# HPC on OpenStack

#### the good, the bad and the ugly

*Ümit Seren HPC Engineer at the Vienna BioCenter*  Github: @timeu Twitter: @timeu\_s

5th EasyBuild User Meeting - Jan 30th, 2020 - Barcelona

#### The "Cloudster" and How we're Building it!



Shamelessly stolen from Damien François Talk --"The convergence of HPC and BigData What does it mean for HPC sysadmins?" -FOSDEM 2019

#### Who Are We?

- Part of Cloud Platform Engineering Team at molecular biology research institutes (IMP, IMBA,GMI) located in Vienna, Austria at the Vienna Bio Center.
- Tasked with delivery and operations of IT infrastructure for ~ 40 research groups (~ 500 scientists).
- IT department delivers full stack of services from workstations, networking, application hosting and development (among many others).
- Part of IT infrastructure is delivery of HPC services for our campus
- 14 People in total for everything.

#### Vienna BioCenter Computing Profile

- Computing infrastructure almost exclusively dedicated to bioinformatics (genomics, image processing, cryo electron microscopy, etc.)
- Almost all applications are data exploration, analysis and data processing, no simulation workloads
- Have all machinery for data acquisition on site (sequencers, microscopes, etc.)
- Operating and running several compute clusters for batch computing and several compute clusters for stateful applications (web apps, databases, etc.)

#### What We Had Before

- Siloed islands of infrastructure
- Cant talk to other islands, can't access data from other island (or difficult logistics for users)
- Nightmare to manage
- No central automation across all resources easily possible



#### Meet the CLIP Project

- OpenStack was chosen to be evaluated further as platform for this
- Setup a project "CLIP" (Cloud Infrastructure Project) and formed project team (4.0 FTE) with a multi phase approach to delivery of the project.
- Goal is to implement not only a new HPC platform but a software defined datacenter strategy based on OpenStack and deliver HPC services on top of this platform
- Delivered in multiple phases

#### What We're Aiming At

R	HPC Cluster					Sp	pai	ŝ	Jupyt						
Linux VM	Linux VM	Linux VM	Linux VM	Linux VM	Linux VM	Linux VM	Linux VM	Linux VM	Linux VM	Linux VM	Linux VM	Linux VM	Linux VM	Linux VM	Linux VM
E	C			0	pen	Sta	ck p	oriv	ate	clo	ud				
Com No	ompute Compute Compute Node Node N		Com	i pute	Com	inpute	Com	inpute	Com	in pute	Com	il-			

#### **CLIP Cloud Architecture Hardware**



- Heterogeneous nodes (high core count, high clock, large memory, GPU accelerated, NVME)
- ~ 200 compute nodes and ~ 7700 Intel SkyLake cores
- 100GbE SDN RDMA capable Ethernet and some nodes with 2x or 4x ports
- ~ 250TB NVMe IO Nodes ~ 200Gbyte/s

#### Tasks Performed within "CLIP"

#### applications on HPC Dez. Feb. Jan. systems" by Erich 2017 2018 2019 Birngruber at 16:00 2 months 8 months 4 months Plan POC **Production** Analysis Deployment Basic **Deeper understanding Interactive Application Production deployment** Deployment, tooling, operations & understanding Cloud & Slurm payload JupyerHub, Rstudio benchmarking Small scale Actual POC Analysis Deployment Production 12 months 10 months since 6 months Jul. 2019

Interactive

### Deploying and Operating the Cloud

### Deploying the Cloud - TripleO (OoO)

- TripleO (OoO): Openstack on OpenStack
- Undercloud: single node deployment of OpenStack.
  - Deploys the **Overcloud**
- **Overcloud**: HA deployment of OpenStack.
  - Cloud for Payload
- Installation with **GUI** or **CLI** ?



#### Deploying the Cloud - Should we use the GUI ?



#### Deploying the Cloud - Should we use the GUI?

O Deployment in progress		3
Resources		
Filter	Showing <b>52</b> of <b>52</b> items	
Name	Status	Updated Time
MysqlRootPassword	CREATE_COMPLETE	2016-11-24T07:00:08Z
PcsdPassword	CREATE_COMPLETE	2016-11-24T07:00:08Z
VipMap	CREATE_COMPLETE	2016-11-24T07:00:08Z
RabbitCookie	CREATE_COMPLETE	2016-11-24T07:00:08Z
Controller	INIT_COMPLETE	2016-11-24T07:00:08Z
ObjectStorage	INIT_COMPLETE	2016-11-24T07:00:08Z
ObjectStoragelpListMap	INIT_COMPLETE	2016-11-24T07:00:08Z
ControllerIpListMap	INIT_COMPLETE	2016-11-24T07:00:08Z
BlockStorageServiceChain	CREATE_IN_PROGRESS	2016-11-24T07:00:08Z
ComputeHostsDeployment	INIT_COMPLETE	2016-11-24T07:00:08Z
RedisVirtualIP	CREATE_COMPLETE	2016-11-24T07:00:08Z
Ctorego\//ictual/D	OPENTE COMPLETE	2016 11 24707:00:007

