars*=None)

2.18.2 Indices and tables

- *genindex*
- *modindex*
- *search*

2.19 IcsUtils.JsonDiff Common Library

2.19.1 JsonDiff: Library for JSON DIFF

This is the JsonDiff common library.

```
class opslib.icsutils.jsondiff.Comparator(fp1=None,fp2=None,include=[],exclude=[],ignore_add=False)
```

Main workhorse for JSON Comparator

```
__dict__ = dict_proxy({'__module__': 'opslib.icsutils.jsondiff', '_filter_results': <function _filter_results at 0x4831d70>})
```

```
__init__(fp1=None,fp2=None,include=[],exclude=[],ignore_add=False)
```

Parameters

- **fp1** (*object*) – file object (opened with read permission)
- **fp2** (*object*) – file object (opened with read permission)
- **include** (*list*) – a list of attributes to include in the comparison
- **exclude** (*list*) – a list of attributes to exclude in the comparison
- **ignore_add** (*bool*) – whether to ignore the added items in the comparison

Example:

```
>>> from opslib.icsutils.jsondiff import Comparator
>>> import json
>>> old_json = {
...     "name": "opslib",
...     "version": "1.2.0",
...     "members": {
...         "role": "ops",
...         "group": [ "ops", "devops" ]
...     }
... }
>>> new_json = {
...     "name": "opslib",
...     "version": "1.3.0",
...     "members": {
...         "role": "devops",
...         "group": [ "devops" ]
...     }
... }
>>> json.dump(old_json, open("old.json", "w"))
>>> json.dump(new_json, open("new.json", "w"))
>>> fp_old = open("old.json", "r")
>>> fp_new = open("new.json", "r")
>>> engine = Comparator(fp_old, fp_new)
>>> res = engine.compare_dicts()
```



```
>>> print json.dumps(res, sort_keys=True, indent=4)
{
  "members": {
    "group": {
      "0": {
        "+++": "devops",
        "---": "ops"
      },
      "1": {
        "---": "devops"
      }
    },
    "role": {
      "+++": "devops",
      "---": "ops"
    }
  },
  "version": {
    "+++": "1.3.0",
    "---": "1.2.0"
  }
}
```

__module__ = 'opslib.icsutils.jsondiff'

__weakref__

list of weak references to the object (if defined)

_compare_arrays (*old_arr, new_arr*)

simpler version of `compare_dicts`; just an internal method, because it could never be called from outside.

We have it guaranteed that both `new_arr` and `old_arr` are of type list.

_compare_elements (*old, new*)

Unify decision making on the leaf node level.

_compare_scalars (*old, new, name=None*)

Be careful with the result of this function. Negative answer from this function is really `None`, not `False`, so deciding based on the return value like in

```
if self._compare_scalars(...):
```

leads to wrong answer (it should be `if self._compare_scalars(...) is not None:`)

_filter_results (*result*)

Whole -i or -x functionality. Rather than complicate logic while going through the object's tree we filter the result of plain comparison.

Also clear out unused keys in result

_is_incx_key (*key, value*)

Is this key excluded or not among included ones? If yes, it should be ignored.

compare_dicts (*old_obj=None, new_obj=None*)

The real workhorse

`opslib.icsutils.jsondiff.is_scalar` (*value*)

Primitive version, relying on the fact that JSON cannot contain any more complicated data structures.

2.19.2 Indices and tables

- *genindex*
- *modindex*
- *search*

Indices and tables

- *genindex*
- *modindex*
- *search*

O

- `opslib.icsas`, 13
- `opslib.icsec2`, 5
- `opslib.icselb`, 11
- `opslib.icsexception`, 28
- `opslib.icslog`, 26
- `opslib.icsmeta`, 23
- `opslib.icsr53`, 20
- `opslib.icss3`, 18
- `opslib.icssgroup`, 29
- `opslib.icssns`, 19
- `opslib.icssqs`, 32
- `opslib.icsutils.cli`, 39
- `opslib.icsutils.daemon`, 38
- `opslib.icsutils.icsalert`, 33
- `opslib.icsutils.jsoncli`, 40
- `opslib.icsutils.jsondiff`, 44
- `opslib.icsutils.jsonsubs`, 40
- `opslib.icsutils.misc`, 34