

INSTANCES

Screenshot of the AWS EC2 Instances page showing a list of running instances. The instance RIAK 1 is selected.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
RIAK 1	i-03a9c25e9a513e3e6	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
KONG/JUM...	i-093ac3bbf3c5fdaaf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 2	i-095c52ca36bd7484d	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 3	i-0abd48f9dea74eebf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 4	i-0d139840edbfb50b53	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
nat	i-0efffcfcbb6hf12704	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-54-145-145-75

Instance: i-03a9c25e9a513e3e6 (RIAK 1) Private IP: 12.0.1.9

Description Status Checks Monitoring Tags Usage Instructions

Instance ID: i-03a9c25e9a513e3e6	Public DNS (IPv4): -
Instance state: running	IPv4 Public IP: -
Instance type: t2.micro	IPv6 IPs: -
Elastic IPs:	Private DNS: ip-12-0-1-9.ec2.internal
Availability zone: us-east-1f	Private IPs: 12.0.1.9

Screenshot of the AWS EC2 Instances page showing a list of running instances. The instance RIAK 2 is selected.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
RIAK 1	i-03a9c25e9a513e3e6	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
KONG/JUM...	i-093ac3bbf3c5fdaaf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 2	i-095c52ca36bd7484d	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 3	i-0abd48f9dea74eebf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 4	i-0d139840edbfb50b53	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
nat	i-0efffcfcbb6hf12704	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-54-145-145-75

Instance: i-095c52ca36bd7484d (RIAK 2) Private IP: 12.0.1.122

Description Status Checks Monitoring Tags Usage Instructions

Instance ID: i-095c52ca36bd7484d	Public DNS (IPv4): -
Instance state: running	IPv4 Public IP: -
Instance type: t2.micro	IPv6 IPs: -
Elastic IPs:	Private DNS: ip-12-0-1-122.ec2.internal
Availability zone: us-east-1f	Private IPs: 12.0.1.122

https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#instances:instanceState=running;sort=instanceId

Rohan Kamat N. Virginia Support

EC2 Dashboard Events Tags Reports Limits

INSTANCES Instances Launch Templates Spot Requests Reserved Instances Dedicated Hosts Scheduled Instances Capacity Reservations

IMAGES AMIs Bundle Tasks

ELASTIC BLOCK STORE

Feedback English (US)

Type here to search

Launch Instance Connect Actions

Instance State : Running Add filter

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
RIAK 1	i-03a9c25e9a513e3e6	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
KONG/JUM...	i-093ac3bbf3c5fdaf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 2	i-095c52ca36bd7484d	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 3	i-0abd48f9dea74eebf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 4	i-0d139840edbf50b53	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
nat	i-0efffcfchh6bfdf2704	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-54-145-145-7

Instance state: running
Instance type: t2.micro
Elastic IPs:
Availability zone: us-east-1f
Security groups: default, Riak KV 2-2 Series-2-2-AutogenByAWSMP-, view inbound rules, view outbound rules
Scheduled events: No scheduled events
AMI ID: riak - 2.2.0-e99e7c96-8cc2-437e-b151-a5ebd502c1d5-ami-55f5de42.3
IPv4 Public IP: -
IPv6 IPs: -
Private DNS: ip-12-0-1-165.ec2.internal
Private IPs: 12.0.1.165
Secondary private IPs:
VPC ID: vpc-085ded1dd06a861b0
Subnet ID: subnet-0fe7ac249b194dcff

Feedback English (US)

Type here to search

Launch Instance Connect Actions

Instance State : Running Add filter

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
RIAK 1	i-03a9c25e9a513e3e6	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
KONG/JUM...	i-093ac3bbf3c5fdaf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 2	i-095c52ca36bd7484d	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 3	i-0abd48f9dea74eebf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 4	i-0d139840edbf50b53	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
nat	i-0efffcfchh6bfdf2704	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-54-145-145-7

Availability zone: us-east-1f
Security groups: default, Riak KV 2-2 Series-2-2-AutogenByAWSMP-, view inbound rules, view outbound rules
Scheduled events: No scheduled events
AMI ID: riak - 2.2.0-e99e7c96-8cc2-437e-b151-a5ebd502c1d5-ami-55f5de42.3
Platform: -
IAM role: -
Private IPs: 12.0.1.148
Secondary private IPs:
VPC ID: vpc-085ded1dd06a861b0
Subnet ID: subnet-0fe7ac249b194dcff
Network interfaces: eth0
Source/dest. check: True

Feedback English (US)

Type here to search

Launch Instance Connect Actions

Instance State : Running Add filter

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
RIAK 1	i-03a9c25e9a513e3e6	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
KONG/JUM...	i-093ac3bbf3c5fdaf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 2	i-095c52ca36bd7484d	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 3	i-0abd48f9dea74eebf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 4	i-0d139840edbf50b53	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
nat	i-0efffcfchh6bfdf2704	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-54-145-145-7

Availability zone: us-east-1f
Security groups: default, Riak KV 2-2 Series-2-2-AutogenByAWSMP-, view inbound rules, view outbound rules
Scheduled events: No scheduled events
AMI ID: riak - 2.2.0-e99e7c96-8cc2-437e-b151-a5ebd502c1d5-ami-55f5de42.3
Platform: -
IAM role: -
Private IPs: 12.0.1.148
Secondary private IPs:
VPC ID: vpc-085ded1dd06a861b0
Subnet ID: subnet-0fe7ac249b194dcff
Network interfaces: eth0
Source/dest. check: True

https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#instances:instanceState=running;sort=instanceId

Rohan Kamat N. Virginia Support

EC2 Dashboard Events Tags Reports Limits

INSTANCES Instances Launch Templates Spot Requests Reserved Instances Dedicated Hosts Scheduled Instances Capacity Reservations

IMAGES AMIs Bundle Tasks

ELASTIC BLOCK STORE

Feedback English (US)

Type here to search

Launch Instance Connect Actions

Instance State : Running Add filter

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
RIAK 1	i-03a9c25e9a513e3e6	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
KONG/JUM...	i-093ac3bbf3c5fdaf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 2	i-095c52ca36bd7484d	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 3	i-0abd48f9dea74eebf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 4	i-0d139840edbf50b53	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
nat	i-0efffcfchh6bfdf2704	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-54-145-145-7

Availability zone: us-east-1f
Security groups: default, Riak KV 2-2 Series-2-2-AutogenByAWSMP-, view inbound rules, view outbound rules
Scheduled events: No scheduled events
AMI ID: riak - 2.2.0-e99e7c96-8cc2-437e-b151-a5ebd502c1d5-ami-55f5de42.3
Platform: -
IAM role: -
Private IPs: 12.0.1.148
Secondary private IPs:
VPC ID: vpc-085ded1dd06a861b0
Subnet ID: subnet-0fe7ac249b194dcff
Network interfaces: eth0
Source/dest. check: True

Feedback English (US)

Type here to search

Launch Instance Connect Actions

Instance State : Running Add filter

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
RIAK 1	i-03a9c25e9a513e3e6	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
KONG/JUM...	i-093ac3bbf3c5fdaf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 2	i-095c52ca36bd7484d	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 3	i-0abd48f9dea74eebf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 4	i-0d139840edbf50b53	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
nat	i-0efffcfchh6bfdf2704	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-54-145-145-7

Availability zone: us-east-1f
Security groups: default, Riak KV 2-2 Series-2-2-AutogenByAWSMP-, view inbound rules, view outbound rules
Scheduled events: No scheduled events
AMI ID: riak - 2.2.0-e99e7c96-8cc2-437e-b151-a5ebd502c1d5-ami-55f5de42.3
Platform: -
IAM role: -
Private IPs: 12.0.1.148
Secondary private IPs:
VPC ID: vpc-085ded1dd06a861b0
Subnet ID: subnet-0fe7ac249b194dcff
Network interfaces: eth0
Source/dest. check: True

https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#Instances:instanceState=running;sort=instanceId

Rohan Kamat N. Virginia Support

EC2 Dashboard Events Tags Reports Limits Instances Launch Templates Spot Requests Reserved Instances Dedicated Hosts Scheduled Instances Capacity Reservations Images AMIs Bundle Tasks Elastic Block Store