#### Deploying the Cloud - Code as Infra & GitOps !

- Web GUI does not scale
  - → Disable the Web UI and deploy from the CLI
- TripleO internally uses *heat* to drive *puppet* that drives *ansible* (ツ)\_/
- Use *ansible* to drive the TripleO installer and rest of infra
- Entire end-2-end deployment from code



#### Deploying the Cloud - Pitfalls and Solutions!

- TripleO is slow because **Heat**  $\rightarrow$  **Puppet**  $\rightarrow$  **Ansible** !!
  - Update takes ~ 60 minutes even for simple config change
- Customize using ansible instead ? Unfortunately not robust :-(
  - Stack update (scale down/up) will overwrite our changes
  - $\circ \rightarrow$  services can be down
- $\rightarrow$  Let's compromise: Use both
  - Iterate with ansible  $\rightarrow$  Use TripleO for final configuration
- Ansible everywhere else !
  - Network, Moving nodes between environments, etc

#### **Operating the Cloud - Package Management**

- 3 environments & infra as code: reproducibility and testing of upgrades
- What about software versions ? → **Satellite/Foreman** to the rescue !
- Software Lifecycle environments ↔ Openstack environments

#### Lifecycle Environment Paths

Content ViewsProductsYum RepositoriesDocker RepositoriesPackagesErrata571499522895100

+ Add New Environment

	Dev	Staging	Prod
Content Views	1	1	1
Content Hosts	8	8	199

Oreate Environment Path

#### **Operating the Cloud - Package Management**

- 1. Create Content Views (contains RPM repos and containers)
- 2. Publish new versions of Content Views
- 3. **Test** in dev/staging and **roll** them **forward** to production

ccv-clip												Publ	sh Ne	w Vers	sion	Select	Action	~
Content View	s » ccv-clip »	Versions ≓																
Details	Versions	Content Views	History	Tasł	asks													
Filter			Search 🝷	-														

Version	Status	Environments	Content	Description	Actions
Version 17.0	Published (2020-01-10 14:05:04 +0100)	Library	51050 Packages 5026 Errata ( 893 🛦 3346 産 787 🖬 )	Updated RHEL OS base packages, OSP13 RPMs and Cisco ACI RPMs (4.2.3)	Promote ~
Version 16.0	Promoted to Dev (2019-10-21 16:51:53 +0200)	Dev	50336 Packages 4958 Errata ( 869 🛦 3308 🕉 781 🗈 )	Updated RHEL OS base packages, OSP13 RPMs and Cisco ACI RPMs	Promote v
Version 11.0	Promoted to Library (2019- 10-21 15:50:38 +0200)	Staging Prod	46413 Packages 4474 Errata ( 766 🛦 2983 ।∰ 725 🖿 )	Upgrade OSP13 packages	Promote ~

#### **Operating the Cloud - Tracking Bugs in OS**

- How to keep track of bugs in OpenStack ?
- $\rightarrow$  Track bugs, workaround and the status in JIRA project (CRE)



#### Deploying and operating the Cloud - Summary

Lessons learned and pitfalls of OpenStack/Tripleo:

- OpenStack and TripleO are complex piece of software
  - Dev/staging environment & package management
- Upgrades can break the cloud in unexpected ways.
  - $\circ$  OSP11 (non-containerized)  $\rightarrow$  OSP12 (containerized)
- Containers are no free lunch
  - Container build pipeline for customizations
- TripleO is a supported out of the box installer for common cloud configurations
  - Exotic configurations are challenging
- "Flying blind through clouds is dangerous":
  - Continuous performance and regression testing
- Infra as code (end to end) way to go
  - Requires discipline (proper PR reviews) and release management

- How can we make sure and monitor that the cloud works during operations ?
- We leverage OpenStack's own tempest testing suite to run verification against our deployed cloud.
- First smoke test (~ 128 tests) and if this is successful run full test (~ 3000 tests) against the cloud.

Jenkins	Pipelines	Administration	Ð	] Logout
Pipelines Q				New Pipeline
NAME	HEALTH	BRANCHES	PR	
clip / OLD / clip-tempest-full	<b>6</b>	100	æ.	\$
clip / OLD / clip-tempest-smoke	<b>_</b>	-	-	Å
clip / benchmark / perfkit-dev	G			$\stackrel{\sim}{\sim}$
clip / benchmark / perfkit-staging	-	-	-	Å
clip / clip-ansible-playground	-	(4)		
clip / clip-docker-images		14	1	☆
clip / platinum-backend		-		Å
clip / platinum-frontend		15	575	Å
clip / 🖿 / dev / full		1.2		\$
clip / 🖿 / dev / smoke	4			Å
clip / 🖿 / staging / full		(**)	-	Å
clip / 🖿 / staging / smoke	4	). <del>-</del>	-	Å

- How can we make sure and monitor that the cloud works during operations ?
- We leverage OpenStack's own tempest testing suite to run verification against our deployed cloud.
- First smoke test (~ 128 tests) and if this is successful run full test (~ 3000 tests) against the cloud.

