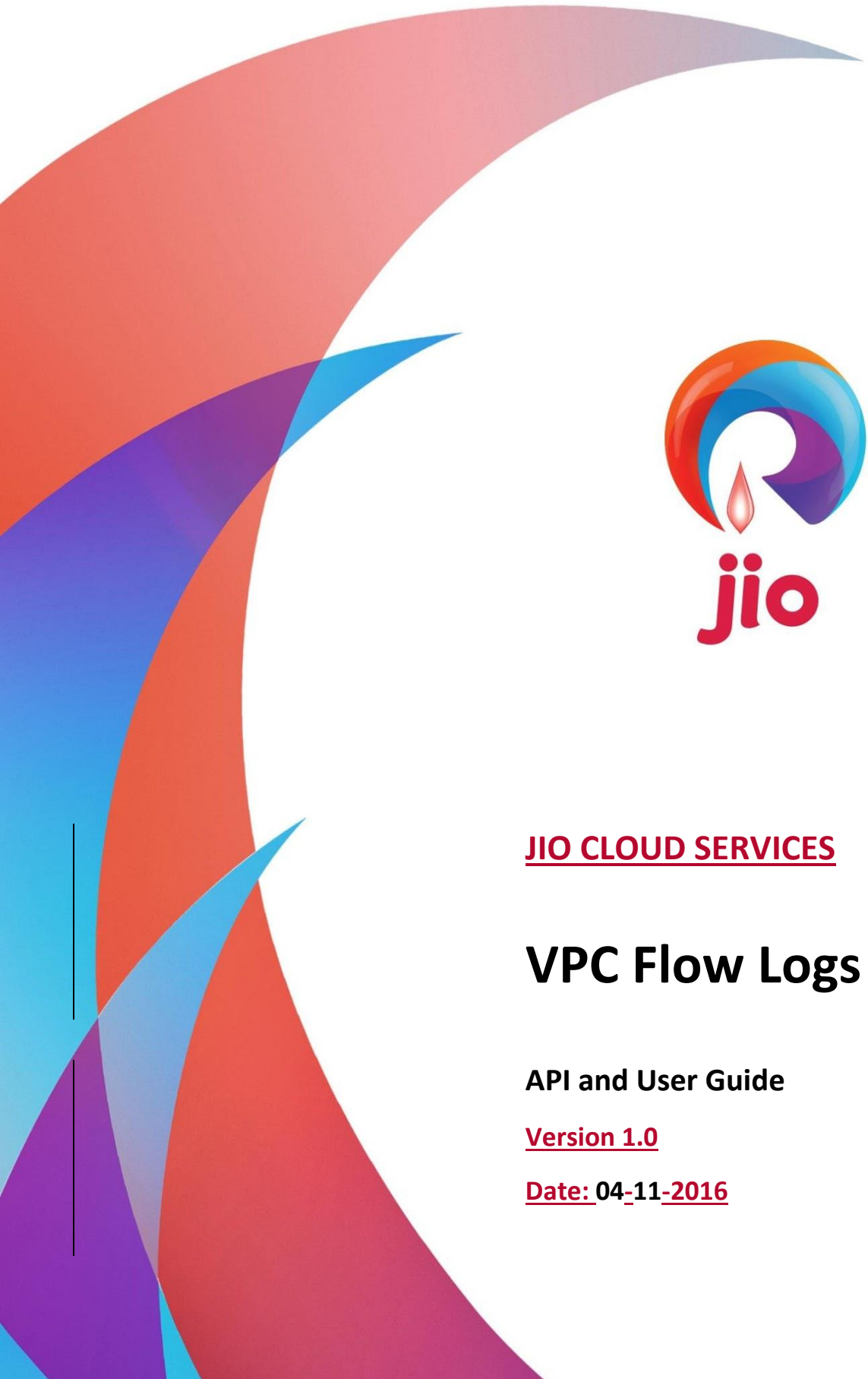

JIO CLOUD SERVICES

VPC Flow Logs feature

API and User Guide

Version 1.0

Date: 04-11-2016

Table of Contents

2 Overview of JCS VPC feature Flow logs 3
2.1 VPC Flow logs Architecture 3
3 API Details 4
3.1 Describe Flow logs 4
Sample request 5
Sample response 5
4 Sample python Client Application to call VPC flow logs API 7
4.1 Installation from source 7
4.2 Configuration 7
4.3 Usage 8

2 Overview of JCS VPC feature Flow logs

JCS flow logs feature gives insight on customers traffic flows inside JCS overlay network. A VPC flow is defined as an L4 (TCP or UDP) session between two VM or between external client (outside JCS) to a VM. These flows are dynamically created and terminated as L4 sessions come and go on a customer VM. Flow information is stored on each compute node of a VPC which is periodically (2 seconds) updated to a cluster of VPC flow log server. We provide JCS API(s) to access this information in REST like API format.