Launch Instance Connect Actions

Instance State : Running Add filter

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
KONG/JUM...	i-093ac3bbf3c5daaf	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-34-239-180-96
RIAK 2	i-095c52ca36bd7484d	t2.micro	us-east-1f	running	2/2 checks ...	None	
RIAK 3	i-0abd48f9dea74eef	t2.micro	us-east-1f	running	2/2 checks ...	None	
RIAK 4	i-0d139840edb50b53	t2.micro	us-east-1f	running	2/2 checks ...	None	
nat	i-0ef0fcbbb6bf2704	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-54-145-145-21
RIAK 5	i-0f04d4f4ad510a61b	t2.micro	us-east-1f	running	2/2 checks ...	None	

Instance state: running
Instance type: t2.micro
Elastic IPs
Availability zone: us-east-1f
Security groups: default, Riak KV 2.2 Series-2-2-AutogenByAWSMP-, view inbound rules, view outbound rules
Scheduled events: No scheduled events
AMI ID: riak - 2.2.0-e99e7c96-8cc2-437e-b5f1-5e0b5f02a1d6 ami E5E5-0a12
VPC ID: vpc-085ded1dd06a861b0
Subnet ID: subnet-0fe7ac249b194dcff

Feedback English (US) Type here to search Privacy Policy Terms of Use 1:51 AM 11/22/2018

Kong/JUMPBOX

https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#Instances:instanceState=running;sort=instanceId

Rohan Kamat N. Virginia Support

EC2 Dashboard Events Tags Reports Limits Instances Launch Templates Spot Requests Reserved Instances Dedicated Hosts Scheduled Instances Capacity Reservations Images AMIs Bundle Tasks Elastic Block Store

Launch Instance Connect Actions

Instance State : Running Add filter

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
RIAK 1	i-03a9c25e9a513e3e6	t2.micro	us-east-1f	running	2/2 checks ...	None	
RIAK 2	i-095c52ca36bd7484d	t2.micro	us-east-1f	running	2/2 checks ...	None	
RIAK 3	i-0abd48f9dea74eef	t2.micro	us-east-1f	running	2/2 checks ...	None	
RIAK 4	i-0d139840edb50b53	t2.micro	us-east-1f	running	2/2 checks ...	None	
nat	i-0ef0fcbbb6bf2704	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-54-145-145-21
KONG/JUM...	i-093ac3bbf3c5daaf (KONG/JUMPBOX)	Public DNS: ec2-34-239-180-96.compute-1.amazonaws.com					

Description Status Checks Monitoring Tags

Instance ID: i-093ac3bbf3c5daaf
Public DNS (IPv4): ec2-34-239-180-96.compute-1.amazonaws.com
Instance state: running
IPv4 Public IP: 34.239.180.96
Instance type: t2.micro
IPv6 IPs: -
Elastic IPs
Private DNS: ip-12-0-0-46.ec2.internal
Availability zone: us-east-1f
Private IPs: 12.0.0.46

Feedback English (US) Type here to search Privacy Policy Terms of Use 1:51 AM 11/22/2018

AWS SUBNET SETUP

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR
subnet-053ff43b	vpc-c0bc6fba	available	172.31.48.0/20	4091	-	
Public subnet	subnet-0d2116e3c5aede58	available	vpc-085ded1dd06a861b0 ...	12.0.0.0/24	249	-
Private subnet	subnet-0fe7ac249b194dcff	available	vpc-085ded1dd06a861b0 ...	12.0.1.0/24	246	-
	subnet-5999cc13	available	vpc-c0bc6fba	172.31.16.0/20	4091	-
	subnet-6743c33b	available	vpc-c0bc6fba	172.31.32.0/20	4091	-
	subnet-b04d34bf	available	vpc-c0bc6fba	172.31.64.0/20	4091	-
	subnet-b1911d6	available	vpc-c0bc6fba	172.31.0.0/20	4091	-
	subnet-d2b230fc	available	vpc-c0bc6fba	172.31.80.0/20	4091	-

RIAK START ON EACH INSTANCE

```
ec2-user@ip-12-0-1-189:~$ 
[ECDSA key fingerprint is MD5:02:98:52:d6:eb:42:db:aa:e0:a7:99:a5:32:4c:2f:22.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '12.0.1.9' (ECDSA) to the list of known hosts.
Last login: Wed Nov 16 16:59:22 2016 from ec2-54-208-134-181.compute-1.amazonaws.com

[ec2-user@ip-12-0-1-189 ~]$ sudo riak start
[ec2-user@ip-12-0-1-189 ~]$ exit
logout
Connection to 12.0.1.9 closed.
[ec2-user@ip-12-0-0-174 ~]$ ssh -i cmpe281virginia.pem ec2-user@12.0.1.122
The authenticity of host '12.0.1.122 (12.0.1.122)' can't be established.
ECDSA key fingerprint is SHA256:Id+o4dgVvj4f0r8cDhKqXuAobBda3PSIUWWL8w5r5w.
ECDSA key fingerprint is MD5:b1:05:9c:61:0ce7:00:b3:e4:bc:fbe:97:84:59:8c.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '12.0.1.122' (ECDSA) to the list of known hosts.
Last login: Wed Nov 16 16:59:22 2016 from ec2-54-208-134-181.compute-1.amazonaws.com

[ec2-user@ip-12-0-1-189 ~]$ 
[ec2-user@ip-12-0-1-189 ~]$ 

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-189 ~]$ sudo riak start
[ec2-user@ip-12-0-1-189 ~]$ exit
logout
Connection to 12.0.1.122 closed.
[ec2-user@ip-12-0-0-174 ~]$ ssh -i cmpe281virginia.pem ec2-user@12.0.1.48
The authenticity of host '12.0.1.48 (12.0.1.48)' can't be established.
ECDSA key fingerprint is SHA256:UXl1rm2leTktSngdmkh1PTsRyEqvGlbLMFSpYgttE.
ECDSA key fingerprint is MD5:6c:bc:0d:8f:f2:a:cd:b9:7d:d5:0b:9e:cc:d8:c4:f3.
Are you sure you want to continue connecting (yes/no)? yes
[ec2-user@ip-12-0-0-174 ~]$ 
```

```
ec2-user@ip-12-0-1-189:~$ https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-165 ~]$ sudo riak-admin cluster join riak@12.0.1.9
Node did not respond to ping!
[ec2-user@ip-12-0-1-165 ~]$ sudo riak start
[ec2-user@ip-12-0-1-165 ~]$ sudo riak-admin cluster join riak@12.0.1.9
Join failed. Try again in a few moments.
[ec2-user@ip-12-0-1-165 ~]$ sudo riak-admin cluster join riak@12.0.1.9
Success: staged join request for 'riak@12.0.1.165' to 'riak@12.0.1.9'
[ec2-user@ip-12-0-1-165 ~]$ exit
logout
Connection to 12.0.1.165 closed.
[ec2-user@ip-12-0-0-174 ~]$ ssh -i cmpe281virginia.pem ec2-user@12.0.1.48
Last login: Thu Nov 22 04:10:27 2018 from ip-12-0-0-174.ec2.internal
      _\   _/ 
     _\ \_ /  Amazon Linux AMI
      _\ \_ |_ |

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-48 ~]$ sudo riak-admin cluster join riak@12.0.1.9
Success: staged join request for 'riak@12.0.1.48' to 'riak@12.0.1.9'
[ec2-user@ip-12-0-1-48 ~]$ exit
logout
Connection to 12.0.1.48 closed.
[ec2-user@ip-12-0-0-174 ~]$ ssh -i cmpe281virginia.pem ec2-user@12.0.1.189
Last login: Thu Nov 22 04:10:48 2018 from ip-12-0-0-174.ec2.internal
      _\   _/ 
     _\ \_ /  Amazon Linux AMI
      _\ \_ |_ |

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-189 ~]$ sudo riak-admin cluster join riak@12.0.1.9
Success: staged join request for 'riak@12.0.1.189' to 'riak@12.0.1.9'
[ec2-user@ip-12-0-1-189 ~]$
```