ommit: -	⊙ 6 months ago	Started by upstream pipeline "clip/testing/staging/smoke" build #43
		21 tests have failed Three are 1 new tests (aline, 20 wolding failing and 204 skipped.
ew failing - 1		
> test_hotplug_nic[compute,id=c5adff73=e961=41f1=b4a9=2	343614f18cfa,network] - tempest.scenario.test_network_basic_ops.TestNetworkBasicOps	
isting failures - 20		
> test_hotplug_nic[compute.id-c5adff73-e961-41f1-b4a9-3	343614f18cfa.network] - tempest.scenario.test_network_basic_ops.TestNetworkBasicOps	
> test_volume_backup_export_import[id-a99c54a1-dd80-4	4724-8a13-13bf58d4068d] - tempest.api.volume.admin.test_volumes_backup.VolumesBackupsAdminTest	
> test_volume_backup_reset_status[id-47a35425-a891-4e1	13-961c-c45deea21e94] - tempest.api.volume.admin.test_volumes_backup.VolumesBackupsAdminTest	
> test_create_list_show_delete_interfaces[id-73fe8f02-590d	d-4bf1-b184-e9ca81065051.network] - tempest.api.compute.servers.test_attach_interfaces.AttachInterfacesTestJSON	
> test_list_user_projects[id-a831e70c-e35b-430b-92ed-81e	ebbc5437b8] - tempest.aplidentity.admin.v3.test_users.UsersV3TestUSON	
> test implied roles create check show delete[id-c90c316	6c-d706-4728-bcba-eb1912081b69] - tempest.api.identity.admin.v3.test_roles.RolesV3TestJSON	
> test snapshot backuplid-bbcfa285-af7f-479e-8c1a-8c34	4fc16543cl - tempest.api.volume.test.volumes.snapshots.VolumesSnapshotTestJSON	
> test list endpoints for tokenlid-ca3ea6f7-ed08-4a61-adl	dbd-96906456ad31) - tempest.api.identity.admin.v2.test_tokens.TokensTestJSON	
> test snapshot create delete with volume in use/comput	ite.id-8567b54c-4455-446d-a1cf-651ddeaa3ff2) - tempest.api.volume.test_volumes_snapshots.VolumesSnapshotTestJSON	
> test snapshot create offline delete online[compute.id-52	v210a1de-85a0-11e6-bb21-641c676a5d611 - tempest.api.volume.test.volumes.snapshots.VolumesSnapshotTestUSON	
> test backup create and restore to an existing volumelik	id-b5d837b0-7066-455d-88fc-4a721a8993061 - cinder.tests.tempest.api.volume.test_volume_backup.VolumesBackupsTest	
> test incremental backuplid-c810fe2c-cb40-43ab-96aa-4	471b74516a98) - cinder.tests.tempest.api.volume.test_volume_backup.VolumesBackupsTest	
> test volume snapshot backuplid-885410c6-cd1d-452c-	a409-7c32b7e0be15) - cinder.tests.tempest.api.volume.test_volume_backup.VolumesBackupsTest	
> test backun create attached volume[compute id-07af86	f6d-80af-44r9-a5dr-r8427h1h62e6i - temnest ani volume test volumes hackun VolumesBackunsTest	
> test bootable volume backup and restorelid-2a8ba340-	-dff2-4511-9db7-646f07156b15.imaze] - tempest.api.volume.test_volumes_backup.VolumesBackupsTest	
test haremetal server ons/compute id-549173a5-38ec-4	42hh-h0e2-c8h9H4a0R943 image network1 - immic temnest nhurin tests scenario test haremetal hasic ons BaremetalBasicOns	
> test volume backun create ret detailed list restore dele	etelid-a66eb488-8ee1-47d4-8e9f-575a095728c61 - tempest ani volume test volumes backun VolumesBackun/Gest	
> test hidden stack - heat interrationtests functional test	dark tars StarkTarTed	
<ul> <li>test create ebs image and check boot/compute id-26-2</li> </ul>	anna sugaanna nigrea	
> test unlume host extremicompute id.557rd2r2.deb8.d	ddca-00ba-86/24/54/311b imaga upluma] - tempert consciontert upluma boot nation TetMolumaBootDation	
<ul> <li>text haremetal hear from volume/compute/di/de05e61</li> </ul>	AB221-AAs-h785-5754518-01ef imme network volume] - konje tempert pludn tets removin tet harmetal boot from volume]	B recent a DEV
y test parenteral poor from total net compare a doctored	1011 Historio Science and a contraction of the compact of the contract of the	
pped - 204		
> setUpClass (tempest.api.compute.admin.test_keypairs_v2	:10.KeyPairsV210TestJSON) -	
> setUpClass (tempest.api.compute.admin.test_security_group)	sup_default_rules.SecurityGroupDefaultRulesTest) -	
> setUpClass (tempest.api.compute.admin.test_fixed_ips.Fix	redIPsTestJson) -	
> setUpClass (tempest.api.compute.servers.test_servers.Ser	verShowV247Test) -	
> setUpClass (tempest-api.compute.admin.test_server_diagr	nostics.ServerDiagnosticsV248Test) -	
> setUpClass (tempest.api.compute.admin.test_volume_swa	ap.TestVolumeSwap) -	
> setUpClass (tempest.api.compute.certificates.test_certific	ates.CertificatesV2TestJSON) -	
> setUpClass (tempest.api.compute.admin.test_auto_allocat	te_network.AutoAllocateNetworkTest) -	
> setUpClass (tempest.api.compute.admin.test_live_migratic	on.LiveAutoBlockMigrationV225Test) -	
> setUpClass (tempest.api.compute.admin.test_server_diagr	nostics_negative.ServerDiagnosticsNegativeV248Test) =	
> setUpClass (tempest.api.compute.admin.test_fixed_ips_ne	ngative.FixedIPsNegativeTestJson) -	
> setUpClass (tempest.api.compute.admin.test_live_migratic	on_negative.LiveMigrationNegativeTest) -	
> test_cold_migration[id-4bf0be52-3b6f-4746-9a27-31436	s36fe30d] = tempest.api.compute.admin.test_migrations.MigrationsAdminTest	
> test_list_migrations_in_flavor_resize_situation(id-1b51206	v2-8093-438e-b47a-37d2f597cd64] - tempesLapi.compute.admin.test_migrations.MigrationsAdminTest	