2.1 VPC Flow logs Architecture

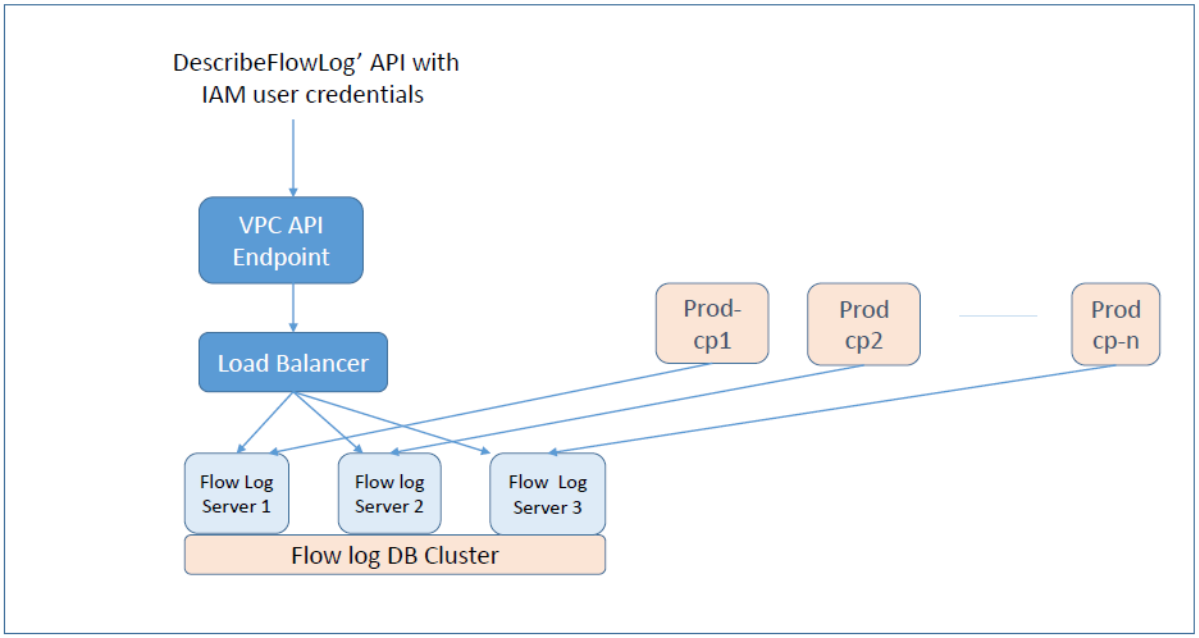

Figure 1: VPC Flow logs feature Architecture

VPC flow logs API are same as other VPC API exposed by VPC service. To call VPC API, first one need to create IAM user by either logging to JCS console or through a manual request to JCS team. Then IAM credentials will be used to encrypt HTTP payload of API requests. We have one python client which can be used to create JCS API requests in prescribed format. It is described later in this section.

There are two layers of security to ensure only valid Admin can look at VPC flow logs of any JCS account. One is provided by IAM authentication where any valid IAM account holder can look at his own account's VPC flow logs. Second is an admin password mapped to Infosec Admin account which enables this account to look at all the JCS account's VPC flow logs.

3 API Details

3.1 Describe Flow logs

Description	Request	Response
Describe Flow logs for any JCS account.	start_time dd-mm-yyyy hr:mm:ss Type – String Required – Yes Condition: valid start time is between now and last 15 days. IST time zone should be used for date. end_time dd-mm-yyyy hr:mm:ss Type – String Required – Yes Condition: Difference between start time and end time should not exceed 30 minutes as response becomes too huge in case of heave traffic. One can make multiple requests in case more than 30 minutes data is required admin_password xxxxxxxx Type – String Required – Optional. Required if you want to query flow logs for other accounts account acc-xxxxxxxx Type – String Required – Optional. direction_ing 0 for ingress 1 for egress Type – Boolean Required – Optional. Must if explicit account is specified	RequestId The ID of the request Type – String Items (flow logs) Items : String

Sample request

https://vpc.ind-west-1.jiocloudservices.com/?Action=DescribeFlowLog
&start_time=”03-11-2016 17:00:00”
&end_time=”03-11-2016 17:05:00”
&admin_password=takemefromrequest
&account=acc-789189089
&AUTHPARAMS