Windows Type here to search 8:16 PM 11/21/2018

```
ec2-user@ip-12-0-1-189:~$ Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-189 ~]$ sudo riak-admin cluster join riak@12.0.1.9
Success: staged join request for 'riak@12.0.1.189' to 'riak@12.0.1.9'
[ec2-user@ip-12-0-1-189 ~]$ sudo riak-admin cluster plan
----- Staged Changes -----
Action       Details(s)
join        'riak@12.0.1.122'
join        'riak@12.0.1.165'
join        'riak@12.0.1.189'
join        'riak@12.0.1.48'

NOTE: Applying these changes will result in 1 cluster transition
#####
# After cluster transition 1/1
#####

----- Membership -----
Status      Ring    Pending   Node
valid      0.0%    20.3%    'riak@12.0.1.122'
valid      0.0%    20.3%    'riak@12.0.1.165'
valid      0.0%    20.3%    'riak@12.0.1.189'
valid      0.0%    20.3%    'riak@12.0.1.48'
valid     100.0%   18.8%    'riak@12.0.1.9'

Valid:5 / Leaving:0 / Exiting:0 / Joining:0 / Down:0

Transfers resulting from cluster changes: 52
 13 transfers from 'riak@12.0.1.9' to 'riak@12.0.1.122'
 13 transfers from 'riak@12.0.1.9' to 'riak@12.0.1.165'
 13 transfers from 'riak@12.0.1.9' to 'riak@12.0.1.48'
 13 transfers from 'riak@12.0.1.9' to 'riak@12.0.1.189'

[ec2-user@ip-12-0-1-189 ~]$ sudo riak-admin cluster status
--- Cluster Status ---
Ring ready: true

+-----+-----+-----+-----+
| node      |status | avail |ring |pending|
+-----+-----+-----+-----+
```

Windows Type here to search 8:17 PM 11/21/2018

```

ec2-user@ip-12-0-1-189:~ 
valid      0.0%    20.3%  'riak@12.0.1.122'
valid      0.0%    20.3%  'riak@12.0.1.165'
valid      0.0%    20.3%  'riak@12.0.1.189'
valid      0.0%    20.3%  'riak@12.0.1.48'
valid   100.0%   18.8%  'riak@12.0.1.9'
-----
Valid:5 / Leaving:0 / Exiting:0 / Joining:0 / Down:0
Transfers resulting from cluster changes: 52
 13 transfers from 'riak@12.0.1.9' to 'riak@12.0.1.122'
 13 transfers from 'riak@12.0.1.9' to 'riak@12.0.1.165'
 13 transfers from 'riak@12.0.1.9' to 'riak@12.0.1.48'
 13 transfers from 'riak@12.0.1.9' to 'riak@12.0.1.189'
[ec2-user@ip-12-0-1-189 ~]$ sudo riak-admin cluster status
---- Cluster Status ----
Ring ready: true
+-----+-----+-----+-----+
|     node | status | avail | ring | pending |
+-----+-----+-----+-----+
| riak@12.0.1.122 | joining! | up | 0.0% | -- |
| riak@12.0.1.165 | joining! | up | 0.0% | -- |
| riak@12.0.1.189 | joining! | up | 0.0% | -- |
| riak@12.0.1.48 | joining! | up | 0.0% | -- |
| (C) riak@12.0.1.9 | valid | up | 100.0% | -- |
+-----+-----+-----+-----+
Key: (C) = Claimant; availability marked with '!' is unexpected
[ec2-user@ip-12-0-1-189 ~]$ sudo riak-admin cluster commit
Cluster changes committed
[ec2-user@ip-12-0-1-189 ~]$ sudo riak-admin member status
===== Membership =====
Status  Ring  Pending  Node
-----+
valid   3.1%  20.3%  'riak@12.0.1.122'
valid   3.1%  20.3%  'riak@12.0.1.165'
valid   3.1%  20.3%  'riak@12.0.1.189'
valid   3.1%  20.3%  'riak@12.0.1.48'
valid   87.5% 18.8%  'riak@12.0.1.9'
-----
Valid:5 / Leaving:0 / Exiting:0 / Joining:0 / Down:0
[ec2-user@ip-12-0-1-189 ~]$ 

```

Kong

```

ec2-user@ip-12-0-0-46:~ 
[ec2-user@ip-12-0-0-46 ~]$ sudo docker images
REPOSITORY          TAG        IMAGE ID       CREATED
SIZE
cassandra           2.2       3d3a7ec01a18   5 days ago
494MB
kong                0.9.9     dd3d1f1dd40f   10 months ago
334MB
[ec2-user@ip-12-0-0-46 ~]$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS              PORTS              NAMES
a07514f3bf6        kong:0.9.9        "/docker-entrypoint..."   51 seconds ago
Up 50 seconds        0.0.0.0:7946->7946/tcp, 0.0.0.0:8000-8001->8000-8001/tcp
cp_0.0.0.0:8443->8443/tcp, 0.0.0.0:7946->7946/udp kong
87a706d3aa7b        cassandra:2.2     "docker-entrypoint.s..." About a minute
ago
Up About a minute   7000-7001/tcp, 7199/tcp, 9042/tcp, 9160/tcp, 0.0.0.0:902->9402/tcp kong-database
[ec2-user@ip-12-0-0-46 ~]$ sudo docker ps --all --format "table {{.ID}}\t{{.Name}}
\t{{.Image}}\t{{.Status}}"
CONTAINER ID        NAMES               IMAGE            STATUS
a07514f3bf6        kong               kong:0.9.9      Up About a minute
87a706d3aa7b        kong-database     cassandra:2.2  Up About a minute
[ec2-user@ip-12-0-0-46 ~]$ 

```

API REGISTRATION ON KONG

The screenshot shows the Postman application interface. In the top navigation bar, 'File', 'Edit', 'View', 'Help', 'New', 'Import', 'Runner', and 'My Workspace' are visible. The 'Collections' tab is selected. A POST request is being made to `http://34.239.180.96:8001/apis`. The request body is a JSON object:

```
1 [ {  
2   "name": "node1",  
3   "request_path": "/node1",  
4   "strip_request_path": "true",  
5   "preserve_host": "true",  
6   "upstream_url": "http://12.0.1.9:8098/"  
7 }]
```

The response status is 201 Created, time 224 ms, size 414 B. The JSON response body is:

```
1 {  
2   "upstream_url": "http://12.0.1.9:8098/",  
3   "strip_request_path": true,  
4   "request_path": "/node1",  
5   "id": "579bdb27-a9be-4ace-862f-4cf63b914a96",  
6   "created_at": 1542865529000,  
7   "preserve_host": true,  
8   "name": "node1"  
9 }
```

The screenshot shows the Postman application interface. In the top navigation bar, 'File', 'Edit', 'View', 'Help', 'New', 'Import', 'Runner', and 'My Workspace' are visible. The 'Collections' tab is selected. A POST request is being made to `http://34.239.180.96:8001/apis`. The request body is a JSON object:

```
1 [ {  
2   "name": "node2",  
3   "request_path": "/node2",  
4   "strip_request_path": "true",  
5   "preserve_host": "true",  
6   "upstream_url": "http://12.0.1.122:8098/"  
7 }]
```

The response status is 201 Created, time 214 ms, size 416 B. The JSON response body is:

```
1 {  
2   "upstream_url": "http://12.0.1.122:8098/",  
3   "strip_request_path": true,  
4   "request_path": "/node2",  
5   "id": "7b777f3c-24c6-4630-8801-4a6b7e7567cf",  
6   "created_at": 1542865726000,  
7   "preserve_host": true,  
8   "name": "node2"  
9 }
```

Postman

File Edit View Help

New Import Runner

My Workspace Invite Upgrade

History Collections Trash

POST http://34.239.180.96:8001/apis

Send Save

No Environment

Body Cookies Headers (6) Test Results Status: 201 Created Time: 184 ms Size: 416 B Save Download

Pretty Raw Preview JSON

```
1 [ {  
2   "name": "node3",  
3   "request_path": "/node3",  
4   "strip_request_path": "true",  
5   "preserve_host": "true",  
6   "upstream_url": "http://12.0.1.165:8098/"  
7 } ]
```

Build Browse

Type here to search

9:53 PM 11/21/2018

Postman

File Edit View Help

New Import Runner

My Workspace Invite Upgrade

History Collections Trash

POST http://34.239.180.96:8001/apis

Send Save

No Environment

Body Cookies Headers (6) Test Results Status: 201 Created Time: 227 ms Size: 415 B Save Download

Pretty Raw Preview JSON

none form-data x-www-form-urlencoded raw binary JSON (application/json)

```
1 [ {  
2   "name": "node4",  
3   "request_path": "/node4",  
4   "strip_request_path": "true",  
5   "preserve_host": "true",  
6   "upstream_url": "http://12.0.1.48:8098/"  
7 } ]
```