Task overvie

**v** Authentic

validate nov

- Ok, the Cloud works but what about performance ? How can we make sure that OS performs when upgrading software packages etc ?
- We plan to use *Browbeat* to run *Rally* (control plane performance/stress testing), *Shaker* (network stress test) and *PerfkitBenchmarker* (payload performance) tests on a regular basis or before and after software upgrades or configuration changes

Service	Service-level agreement													
Criterion	, lovor a	groom	Detail					Su	ICCess					
max avo du	ration		Average dura	ition of one iteratio	n 0.80s <= 60.00s	Passed		To	ue					
max_second	is_per_iteration		Maximum sec	conds per iteration	0.90s <= 60.00s -	Passed		Tri	ue					
failure_rate			Failure rate c	riteria 0.00% <= 0.0	10% <= 0.00% - Pas	ssed		Tri	ue					
Total du	urations	Min (sec)	Median (sec)	90%ile (sec)	95%ile (sec)	Max (sec)	Avg (sec)	Success	Cou					
authenticate.	keystone	0.656	0.805	0.855	0.873	0.901	0.802	100.0%	100					
total		0.656	0.805	0.855	0.873	0.901	0.802	100.0%	100					
-> duration		0.656	0.805	0.855	0.873	0.901	0.802	100.0%	100					
-> idle_durati	lon	0	0	0	0	0	0	100.0%	100					
0.600			~~~~~	~~~			~~~~	1						
0.600 0.400 0.200						55 Il ide_duration	1 0.000	1						
0.600 0.400 0.200 0.000	10	20	30	40	50 60	55 idle_duration duration	1 0.000 0.782 j0	90						
0.600 0.400 0.200 0.000	10	20	30	40 Heration se	50 60 guence number	55 idle_duration duration	n 0.000 0.782 30	90						
0.600 0.400 0.200 0.000	10 īle	20	30	40 Iteration se	50 60 quence number	55 ide_duration	1 0.000 0.782 J0	90						
0.600 0.400 0.200 0.000 Load Prof	10 île	20	30	40 Beration se	50 60 quence number	55 ide duration duration	1 0.000 0.782 j0	90 parallel Iteration	15					
0.800 0.400 0.200 0.000	10 Tile	20	30	40 Beration se	50 60 guance number	55 ide_duration duration	1 0.000 0.782 30	90 sarallel iteration	15					
0.600 0.400 0.200 Load Prof	10 île 1.00	20	30	40 Iteration se 4.00 5.00 Timelin	50 00 00 00 00 00 00 00 00 00 00 00 00 0	55 idie duration duration	a 0.000 0.782 j0	90 sarallel iteration	15					
0.000 0.200 0.200 Load Prof	10 110	20	30	40 Iteration se 4.00 5.00 Timein	50 00 guarde number 6.00 e (seconds)	55 Idle_duration duration	1 0.000 0.782 0 8.00 9	90 sarallel lteration	16					
0.600 0.600 0.200 0.000 Load Prof	10 11e 1.00	20	30	40 Baration se 4.00 5.00 Timelin	50 60 1 guence number 6.00 e (seconds)	55 ide_duration duration	0 0.000 0.782 j0	90 parallel iteration	16					
0.000 0.000 0.000 0.000 0.000	10 Tile 1.00	20	30 3.00	40 heration se 4.00 5.00 Timeiro 31	50 (00) guerice number 6.00 e (seconds)	55 Ide duration	8.00 9	90 parallel iteration 00 10	15 .00					
0.000 0.000 0.000 Load Prof	10 Tile 1.00	20	30 3.00	40 Beration se 4.00 5.00 Trmein 21 25	50 60 quance number 6.00 e (seconds)	55 ide_duration duration 7.00	5 0.000 0.782 jp 8.00 9	90 parallel teration .00 10	15 .00					
0.600 0.000 0.000 Load Prof 5 8.00	10 ile 1.00	20	30 3.00	400 5.00 Trineiro	50 60 quarce rumber 6.00 e (seconds)	55 ide_duration duration 7.00	8.00 9	90 Sarallel Iteration 00 10	15 .00					
0.000 0.000 0.000 Load Prof	10 The 1.00	20	30 3.00	40 Resiston se 4.00 5.00 Timelin 25 20	50 60 guarce number 6.00 e (seconds)	55 Ide_duration 7.00	8.00 9	90 Sarallel Iteration	15 .00					