Sample response


```
"DescribeFlowLogResponse": {
    "@xmlns": "http://vpc.ind-west-1.jiocloudservices.com/doc/2016-03-01/",
    "requestId": "req-89787a91-d711-49bd-8673-5bf6bee68ae5",
    "value": {
      "item": [
        {
          "action": "pass",
          "agg-bytes": "842",
          "agg-packets": "4",
          "destip": "10.140.216.121",
          "destvn": "default-domain:services:public",
          "directionIng": "1",
          "dport": "443",
          "nwAceUuid": "00000000-0000-0000-0000-000000000001",
          "protocol": "6",
          "setupTime": "1478172458053492",
          "sgRuleUuid": "bc5b4e1a-6403-4cf7-9b8a-c3e7d105dcf2",
          "sourceip": "10.140.213.62",
          "sourcevn": "default-domain:services:public",
          "sport": "44568",
          "teardownTime": "1478172644373505",
          "underlayProto": "1",
          "underlaySourcePort": "0",
          "uuidKey": "00025d0b-60d6-4d6b-b3dd-5878d6af8b18"
        },
```


Response parameter	Description	Response
items	Get description / information of the VPC Flow logs	action pass or dropped from Security-group Type – String Agg-bytes Total numbers of bytes passed through from inception of flow Type – Integer Agg-packets

		Total numbers of packets passed through from inception of flow Type – Integer destip The Destination IP of the flow Type – String destvn destination network(public internet or customer network if destination is within VPC) Type – String directionIng 1 for egress, 0 for ingress Type – boolean dport L4 destination port Type – number newAceUuid Not valid now protocol L4 protocol in IP header Type – number setupTime Time when first packet for this flow was received Type: number in UST sgRuleUuid: Security group which allowed or dropped this flow Type: String Sourceip Source IP of the flow Type: String Sourcevn Source network(public internet or customer network if destination is within VPC) Type – String Sport Source L4 port in IP header Type: number Teardowntime Time when flow was terminated Type: number in UST underlayProto Payload type of outer tunnel header used in underlay. 1 for IPv4 now Type: Number underlaySourcePort

		Source port of underlay traffic. 0 as of now as we support only IPv4 tunnelling as of now UuidKey Unique Identifier for this flow Type: String

4 Sample python Client Application to call VPC flow logs API

A sample python application has been created to demonstrate usage of VPC flow log API. It can be executed from your VM or laptop with connectivity to RJIL network. Steps to use this utility is same as using JCS CLI.

It is highly advised to use this client when you are collecting flow log stats for ALL JCS ACCOUNTS as ADMIN. As data size for flow stats for whole JCS accounts is really huge (average 10 MB for 2 seconds) so collecting stats for meaningful time interval (say 30 minutes) would result in huge response. This client will take care of breaking large responses into smaller chunks and integrate them in one file.

Steps for using this utility (Same as using JCS CLI)

4.1 Installation from source

```
git clone https://github.com/gdpak/jcsclient.git
cd jcsclient
sudo pip install -r requirements.txt
sudo python setup.py develop
```

4.2 Configuration

Copy openrc.sample to create openrc file, put your actual credentials in this file and then source this file.

```
cp openrc.sample openrc
# Update openrc now, and add your access/secret keys
# If you are admin user then get admin_password from us after providing us your
account id.
# following should be set in your openrc
# export ACCESS_KEY=(ACCESS_KEY_FOR_YOUR_ACCOUT)
# export SECRET_KEY=(SECRET_KEY_FOR_YOUR_ACCOUNT)
# export VPC_URL=https://vpc.ind-west-1.internal.jiocloudservices.com
# export VPC_FLOW_LOGS_ADMIN_PASS="YOUR_ADMIN_PASSWORD"
source openrc
```

4.3 Usage

```
cd tests

vagrant@dpak-db3:~/jcsclient/tests$ ./vpc_flow_logs_ut.py -h
Usage: vpc_flow_logs_ut.py -s <start_time> -e <end_time>
Example - ./vpc_flow_logs_ut.py -s "08-11-2016 11:30:00" -e "08-11-2016
11:31:00"

-h or --help            help

Mandatory parameters:
-s              start_time              start_time in IST timezone format
'dd-mm-yyyy hh:mm:ss'
-e              end_time                end_time   in IST timezone format
'dd-mm-yyyy hh:mm:ss'

Optional parameters:
-a              Account id
-d              Select direction. Options are 1 for ingress 0 for egress. Required
for query on account_id
-o              Select Output file
```

4.4 Sample output

```
vagrant@dpak-db3:~/jcsclient/tests$ ./vpc_flow_logs_ut.py -s "08-11-2016 16:30:00"
-e "08-11-2016 16:35:02"

 VPC Flow Logs Collection Starting at 08-11-2016 13:20:51 Local

 9% percent completed
 19% percent completed
 29% percent completed
 39% percent completed
 49% percent completed
 59% percent completed
 69% percent completed
 79% percent completed
 89% percent completed
 99% percent completed
 VPC Flow Logs collection Finished at 08-11-2016 13:34:23 Local

 Results are stored at - my_vpc_flow_log.txt
```