Build Browse

Type here to search

9:53 PM 11/21/2018

Postman

File Edit View Help

New Import Runner

My Workspace Invite Upgrade

History Collections Trash

POST http://34.239.180.96:8001/apis

Send Save

Body Cookies Headers (6) Test Results

Status: 409 Conflict Time: 306 ms Size: 311 B Save Download

Pretty Raw Preview JSON

```
1 {  
2   "name": "nodes",  
3   "request_path": "/node5",  
4   "strip_request_path": "true",  
5   "preserve_host": "true",  
6   "upstream_url": "http://12.0.1.189:8098/"  
7 }
```

Body Cookies Headers (6) Test Results

Status: 409 Conflict Time: 306 ms Size: 311 B Save Download

Pretty Raw Preview JSON

```
1 {  
2   "name": "already exists with value 'nodes'",  
3   "request_path": "/node5"  
4 }
```

Type here to search

9:53 PM 11/21/2018

This screenshot shows the Postman application interface. In the center, there's a request panel with a POST method and the URL 'http://34.239.180.96:8001/apis'. Below the URL, the request body is displayed as a JSON object. The response section shows a 409 Conflict status with a detailed error message: 'already exists with value 'nodes''. On the left sidebar, there's a 'Collections' tab where a collection named 'midterm' is selected, showing 8 requests. The bottom of the screen shows a Windows taskbar with various pinned icons.

Ping

Postman

File Edit View Help

New Import Runner

My Workspace Invite Upgrade

History Collections Trash

GET ping Node1

Send Save

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION	... Bulk Edit
Key	Value	Description	

Body Cookies Headers (8) Test Results

Status: 200 OK Time: 291 ms Size: 270 B Save Download

Pretty Raw Preview HTML

```
i 1 ok
```

Type here to search

11:07 PM 11/21/2018

This screenshot shows the Postman application interface again. This time, a GET request is made to 'http://34.239.180.96:8000/node1/ping'. The response is a 200 OK status with the body 'ok'. The left sidebar shows a 'Collections' tab with a collection named 'Personal Project' selected, containing 10 requests. The bottom of the screen shows a Windows taskbar with various pinned icons.

Postman

File Edit View Help

New Import Runner

My Workspace Invite Upgrade

ping node 2

GET http://34.239.180.96:8000/node2/ping

Send Save

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION
Key	Value	Description

Status: 200 OK Time: 171 ms Size: 270 B Save Download

Pretty Raw Preview HTML

i 1 OK

Build Browse

Type here to search

Node1 Node2 node3 node4 node5 ping Node1 ping node 2 ping node 3 ping node 4 ping node 5

10 requests

Personal Project

11:08 PM 11/21/2018

This screenshot shows the Postman application window. The left sidebar lists various API endpoints under 'Personal Project'. The main panel displays a successful GET request to 'http://34.239.180.96:8000/node2/ping' with a status of 200 OK. The response body is shown as 'OK'. The bottom status bar indicates the time as 11:08 PM on 11/21/2018.

Postman

File Edit View Help

New Import Runner

My Workspace Invite Upgrade

ping node 3

GET http://34.239.180.96:8000/node3/ping

Send Save

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION
Key	Value	Description

Status: 200 OK Time: 381 ms Size: 270 B Save Download

Pretty Raw Preview HTML

i 1 OK

Build Browse

Type here to search

Node1 Node2 node3 node4 node5 ping Node1 ping node 2 ping node 3 ping node 4 ping node 5

10 requests

Personal Project

11:08 PM 11/21/2018

This screenshot shows the Postman application window. The left sidebar lists various API endpoints under 'Personal Project'. The main panel displays a successful GET request to 'http://34.239.180.96:8000/node3/ping' with a status of 200 OK. The response body is shown as 'OK'. The bottom status bar indicates the time as 11:08 PM on 11/21/2018.

Postman

File Edit View Help

New Import Runner

My Workspace Invite

No Environment

ping node 4

GET http://34.239.180.96:8000/node4/ping

Send Save

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION
Key	Value	Description

Status: 200 OK Time: 209 ms Size: 268 B Save Download

Pretty Raw Preview HTML

i 1 OK

Build Browse

Type here to search

11:08 PM 11/21/2018

This screenshot shows the Postman application interface. In the center, there's a request builder for a GET method to the URL 'http://34.239.180.96:8000/node4/ping'. The 'Body' tab is selected. Below the request, the response status is shown as 'Status: 200 OK' with a duration of 'Time: 209 ms' and a size of 'Size: 268 B'. At the bottom of the main window, there are tabs for 'Pretty', 'Raw', 'Preview', and 'HTML'. The status bar at the bottom right indicates the time as '11:08 PM' and the date as '11/21/2018'.

Postman

File Edit View Help

New Import Runner

My Workspace Invite

No Environment

ping node 5

GET http://34.239.180.96:8000/node5/ping

Send Save

Pretty Raw Preview HTML

i 1 OK

Build Browse

Type here to search

11:41 PM 11/21/2018

This screenshot shows the Postman application interface, similar to the first one but with a different request. It's a GET method to 'http://34.239.180.96:8000/node5/ping'. The 'Body' tab is selected. Below the request, the response status is shown as 'Status: 200 OK' with a duration of 'Time: 209 ms' and a size of 'Size: 268 B'. At the bottom of the main window, there are tabs for 'Pretty', 'Raw', 'Preview', and 'HTML'. The status bar at the bottom right indicates the time as '11:41 PM' and the date as '11/21/2018'.

Bucket properties

The screenshot shows the Postman application interface. In the top navigation bar, there are tabs for 'File', 'Edit', 'View', 'Help', 'New', 'Import', 'Runner', and 'My Workspace'. Below the navigation bar, there is a toolbar with various icons for different HTTP methods (GET, POST, PUT, DELETE, etc.). The main workspace shows a 'GET' request being sent to the URL `http://34.239.180.96:8000/node1/types/subjects/buckets/cmpe281/props`. The response body is displayed in a code editor-like view with syntax highlighting, showing a JSON object with numerous fields. On the left side, there is a sidebar with a 'History' section containing several recent requests, a 'Collections' section, and a 'Trash' section.

Set Bucket Properties

The screenshot shows the Postman application interface. In the top navigation bar, there are tabs for 'File', 'Edit', 'View', 'Help', 'New', 'Import', 'Runner', and 'My Workspace'. Below the navigation bar, there is a toolbar with various icons for different HTTP methods (GET, POST, PUT, DELETE, etc.). The main workspace shows a 'PUT' request being sent to the URL `http://34.239.180.96:8000/node1/types/subjects/buckets/cmpe281/props`. The request body contains the JSON object `{"props": {"allow_mult": false}}`. On the left side, there is a sidebar with a 'History' section containing several recent requests, a 'Collections' section, and a 'Trash' section. At the bottom of the screen, the Windows taskbar is visible, showing various pinned icons and the system tray.

KEY INSERTION

The screenshot shows the Postman application interface. In the center, there is a POST request window with the URL `http://34.239.180.96:8000/node1/types/subjects/buckets/cmpe281/keys/lab1`. The request body contains the JSON `{ "Task": "AWS essentials" }`. Below the request, the response status is `204 No Content`, time `2086 ms`, and size `323 B`. On the left sidebar, under the 'Collections' tab, there is a list of requests including `GET node 5`, `GET ping Node1`, `GET ping node 2`, `GET ping node 3`, `GET ping node 4`, `GET ping node 5`, `GET Get Bucket properties`, `GET set bucket properties`, and `GET Add key from node 1`. A collection named `Starbucks Test Collection` is also listed. At the bottom of the screen, the Windows taskbar shows various pinned icons.

Get keys from each node

The screenshot shows the Postman application interface. In the center, there is a GET request window with the URL `http://34.239.180.96:8000/node1/types/subjects/buckets/cmpe281/keys/lab1`. The request body contains the JSON `{ "Task": "AWS essentials" }`. Below the request, the response status is `200 OK`, time `186 ms`, and size `537 B`. On the left sidebar, under the 'Collections' tab, there is a list of requests including `GET ping node 2`, `GET ping node 3`, `GET ping node 4`, `GET ping node 5`, `GET Get Bucket properties`, `GET set bucket properties`, `GET Add key from node 1`, `GET node1 key`, `GET node2 key`, `GET node3 key`, `GET node 4 key`, `GET node 5 key`, and `Starbucks Test Collection`. A collection named `node1 key` is selected. At the bottom of the screen, the Windows taskbar shows various pinned icons.