- Ok, the Cloud works but what about performance ? How can we make sure that OS performs when upgrading software packages etc ?
- We plan to use *Browbeat* to run *Rally* (control plane performance/stress testing), *Shaker* (network stress test) and *PerfkitBenchmarker* (payload performance) tests on a regular basis or before and after software upgrades or configuration changes

overview	Authentic								
ATT	/ turieritio	ate.keys	tone [15	] (734.2	38s)				
The	Check Keystone Clie	nt.							
uthenticate	Quantinu	ale legat tee	k.						
me	Overview	ans input tas	ĸ						
one [2]	Load duration: 722.7	76 s Full duratio	n: 734.238 s Iter	ations: 10000 F	ailures: 0 Starter	at: 2019-01-1	4T18:15:08		
one [3]									
une (4)	Service-leve	el agreem	ent						
one [5]	Criterion		Detail					Su	ccess
me [6]	max_avg_duration		Average dur	ation of one iterati	on 22.87s <= 6.00s	Failed		Fal	se
ne [7]	max_seconds_per_ite	aration	Maximum se	conds per iteratio	n 30.51s <= 30.00s ·	Failed		Fal	se
one [8]	failure_rate		Failure rate	criteria 0.00% <= 0	.00% <= 0.00% - Pas	sed		Tru	e
one [9]									
me [10]	Total duration	ons							
one [11]	Action	Min (sec)	Median (sec)	90%ile (sec)	95%ile (sec)	Max (sec)	Avg (sec)	Success	Count
me [12]	authenticate.keystone	11.484	22.982	24.665	25.582	30.508	22.871	100.0%	10000
ne [13]	total	11.484	22.982	24.666	25.582	30.508	22.871	100.0%	10000
ne [14]	-> duration	11.484	22.982	24.666	25.582	30.508	22.871	100.0%	10000
ne [15]	-> idle_duration	0	0	0	0	0	0	100.0%	10000
ne [16]	• St	acked OStream	OExpanded				<ul> <li>duration</li> </ul>	ldle_dura	tion
ne [17]	25.921	4 14 14 14 14	a solution of		L			a di la di	
ne [18]	20.000	h. h. e. see a	e en en eles	An Six diamage	outre only their	- 1 - 1 M	THE SPECIFIC STREET	and appropriate	
ne [19]									
ne (20)	15.000								
ne [21]	10.000								
no 1991									
no feel	5,000								
ne (23)	5.000								
ne [23] ne [24]	0.000	1000 2000	3000	4000	5000 6000	7000	8000	9000	100
ne [23] ne [24]	0.000	1000 2000	3000	4000 Iteration se	5000 6000 quence number	7000	8000	9000	100
na [23] ne [24] e_neutron e_neutron [2]	0.000	1000 2000	3000	4000 Iteration se	5000 6000 quence number	7000	8000	9000	10
ne [23] ne [24] e_neutron e_neutron [2] e_neutron [3]	Load Profile	1000 2000	3000	4000 Iteration se	5000 6000 quence number	7000	8000 ● P	9000 arailei iteratior	100
m (23) the [24] a_neutron a_neutron [2] b_neutron [3] a_neutron [4]	Load Profile	1000 2000	3000	4000 Iteration se	5000 6000 quence number	7000	8000 • P	9000 arallel iteration	10
me (z4) nn (23) e_neutron e_neutron [2] e_neutron [3] e_neutron [4] e_neutron [5]	Load Profile	1000 2000	3000	4000 Iteration se	5000 6000 quence number	7000	8000	9000 arallel iteration	100
me [23] me [23] e_neutron e_neutron [2] e_neutron [3] e_neutron [4] e_neutron [5] e_neutron [6]	200 100	1000 2000	3000	4000 Iteration se	5000 6000 quence number	7000	8000	9000 arailel iteration	100
ne (24) ne (24) e_nektron [2] e_nektron [3] e_nektron [5] e_nektron [6] e_nektron [7]	5.000 0.000 Load Profile	1000 2000	3000	4000 Iteration se 300.00	5000 6000 quence number 400.00	7000	9000 • P 600.00	9000 arallel iteration 7	100
ne (23) ne (24) e_neuron (2) e_neuron (3) e_neuron (4) e_neuron (5) (	5.000 0.000 Load Profile 200 100 8.00	1000 2000	3000	4000 Iteration se 300.00 Timelin	5000 6000 quence number 400.00 e (seconds)	7000	9000 • P 600.00	9000 arallel iteration 7	100 8 00.00
ee [23] ee [24] e, netation e, netation [3] e, netation [4] e, netation [5] e, netation [6] e, netation [6] e, netation [6] e, netation [6]	5.000 0.000 Load Profile 200 100 8.00	100.00	3000	4000 Iteration se 300.00 Timetin	5000 6000 quence number 400.00 e (seconds)	7000	8000 • p 600.00	9000 arallel iteration 7	10
ne (24) ne (24) e_nearon (3) e_nearon (3) e_nearon (3) e_nearon (6) e_nearon (7) e_nearon (7) e_nearon (7) e_nearon (7) e_nearon (7) e_nearon (7)	Load Profile	100.00	3000	4000 Iberation se 300.00 Timetin	5000 6000 quence number 400.00 e (seconds)	7000	8000	9000 arallel iteration 7	10
به درغ در المعلم الال در المعلم الال در المعلم الا در المعلم المعلم الا در المعلم	Load Profile	100.00	3000 200.00 208.00	4000 Iteration se 300.00 Timetin 1279	5000 6000 quence number 400.00 e (seconds)	7000	8000 • p	9000 arallel iteration 7	100 8 00.00
νη κεί και (2) και (2) κα	Load Profile	100.00 100.00	3000 200.00	4000 Iteration se 300.00 Timetin 1279	5000 6000 guence number 400.00 e (seconds)	7000	8000 • P 600.00	9000 arailel iteration 7	100 s 00.00 task
νη κεί τη κεί μηματος 19 μηματος 19 μημματος 19	Load Profile	1000 2000 100.00	3000 200.00 85 • emos	4000 Iteration se 300.00 Timetin 1200 000	5000 6000 guerce number 400.00 e (seconds)	500.00	8000 P P 600.00	9000 arailel iteration 7	100 15
	Load Profile	1000 2000 100.00	3000 200.00 255 • errors	4000 Iteration se 300.00 Timein 1279 1100 1000 000	500 6000 quence number 400.00 e (aeconds)	500.00	8000 • P 600.00	9000 anallel iteration 7	100 15 00 00 00
	Load Profile	1000 2000 100.00	3000 200.00	4000 Iberation se 300.00 Timein 1220 1000 800 800	5000 6000 quence number 400.00 e (seconds)	500.00	8000 • P 600.00	9000 atallel iteration 7	10 15 000.00
	5.00 8.00 Load Profile 300 8.00 Distribution	1000 2000 100.00 • MCC	3000 200.00 255 • errors	4000 Iteration se 300.00 Timelin 1200 0000 1000 0000 400	5000 6000 quence number 400.00 e (seconds)	500.00	8000	9000 arailel iteration 7	100 8 00.00
κε κ.ς κα (24) κα	5.000 8.000 Load Profile 200 100 8.00 Distribution	000 200 100.00 • succe	3000 200.00 25 • errors	4000 Beration se 300.00 Timelin 1276 1200 500 500 500 500 500 500 500 500 500	5000 6000 guerrice number 400.00 ((seconds)	7000	8000	9000 arallel iteration 7	100
	Load Profile	100 2000 100.00 • succi	3000 200.00 95 • errors	4000 Iteration se 300.00 Timetin 1270 800 800 800 800 800 800 800 800 800 8	5000 6000 querico number 400.00 e (seconds)	500.00	8000	9000 ataliei iteration 7	100 100 100 100
	Load Profile	000 2000 100.00 • succe 0 0000	3000 200.00 ss errors	4000 Iteration se 300.00 Timein 1279 1200 600 600 600 600 600 600 600 600 600	400.00 e (seconds)	7000 500.00	8000	9000 araitel iteration 7 2	100 15 00.00 task