Postman

File Edit View Help

New Import Runner

My Workspace Invite Upgrade

node2 key

GET http://34.239.180.96:8000/node2/types/subjects/buckets/cmpe281/keys/lab1

Send Save Examples (0)

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (14) Test Results Status: 200 OK Time: 122 ms Size: 538 B Save Download

Pretty Raw Preview Auto

1 {Task:"AWS essentials"}

Starbucks Test Collection 0 requests

Type here to search

12:00 AM 11/22/2018

This screenshot shows the Postman application interface. The left sidebar displays a list of requests under the 'Collections' tab, including various ping and key retrieval requests. The main workspace shows a successful GET request to 'http://34.239.180.96:8000/node2/types/subjects/buckets/cmpe281/keys/lab1'. The response body is a JSON object with a single key 'Task' containing the value 'AWS essentials'. The status bar at the bottom indicates the request was successful (200 OK) with a response time of 122 ms and a size of 538 B.

Postman

File Edit View Help

New Import Runner

My Workspace Invite Upgrade

node3 key

GET http://34.239.180.96:8000/node3/types/subjects/buckets/cmpe281/keys/lab1

Send Save Examples (0)

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (14) Test Results Status: 200 OK Time: 123 ms Size: 538 B Save Download

Pretty Raw Preview Auto

1 {Task:"AWS essentials"}

Starbucks Test Collection 0 requests

Type here to search

12:00 AM 11/22/2018

This screenshot shows the Postman application interface, identical to the one above but with a different URL. It displays a successful GET request to 'http://34.239.180.96:8000/node3/types/subjects/buckets/cmpe281/keys/lab1'. The response body is a JSON object with a single key 'Task' containing the value 'AWS essentials'. The status bar at the bottom indicates the request was successful (200 OK) with a response time of 123 ms and a size of 538 B.

Postman

File Edit View Help

New Import Runner

My Workspace Invite Upgrade

node 4 key

GET http://34.239.180.96:8000/node4/types/subjects/buckets/cmpe281/keys/lab1

Send Save Examples (0)

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (14) Test Results Status: 200 OK Time: 118 ms Size: 538 B Save Download

Pretty Raw Preview Auto

```
1 {Task:"AWS essentials"}
```

Starbucks Test Collection 0 requests

Type here to search

12:00 AM 11/22/2018

This screenshot shows the Postman application interface. The left sidebar displays a collection of requests, including several GET requests to nodes 2 through 5 and their respective keys. The main panel shows a successful GET request to 'node4 key' with a status of 200 OK. The response body is a JSON object with a single key-value pair: 'Task' set to 'AWS essentials'. The bottom status bar indicates the time as 12:00 AM on 11/22/2018.

Postman

File Edit View Help

New Import Runner

My Workspace Invite Upgrade

node 5 key

GET http://34.239.180.96:8000/node5/types/subjects/buckets/cmpe281/keys/lab1

Send Save Examples (0)

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (14) Test Results Status: 200 OK Time: 121 ms Size: 538 B Save Download

Pretty Raw Preview Auto

```
1 {Task:"AWS essentials"}
```

Starbucks Test Collection 0 requests

Type here to search

12:00 AM 11/22/2018

This screenshot shows the Postman application interface, identical to the one above but with a different URL. It displays a successful GET request to 'node5 key' with a status of 200 OK. The response body is a JSON object with a single key-value pair: 'Task' set to 'AWS essentials'. The bottom status bar indicates the time as 12:00 AM on 11/22/2018.

Dropping IP tables to create Network Partition:

The screenshot shows a terminal window with several tabs open at the top: 'pylit.py', 'docker networking.md', 'docker commands', 'mysql db name : mysql', '1. docker desktop >> enable kubernetes', 'LAB 9 PART 2', and 'RIAK 1 12.0.1.9'. The terminal itself has two main sections of text.

Section 1 (Top):

```
32      -p 8000:8000 \
33      -p 8443:8443 \
34      -p 8001:8001 \
35      -p 7946:7946 \
36      -p 7946:7946/udp \
37      kong:0.9.9
38
39 Register on Postman
40
41 Create bucket on any node which will be propagated to all
42 sudo riak-admin bucket-type create subjects
43
44 sudo riak-admin bucket-type activate subjects
45
46 Blocking node 1,2,3 on node 4 and 5 :
47 sudo iptables -I INPUT -s 12.0.1.9 -j DROP
48 sudo iptables -I INPUT -s 12.0.1.122 -j DROP
49 sudo iptables -I INPUT -s 12.0.1.165 -j DROP
50 sudo iptables -I INPUT -s 12.0.1.48 -j DROP
51 sudo iptables -I INPUT -s 12.0.1.189 -j DROP
52
53 Unblock Traffic :
54 sudo iptables -D INPUT -s 12.0.1.9 -j DROP
55 sudo iptables -D INPUT -s 12.0.1.122 -j DROP
56 sudo iptables -D INPUT -s 12.0.1.165 -j DROP
57 sudo iptables -D INPUT -s 12.0.1.48 -j DROP
58 sudo iptables -D INPUT -s 12.0.1.189 -j DROP
```

Section 2 (Bottom):

```
[ec2-user@ip-12-0-1-189:~]
[ec2-user@ip-12-0-1-189 ~] [ \_\_!_ ]
https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-189 ~]$ sudo iptables -I INPUT -s 12.0.1.9 -j DROP
[ec2-user@ip-12-0-1-189 ~]$ sudo iptables -I INPUT -s 12.0.1.122 -j DROP
[ec2-user@ip-12-0-1-189 ~]$ sudo iptables -I INPUT -s 12.0.1.165 -j DROP
[ec2-user@ip-12-0-1-189 ~]$ exit
logout
Connection to 12.0.1.48 closed.
[ec2-user@ip-12-0-0-46 ~]$ ssh -i cmpe28virginia.pem ec2-user@12.0.1.189
Last login: Thu Nov 22 09:46:16 2018 from ip-12-0-0-46.ec2.internal
[ec2-user@ip-12-0-0-46 ~] [ \_\_!_ ] Amazon Linux AMI
[ec2-user@ip-12-0-1-189:~]
[ec2-user@ip-12-0-1-189 ~]$ sudo iptables -I INPUT -s 12.0.1.122 -j DROP
[ec2-user@ip-12-0-1-189 ~]$ sudo iptables -I INPUT -s 12.0.1.165 -j DROP
[ec2-user@ip-12-0-1-189 ~]$
```

The screenshot shows a terminal window titled "ec2-user@ip-12-0-1-122:~". The session starts with a curl command to check for updates on Amazon Linux. It then proceeds to run several sudo commands to manage Riak buckets and iptables rules. Finally, it connects to a Postman instance running on port 8080.

```
curl -s https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/ | grep '27 package(s) needed for security, out of 79 available' &gt;> /dev/null
sudo yum update
curl -s https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/ | grep 'Amazon Linux version 2018.03 is available.' &gt;> /dev/null
sudo riak-admin bucket-type create subjects
sudo riak-admin bucket-type activate subjects
sudo iptables -I INPUT -s 12.0.1.9 -j DROP
sudo iptables -I INPUT -s 12.0.1.122 -j DROP
sudo iptables -I INPUT -s 12.0.1.165 -j DROP
sudo iptables -I INPUT -s 12.0.1.48 -j DROP
sudo iptables -I INPUT -s 12.0.1.189 -j DROP
sudo iptables -D INPUT -s 12.0.1.9 -j DROP
sudo iptables -D INPUT -s 12.0.1.122 -j DROP
sudo iptables -D INPUT -s 12.0.1.165 -j DROP
sudo iptables -D INPUT -s 12.0.1.48 -j DROP
sudo iptables -D INPUT -s 12.0.1.189 -j DROP
curl -s https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/ | grep '27 package(s) needed for security, out of 79 available' &gt;> /dev/null
curl -s https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/ | grep 'Amazon Linux version 2018.03 is available.' &gt;> /dev/null
curl -s https://postman-echo.com/get | grep 'kong:0.9.9' &gt;> /dev/null
```

File Edit Selection Find View Goto Tools Project Preferences Help

pylift.py docker networking.md docker commands mysql db name: mysql 1. docker desktop >> enable kubernetes LAB 9 PART 2 RIAK 1 12.0.1.9

```

32     -p 8000:8000 \
33     -p 8443:8443 \
34     -p 8001:8001 \
35     -p 7946:7946 \
36     -p 7946:7946/udp \
37     kong:0.9.9
38
39 Register on Postman
40
41 Create bucket on any node which will be propagated to all
42 sudo riak-admin bucket-type create subjects
43
44 sudo riak-admin bucket-type activate subjects
45
46 Blocking node 1,2,3 on node 4 and 5 :
47 sudo iptables -I INPUT -s 12.0.1.9 -j DROP
48 sudo iptables -I INPUT -s 12.0.1.122 -j DROP
49 sudo iptables -I INPUT -s 12.0.1.165 -j DROP
50 sudo iptables -I INPUT -s 12.0.1.48 -j DROP
51 sudo iptables -I INPUT -s 12.0.1.189 -j DROP
52
53 Unblock Traffic :
54 sudo iptables -D INPUT -s 12.0.1.9 -j DROP
55 sudo iptables -D INPUT -s 12.0.1.122 -j DROP
56 sudo iptables -D INPUT -s 12.0.1.165 -j DROP
57 sudo iptables -D INPUT -s 12.0.1.48 -j DROP
58 sudo iptables -D INPUT -s 12.0.1.189 -j DROP

```

ec2-user@ip-12-0-1-165:~

```

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-122 ~]$ sudo iptables -I INPUT -s 12.0.1.48 -j DROP
[ec2-user@ip-12-0-1-122 ~]$ sudo iptables -I INPUT -s 12.0.1.189 -j DROP
[ec2-user@ip-12-0-1-122 ~]$ exit
logout
Connection to 12.0.1.122 closed.
[ec2-user@ip-12-0-0-46 ~]$ ssh -i cmpe28virginia.pem ec2-user@12.0.1.165
Last login: Thu Nov 22 09:47:05 2018 from ip-12-0-0-46.ec2.internal
[ec2-user@ip-12-0-1-165 ~]$ 

```

Amazon Linux AMI

```

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-165 ~]$ sudo iptables -I INPUT -s 12.0.1.48 -j DROP
[ec2-user@ip-12-0-1-165 ~]$ sudo iptables -I INPUT -s 12.0.1.189 -j DROP
[ec2-user@ip-12-0-1-165 ~]$ 

```

2 lines, 88 characters selected

Type here to search Tab Size: 4 Plain Text 11:55 AM 11/22/2018

Cluster Status

File Edit Selection Find View Goto Tools Project Preferences Help

pylift.py docker networking.md docker commands mysql db name: mysql 1. docker desktop >> enable kubernetes LAB 9 PART 2 RIAK 1 12.0.1.9

```

1 RIAK 1 12.0.1.9
2 RIAK 2 12.0.1.122
3 RIAK 3 12.0.1.165
4 RIAK 4 12.0.1.48
5 RIAK 5 12.0.1.189
6 KONG 18.206.48.111
7 docker ps --all --format "table {{.ID}}\t{{.Names}}\t{{.Image}}\t{{.Status}}"
8 sudo riak-admin cluster status
9
10 sudo riak start (On all Clusters)
11 sudo riak-admin cluster join riak@12.0.1.9
12 sudo riak-admin cluster plan
13 sudo riak-admin cluster status
14 sudo riak-admin cluster commit
15 sudo riak-admin member_status
16
17 Setup Kong To access from Postman
18
19 sudo yum update -y
20 sudo yum install -y docker
21 sudo service docker start
22 sudo usermod -a -G docker ec2-user
23
24 docker network create --driver bridge gumball
25 docker run -d --name kong-database --network gumball -p 9402:9402 cassandra:2.2
26 docker run -d --name kong \
27   --network gumball \
28   -e "KONG_DATABASE=cassandra" \
29   -e "KONG_CASSANDRA_CONTACT_POINTS=kong-database" \
30   -e "KONG_PG_HOST=kong-database" \
31   -p 8000:8000 \
32

```

ec2-user@ip-12-0-1-165:~

node	status	avail	ring	pending
riak@12.0.1.122	valid	up	20.3	--
riak@12.0.1.165	valid	up	20.3	--
riak@12.0.1.189	valid	up	20.3	--
riak@12.0.1.48	valid	down!	20.3	--
(C) riak@12.0.1.9	valid	up	18.8	--

Key: (C) = Claimant; availability marked with '!' is unexpected

---- Cluster Status ----

Ring ready: true

node	status	avail	ring	pending
riak@12.0.1.122	valid	up	20.3	--
riak@12.0.1.165	valid	up	20.3	--
riak@12.0.1.189	valid	down!	20.3	--
riak@12.0.1.48	valid	down!	20.3	--
(C) riak@12.0.1.9	valid	up	18.8	--

Key: (C) = Claimant; availability marked with '!' is unexpected

1:55 AM 11/22/2018

2 lines, 31 characters selected

Type here to search Tab Size: 4 Plain Text

Testing using Postman

The screenshot shows the Postman application interface. On the left, there's a sidebar with a search bar and sections for History, Collections, and Trash. Under History, several GET requests are listed: ping node 2, ping node 3, ping node 4, ping node 5, Get Bucket properties, set bucket properties, Add key from node 1, node1 key, node2 key, node3 key, node 4 key, and node 5 key. A collection named "Starbucks Test Collection" is also visible. The main workspace shows a POST request to the URL `http://34.239.180.96:8000/node1/types/subjects/buckets/cmpe281/keys/lab31`. The Body tab is selected, containing the JSON payload `{Task:"lab31 from node 1"}`. Other tabs include Params, Authorization, Headers, Pre-request Script, Tests, Cookies, and Code. Buttons for Send, Save, and Examples (0) are at the top right. The bottom of the screen shows the Windows taskbar with various pinned icons.

This screenshot shows the same Postman interface after sending the previous POST request. The main workspace now displays a GET request to the same URL. The Body tab is selected, showing the response body which contains the JSON payload `{Task:"lab31 from node 1"}`. The status bar at the bottom indicates a successful 200 OK response with a time of 97 ms and a size of 540 B. Other tabs like Params, Authorization, Headers, and Tests are visible. The bottom of the screen shows the Windows taskbar.

Screenshot of Postman interface showing a collection named "node2 key".

The left sidebar shows a list of requests:

- GET ping node 2
- GET ping node 3
- GET ping node 4
- GET ping node 5
- GET Get Bucket properties
- GET set bucket properties
- GET Add key from node 1
- GET node1 key
- GET node2 key
- GET node3 key
- GET node4 key
- GET node5 key

The main panel shows a GET request to `http://34.239.180.96:8000/node2/types/subjects/buckets/cmpe281/keys/lab31`. The response body is:

```
1 {Task:"lab31 from node 1"}
```

Screenshot of Postman interface showing a collection named "node3 key".

The left sidebar shows a list of requests, identical to the one in the previous screenshot.

The main panel shows a GET request to `http://34.239.180.96:8000/node3/types/subjects/buckets/cmpe281/keys/lab31`. The response body is:

```
1 {Task:"lab31 from node 1"}
```

Screenshot of Postman application showing a collection named "Collections".

The left sidebar shows a list of requests:

- GET ping node 2
- GET ping node 3
- GET ping node 4
- GET ping node 5
- GET Get Bucket properties
- GET set bucket properties
- GET Add key from node 1
- GET node1 key
- GET node2 key
- GET node3 key
- GET node 4 key
- GET node 5 key

A collection named "Starbucks Test Collection" is also listed.

The main panel displays a request for "node 4 key" with the following details:

- Method: GET
- URL: http://34.239.180.96:8000/node4/types/subjects/buckets/cmpe281/keys/lab31
- Params tab (selected):

KEY	VALUE	DESCRIPTION
Key	Value	Description
- Body tab (selected):

Pretty Raw Preview Auto

```
1 not found
2
```
- Test Results: Status: 404 Object Not Found, Time: 82 ms, Size: 292 B

Bottom status bar: Build, Browse, 1:56 AM, 11/22/2018, 17 tabs.

Screenshot of Postman application showing a collection named "Collections".