- Ok, the Cloud works but what about performance ? How can we make sure that OS performs when upgrading software packages etc ?
- We plan to use *Browbeat* to run *Rally* (control plane performance/stress testing), *Shaker* (network stress test) and *PerfkitBenchmarker* (payload performance) tests on a regular basis or before and after software upgrades or configuration changes

	Authentica	ic.reys		1 (+10.9	1431				
ine	Check Keystone Client								
henticate	Overview Detai	e Cailurae	Input taek						
ie.	OKING DOL	3 Futures	mput such						
10 [2]	Load duration: 405.465	s Full duration	n: 416.904 s lite	rations: 10000 F	ailures: 5069 St	arted at: 2019	-01-14T19:04	1:05	
ie [3]									
ne [4]	Service-level	agreem	ent						
e [5]	Criterion		Detail					Su	iccess
e [6]	max_avg_duration		Average dur	ation of one iteratio	n 30.19s <= 6.00s	- Failed		Fa	lse
ie [7]	max_seconds_per_itera	tion	Maximum se	conds per iteration	257.218 <= 30.009	- Failed		Fa	lse
e [8]	failure_rate		Failure rate	criteria 0.00% <= 50	.69% <= 0.00% - Fa	alled		Fa	lse
e [9]	Total duration								
e [10]	lotal duration	IS							
e [11]	Action	Min (sec)	Median (sec)	90%ile (sec)	95%ile (sec)	Max (sec)	Avg (sec)	Success	Count
e [12]	authenticate.keystone	0.019	20.846	32.851	38.226	257.212	16.484	49.3%	10000
e [13]	total	0.019	20.846	32.852	38.226	257.212	16.484	49.3%	10000
e [14]	-> duration	0.019	20.846	32.852	38.226	257.212	16.484	49.3%	10000
e [15]	-> idle_duration	0	0	0	0	0	0	49.3%	10000
e [16]	• Stat	ked O Stream	OExpanded			oduration (	idle_duration	• failed_dura	ation
n [17]	92.579								
e [18]	80.000								
e [19]	60.000								
e [20]									
e [21]	40.000	H rat all	a de la la la	A day and a					
e [22]	20.000								
e [23]									
e [24]	0.000 10	2000	3000	4000	5000 6000	7000	8000	9000	100
				4000					200
neutron				Iteration se	quence number				100
_neutron _neutron [2]	Load Profile			Iteration se	quence number				100
neutron neutron (2) neutron (3)	Load Profile			Iteration se	quence number			parallel iteration	15
neutron [2] neutron [3] neutron [4]	Load Profile			Iteration se	quence number			parallel iteration	15
neutron (2) neutron (3) neutron (4) neutron (5)	Load Profile			Neration se	quence number			parallel iteration	15
neutron [2] neutron [3] neutron [4] neutron [5] neutron [6]	Load Profile			Heration se	quence number			parallel iteration	15
neutron (2) _neutron (3) _neutron (4) _neutron (5) _neutron (6) _neutron (7)	Load Profile	50.00 ::	100.00 11	Neration se	quence number 00 250.0	10 30	1.00	parallel iteration	400.00
neutron (2) _neutron (3) _neutron (4) _neutron (5) _neutron (6) _neutron (7) _neutron (8)	Load Profile	50.00 :	100.00 11	ieration se iteration se 0.00 200 Timelin	,00 250.0 a (seconds)	20 30	0.00	parallel iteration 350.00	400.00
neutron [2] _neutron [3] _neutron [4] _neutron [5] _neutron [6] _neutron [7] _neutron [8] _neutron [9]	Load Profile	50.00	100.00 19	Heraton se beraton se 50.00 2000 Timelin	,00 250.6 e (seconds)	00 30I	2.00	parallel Deration	400.00
neutron [2] neutron [3] neutron [4] neutron [6] neutron [6] neutron [8] neutron [9] neutron [10]	Load Profile	50.00 :	100.00 11	Beraton se beraton se 50.00 200 Timelo	,00 250.0 a (seconds)	00 30I	1.00	parallel iteration 350.00	400.00
reation  pestion  pestion  pestion  pestion  pestion  pestion  pestion  pertion  pertion pertion  pertion pertion pertion pertion pertion pertion pertion pertion pertion pertion pertion pertion pertion pertion pertion pertion pertion per	Load Profile	50.00 :	100.00 11 105 • errors	Beration se 30.00 200 Timelin 5069	00 250.6 e (seconds)	0 300	2.00	parallel iteration	400.00
reation reatio	Load Profile	50.00 :	100.00 11 ISS ●errors	Beration se 30.00 200 Timelin 5069	00 250.6 e (seconds)	0 300	1.00	paratel Heration	400.00
neutron (neutron [2] (neutron [3] (neutron [4] (neutron [6] (neutron [7] (neutron [10] (1	Load Profile	50.00 : • SUCO	100.00 11 sss eerrors	8000 200 30.00 200 Timelin 60	00 250.4	0 30	0.00	paralel teration 350.00	400.00
Instituto Institution [2] Institution [3] Institution [4] Institution [6] Institution [7] Institution [7] Institution [12] Institution [12] Institution [12] Institution [14]	Load Profile	\$0.00 :	100.00 21 His errors	50.00 2000 Timelin 50.00 2000 Timelin 5000 5000 5000 5000	00 250.0 c(seconds)	0 30	0.00	paralel teration 350.00	400.00
reation     (         reation         (         reation         (         reation         (         reation         (         reation         (         reation         (         reation         (         reation         (         reation         (         reation         (         reation         (         (         reation         (         (         (	Load Profile	50.00 :	100.00 21 109 entres 4996	50.00 2000 Trmsin 50.00 2000 Trmsin 5069 90 90 90 90 90 90 90 90 90 90 90 90 90	00 250.0 s (seconds)	0 300	0.00	parallel iteration	400.00
_neutron [2] neutron [4] neutron [4] neutron [4] neutron [6] neutron [7] neutron [7] neutron [10] neutron [11] neutron [12] neutron [13] neutron [14] neutron [14] neutron [15]	Load Profile	50.00 : • SUCO	000.00 1/ es erros	1000 Iteration se 10.00 2000 Timelin 1000 2000 2000 2000	00 250.6 e (seconds)	0 300	1.00	parallel iteration	400.00
restion (restors [2] (restors [3] (restors [4] (restors [6] (restors [6] (restors [6] (restors [6] (restors [12] (restors [12] (restors [12] (restors [14] (restors [15] (restors [15] (restors [15])	Load Profile	50.00 : • SMCO	100.00 21 105 • errors 45%	50.00 200 Timelio 50.00 200 Timelio 5000 200 1000 1000	00 250.0 (seconds)	0 20	2.00	paratel Beration	100 400.00
Instanton (Instanton [2] Instanton [3] Instanton [4] Instanton [4] Instanton [6] Instanton [6] Instanton [10] Instanton [10] Instanto	Load Profile	50.00 S	100.00 21 es errors 49%	4000 Benzion se 50.00 2000 Trmelio 4000 000 000 000 000 000 000	00 250.4 (seconds)	0 30	2.00	paratel Beration	400.00