The left sidebar shows a list of requests, identical to the one in the previous screenshot:

- GET ping node 2
- GET ping node 3
- GET ping node 4
- GET ping node 5
- GET Get Bucket properties
- GET set bucket properties
- GET Add key from node 1
- GET node1 key
- GET node2 key
- GET node3 key
- GET node 4 key
- GET node 5 key

A collection named "Starbucks Test Collection" is also listed.

The main panel displays a request for "node 5 key" with the following details:

- Method: GET
- URL: http://34.239.180.96:8000/node5/types/subjects/buckets/cmpe281/keys/lab31
- Params tab (selected):

KEY	VALUE	DESCRIPTION
Key	Value	Description
- Body tab (selected):

Pretty Raw Preview Auto

```
1 not found
2
```
- Test Results: Status: 404 Object Not Found, Time: 126 ms, Size: 292 B

Bottom status bar: Build, Browse, 1:56 AM, 11/22/2018, 17 tabs.

Test to ensure only new data is available in the ring(Node 1)

The screenshot shows the Postman application interface. On the left, the 'Collections' tab is selected, displaying a list of requests. One request, 'Add key from node 1', is highlighted. The main panel shows a POST request to the URL `http://34.239.180.96:8000/node1/types/subjects/buckets/cmpe281/keys/lab32`. The 'Body' tab is selected, containing the JSON payload `{Task:"lab32 from node 1"}`. The status bar at the bottom indicates the date and time as 11/22/2018 1:57 AM.

Node 4

The screenshot shows the Postman application interface. Similar to the previous one, it displays a list of requests on the left and a POST request to Node 4 in the main panel. The URL is `http://34.239.180.96:8000/node4/types/subjects/buckets/cmpe281/keys/lab32`. The 'Body' tab contains the JSON payload `{Task:"lab32 from node 4"}`. The status bar at the bottom indicates the date and time as 11/22/2018 1:57 AM.

Unblocking Traffic using IP tables

```
File Edit Selection Find View Goto Tools Project Preferences Help
pylit.py docker networking.md docker commands mysql db name :mysql 1.docker desktop >> enable kubernetes LAB 9 PART 2 RIAK 112.0.1.9
38
39 Register on Postman
40
41 Create bucket on any node which will be propagated to all
42 sudo riak-admin bucket-type create subjects
43
44 sudo riak-admin bucket-type activate subjects
45
46 Blocking node 1,2,3 on node 4 and 5 :
47 sudo iptables -I INPUT -s 12.0.1.9 -j DROP
48 sudo iptables -I INPUT -s 12.0.1.122 -j DROP
49 sudo iptables -I INPUT -s 12.0.1.165 -j DROP
50 sudo iptables -I INPUT -s 12.0.1.48 -j DROP
51 sudo iptables -I INPUT -s 12.0.1.189 -j DROP
52
53 Unblock Traffic :
54 sudo iptables -D INPUT -s 12.0.1.9 -j DROP
55 sudo iptables -D INPUT -s 12.0.1.122 -j DROP
56 sudo iptables -D INPUT -s 12.0.1.165 -j DROP
57 sudo iptables -D INPUT -s 12.0.1.48 -j DROP
58 sudo iptables -D INPUT -s 12.0.1.189 -j DROP

[ec2-user@ip-12-0-1-48:~]
|   riak@12.0.1.109 |valid| down! | 20.3| -- |
|   riak@12.0.1.48 |valid| down! | 20.3| -- |
| (C) riak@12.0.1.9 |valid| up | 18.8| -- |
+-----+-----+-----+-----+
Key: (C) = Claimant; availability marked with '!' is unexpected
[ec2-user@ip-12-0-1-165 ~]$ exit
logout
Connection to 12.0.1.165 closed.
[ec2-user@ip-12-0-0-46 ~]$ ssh -i cmpe281virginia.pem ec2-user@12.0.1.48
Last login: Thu Nov 22 09:54:14 2018 from ip-12-0-0-46.ec2.internal
[ec2-user@ip-12-0-0-46 ~]$
[ec2-user@ip-12-0-0-46 ~]$ curl https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
[ec2-user@ip-12-0-0-46 ~]$ curl https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-48 ~]$ sudo iptables -D INPUT -s 12.0.1.9 -j DROP
[ec2-user@ip-12-0-1-48 ~]$ sudo iptables -D INPUT -s 12.0.1.122 -j DROP
[ec2-user@ip-12-0-1-48 ~]$ sudo iptables -D INPUT -s 12.0.1.165 -j DROP
[ec2-user@ip-12-0-1-48 ~]$ exit[ec2-user@ip-12-0-1-48 ~]$ exit
```

```
File Edit Selection Find View Goto Tools Project Preferences Help
pylit.py docker networking.md docker commands mysql db name :mysql 1.docker desktop >> enable kubernetes LAB 9 PART 2 RIAK 1 12.0.1.9

38
39 Register on Postman
40
41 Create bucket on any node which will be propagated to all
42 sudo riak-admin bucket-type create subjects
43
44 sudo riak-admin bucket-type activate subjects
45
46 Blocking node 1,2,3 on node 4 and 5 :
47 sudo iptables -I INPUT -s 12.0.1.9 -j DROP
48 sudo iptables -I INPUT -s 12.0.1.122 -j DROP
49 sudo iptables -I INPUT -s 12.0.1.165 -j DROP
50 sudo iptables -I INPUT -s 12.0.1.48 -j DROP
51 sudo iptables -I INPUT -s 12.0.1.189 -j DROP
52
53 Unblock Traffic :
54 sudo iptables -D INPUT -s 12.0.1.9 -j DROP
55 sudo iptables -D INPUT -s 12.0.1.122 -j DROP
56 sudo iptables -D INPUT -s 12.0.1.165 -j DROP
57 sudo iptables -D INPUT -s 12.0.1.48 -j DROP
58 sudo iptables -D INPUT -s 12.0.1.189 -j DROP

3 lines, 133 characters selected
Type here to search
```

```
[ec2-user@ip-12-0-1-189:~]
https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2016.09 is available.
[ec2-user@ip-12-0-1-48 ~]$ sudo iptables -D INPUT -s 12.0.1.9 -j DROP
[ec2-user@ip-12-0-1-48 ~]$ sudo iptables -D INPUT -s 12.0.1.122 -j DROP
[ec2-user@ip-12-0-1-48 ~]$ sudo iptables -D INPUT -s 12.0.1.165 -j DROP
[ec2-user@ip-12-0-1-48 ~]$ exit
logout
Connection to 12.0.1.48 closed.
[ec2-user@ip-12-0-0-46 ~]$ ssh -i cmpe281virginia.pem ec2-user@12.0.1.189
Last login: Thu Nov 22 09:54:25 2018 from ip-12-0-0-46.ec2.internal
[ec2-user@ip-12-0-1-189 ~] / Amazon Linux AMI
[ec2-user@ip-12-0-1-189 ~]$ sudo iptables -D INPUT -s 12.0.1.9 -j DROP
[ec2-user@ip-12-0-1-189 ~]$ sudo iptables -D INPUT -s 12.0.1.122 -j DROP
[ec2-user@ip-12-0-1-189 ~]$ sudo iptables -D INPUT -s 12.0.1.165 -j DROP
[ec2-user@ip-12-0-1-189 ~]$
```

```
File Edit Selection Find View Goto Tools Project Preferences Help
pylit.py docker networking.md docker commands mysql db name:mysql 1.docker desktop >> enable kubernetes LAB 9 PART 2 RIAK 1 12.0.1.9
38
39 Register on Postman
40
41 Create bucket on any node which will be propagated to all
42 sudo riak-admin bucket-type create subjects
43
44 sudo riak-admin bucket-type activate subjects
45
46 Blocking node 1,2,3 on node 4 and 5 :
47 sudo iptables -I INPUT -s 12.0.1.9 -j DROP
48 sudo iptables -I INPUT -s 12.0.1.122 -j DROP
49 sudo iptables -I INPUT -s 12.0.1.165 -j DROP
50 sudo iptables -I INPUT -s 12.0.1.48 -j DROP
51 sudo iptables -I INPUT -s 12.0.1.189 -j DROP
52
53 Unblock Traffic :
54 sudo iptables -D INPUT -s 12.0.1.9 -j DROP
55 sudo iptables -D INPUT -s 12.0.1.122 -j DROP
56 sudo iptables -D INPUT -s 12.0.1.165 -j DROP
57 sudo iptables -D INPUT -s 12.0.1.48 -j DROP
58 sudo iptables -D INPUT -s 12.0.1.189 -j DROP

ec2-user@ip-12-0-1-122:~$ https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-9 ~]$ sudo iptables -D INPUT -s 12.0.1.48 -j DROP
[ec2-user@ip-12-0-1-9 ~]$ sudo iptables -D INPUT -s 12.0.1.189 -j DROP
[ec2-user@ip-12-0-1-9 ~]$ exit
logout
Connection to 12.0.1.9 closed.
[ec2-user@ip-12-0-0-46 ~]$ ssh -i cmpe281virginia.pem ec2-user@12.0.1.122
Last login: Thu Nov 22 09:54:48 2018 from ip-12-0-0-46.ec2.internal
[ec2-user@ip-12-0-1-122 ~]$ https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-122 ~]$ sudo iptables -D INPUT -s 12.0.1.48 -j DROP
[ec2-user@ip-12-0-1-122 ~]$ sudo iptables -D INPUT -s 12.0.1.189 -j DROP
[ec2-user@ip-12-0-1-122 ~]$ e
```



```
File Edit Selection Find View Goto Tools Project Preferences Help
pylit.py docker networking.md docker commands mysql db name:mysql 1.docker desktop >> enable kubernetes LAB 9 PART 2 RIAK 1 12.0.1.9
38
39 Register on Postman
40
41 Create bucket on any node which will be propagated to all
42 sudo riak-admin bucket-type create subjects
43
44 sudo riak-admin bucket-type activate subjects
45
46 Blocking node 1,2,3 on node 4 and 5 :
47 sudo iptables -I INPUT -s 12.0.1.9 -j DROP
48 sudo iptables -I INPUT -s 12.0.1.122 -j DROP
49 sudo iptables -I INPUT -s 12.0.1.165 -j DROP
50 sudo iptables -I INPUT -s 12.0.1.48 -j DROP
51 sudo iptables -I INPUT -s 12.0.1.189 -j DROP
52
53 Unblock Traffic :
54 sudo iptables -D INPUT -s 12.0.1.9 -j DROP
55 sudo iptables -D INPUT -s 12.0.1.122 -j DROP
56 sudo iptables -D INPUT -s 12.0.1.165 -j DROP
57 sudo iptables -D INPUT -s 12.0.1.48 -j DROP
58 sudo iptables -D INPUT -s 12.0.1.189 -j DROP

ec2-user@ip-12-0-1-165:~$ https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-122 ~]$ sudo iptables -D INPUT -s 12.0.1.48 -j DROP
[ec2-user@ip-12-0-1-122 ~]$ sudo iptables -D INPUT -s 12.0.1.189 -j DROP
[ec2-user@ip-12-0-1-122 ~]$ exit
logout
Connection to 12.0.1.122 closed.
[ec2-user@ip-12-0-0-46 ~]$ ssh -i cmpe281virginia.pem ec2-user@12.0.1.165
Last login: Thu Nov 22 09:54:58 2018 from ip-12-0-0-46.ec2.internal
[ec2-user@ip-12-0-1-165 ~]$ https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
27 package(s) needed for security, out of 79 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-12-0-1-165 ~]$ sudo iptables -D INPUT -s 12.0.1.48 -j DROP
[ec2-user@ip-12-0-1-165 ~]$ sudo iptables -D INPUT -s 12.0.1.189 -j DROP
[ec2-user@ip-12-0-1-165 ~]$
```



File Edit Selection Find View Goto Tools Project Preferences Help

pyflit.py docker networking.md docker commands mysql db name:mysql 1. docker desktop >> enable kubernetes LAB 9 PART 2 RIAK 1 12.0.1.9

```

1 RIAK 1 12.0.1.9
2 RIAK 2 12.0.1.122
3 RIAK 3 12.0.1.165
4 RIAK 4 12.0.1.48
5 RIAK 5 12.0.1.189
6 KONG 18.206.48.111
7 docker ps --all --format "table {{.ID}}\t{{.Names}}\t{{.Image}}"
8 sudo riak-admin cluster status
9
10 sudo riak start (On all Clusters)
11 sudo riak-admin cluster join riak@12.0.1.9
12 sudo riak-admin cluster plan
13 sudo riak-admin cluster status
14 sudo riak-admin cluster commit
15 sudo riak-admin member_status
16
17 Setup Kong To access from Postman
18
19 sudo yum update -y
20 sudo yum install -y docker
21 sudo service docker start
22 sudo usermod -a -G docker ec2-user
23
24 docker network create --driver bridge gumball
25 docker run -d --name kong-database --network gumball -p 9402:9402 cassandra:2.2
26 docker run -d --name kong \
27   --network gumball \
28   -e "KONG_DATABASE=cassandra" \
29   -e "KONG_CASSANDRA_CONTACT_POINTS=kong-database" \
30   -e "KONG_PG_HOST=kong-database" \
31   -p 8000:8000 \
32

```

2 lines, 31 characters selected; Copied 31 characters

Type here to search 11:58 AM 11/22/2018

Testing On each node to ensure new data is reflected

File Edit View Help

New Import Runner My Workspace Invite Upgrade Examples (0)

Filter Collections History Trash

node1 key

GET http://34.239.180.96:8000/node1/types/subjects/buckets/cmpe281/keys/lab32

Send Save

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (14) Test Results Status: 200 OK Time: 177 ms Size: 564 B Save Download

Pretty Raw Preview Auto

{Task:"lab32 from node 4"}

Type here to search 11:59 AM 11/22/2018

Screenshot of Postman interface showing a collection named "node2 key".

The left sidebar shows a list of requests:

- GET ping node 2
- GET ping node 3
- GET ping node 4
- GET ping node 5
- GET Get Bucket properties
- GET set bucket properties
- GET Add key from node 1
- GET node1 key
- GET node2 key
- GET node3 key
- GET node 4 key
- GET node 5 key

The main panel displays a GET request to `http://34.239.180.96:8000/node2/types/subjects/buckets/cmpe281/keys/lab32`. The "Params" tab is selected, showing a table:

KEY	VALUE	DESCRIPTION
Key	Value	Description

The "Body" tab shows the response body:

```
1 {Task:"lab32 from node 4"}
```

Screenshot of Postman interface showing a collection named "node3 key".

The left sidebar shows a list of requests, identical to the one in the previous screenshot:

- GET ping node 2
- GET ping node 3
- GET ping node 4
- GET ping node 5
- GET Get Bucket properties
- GET set bucket properties
- GET Add key from node 1
- GET node1 key
- GET node2 key
- GET node3 key
- GET node 4 key
- GET node 5 key

The main panel displays a GET request to `http://34.239.180.96:8000/node3/types/subjects/buckets/cmpe281/keys/lab32`. The "Params" tab is selected, showing a table:

KEY	VALUE	DESCRIPTION
Key	Value	Description

The "Body" tab shows the response body:

```
1 {Task:"lab32 from node 4"}
```

Screenshot of Postman application showing a collection named "node 4 key".

The left sidebar shows a list of requests:

- GET ping node 2
- GET ping node 3
- GET ping node 4
- GET ping node 5
- GET Get Bucket properties
- GET set bucket properties
- GET Add key from node 1
- GET node1 key
- GET node2 key
- GET node3 key
- GET node 4 key
- GET node 5 key

The main panel displays a GET request to `http://34.239.180.96:8000/node4/types/subjects/buckets/cmpe281/keys/lab32`. The Params tab shows a single parameter:

KEY	VALUE	DESCRIPTION
Key	Value	Description

The Body tab shows the response body:

```
{Task:"lab32 from node 4"}
```

Screenshot of Postman application showing a collection named "node 5 key".

The left sidebar shows a list of requests:

- GET ping node 2
- GET ping node 3
- GET ping node 4
- GET ping node 5
- GET Get Bucket properties
- GET set bucket properties
- GET Add key from node 1
- GET node1 key
- GET node2 key
- GET node3 key
- GET node 4 key
- GET node 5 key

The main panel displays a GET request to `http://34.239.180.96:8000/node5/types/subjects/buckets/cmpe281/keys/lab32`. The Params tab shows a single parameter:

KEY	VALUE	DESCRIPTION
Key	Value	Description

The Body tab shows the response body:

```
{Task:"lab32 from node 4"}
```