• Grafana and Kibana dashboard can show more than individual rally graphs:





• Browbeat can show differences between settings or software versions:

+												-+
Scenario	Action		conc	.	times	I	0b5ba58	3c	2b177f3b		% Diff	
+												-+
create-list-router	neutron.create_route	•	500		32		19.940		15.656		-21.483	
create-list-router	neutron.list_routers		500		32	I	2.588		2.086		-19.410	
create-list-router	neutron.create_netwo	١k	500		32	I	3.294		2.366		-28.177	
create-list-router	neutron.create_subnet	:	500		32	I	4.282		2.866		-33.075	
create-list-port	neutron.list_ports		500		32	I	52.627		43.448		-17.442	
create-list-port	neutron.create_netwo	۰k	500		32	I	4.025		2.771		-31.165	
create-list-port	neutron.create_port		500		32	I	19.458		5.412		-72.189	
create-list-subnet	neutron.create_subnet	:	500		32	I	11.366		4.809		-57.689	
create-list-subnet	neutron.create_netwo	۰k	500		32	I	6.432		4.286		-33.368	
create-list-subnet	neutron.list_subnets		500		32	I	10.627		7.522		-29.221	
create-list-network	neutron.list_networks	;	500		32	I	15.154		13.073		-13.736	
create-list-network	neutron.create_netwo	۰k	500		32	I	10.200		6.595		-35.347	
+												-+
+												-+
UUID		V	ersio	n	Bu	i.	ld		Number of r	un	s	
+												-+
938dc451-d881-4f28-a	6cb-ad502b177f3b	q	ueens		2018-	0	3-20.2		1			
6b50b6f7-acae-445a-a	c53-78200b5ba58c	0	cata		2017-	X	X-XX.X		3			
+												-+

Scrolling through Browbeat 22 documents...

## Deploying the Payload

### Deploying the Cloud - SLURM Cluster

- 2 step process:
  - OpenStack Heat to provision →
     Ansible inventory
  - Ansible playbook/roles<sup>1</sup> for config -> SLURM cluster
- Satellite for package management
- Dev & staging env for testing → roll over to production
- Deploy other complex systems (Spark cluster, k8s, etc)



### Deploying the Cloud - Tunings for HPC

• Tuning, Tuning, Tuning required for excellent performance

Tuning	Caveats / Downside
NUMA clean instances (KVM process layout)	No live migrations No mixing of different VM flavors
Static huge pages (KSM etc.) setup	If not enough memory is left to hypervisor → swapping or host services get OOM. No mixing of different VM flavors
Core isolation (isolcpus)	Performance drop in virtual networking performance $\rightarrow$ SR-IOV
PCI-E passthrough (GPUs, NVME) and SR-IOV (NICs)	No live migrations and less features compared to fully virtualized networking

#### Mean execution time



#### Standard deviation of execution time



2x SLURM BM

#### Deploying the Cloud - Pitfalls and Issues

- Ansible is slow: Slurm playbook takes ~1 hour (clean 2nd run !)
  - Use tags for recurring day 2 operations (i.e new mount points, change of QOS, etc)
- Satellite 👍 for software versions but remove upstream Centos repos after install
- Some issues only hit under scale:
  - SDN scaling issues when provisioning more than 70 nodes. Workaround: scale in batches
- Isolation of environments ends with shared infra components especially when tightly integrating with OpenStack
  - Update of **DEV** environment caused datacenter wide network outage (bug in SDN)
- Beware of unintended consequences of code changes
  - Triggered accidental re-deploy of payload because of single line change in heat template

### HPC on OpenStack - Lessons Learned

#### Bad & Ugly

- OpenStack is *incredibly* complex
- OpenStack is not a product. It is a framework.
- You need 2-3 OpenStack environments (development, staging, prod in our case) to practice and understand upgrades and updates.
- Scaling above certain amount of nodes will be an issue
- Cloud networking is really hard (especially in our case)

#### Good

- Open source software with commercial support
- OpenStack integrates well with existing datacenter infrastructure
- API driven software defined datacenter
- Easily deploy multiple payloads side by side like in a Cloud
- Covers a wide range of use cases ranging from virtualized & baremetal HPC clusters to container orchestration engines

#### **Acknowledgements**

**HPC** Team

Erich Birngruber Petar Forai Petar Jager Ümit Seren

