February 20, 2017

Hugh Blemings (http://hugh.blemings.id.au)

Lwood-20170219 (http://hugh.blemings.id.au/2017/02/20/lwood-20170219/)

Introduction

Welcome to Last week on OpenStack Dev ("Lwood") for the week just past. For more background on Lwood, please refer here (http://hugh.blemings.id.au/openstack/lwood/).

Basic Stats for the week 13 to 19 February for openstack-dev:

~575 Messages (three messages more than the long term average)

~212 Unique threads (up about 18% relative to the long term average)

Traffic picked up a fair bit this week – almost exactly on the long term average for messages. Threads up a bit more – lots of short threads, a mixture of those about project logos and PTG logistics contributing there I think.

Notable Discussions – openstack-dev

Proposed Pike release schedule

Thierry Carrez posted (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112124.html) to the list with some information on the proposed Pike release schedule. The human friendly version is here (http://docs-draft.openstack.org/54/425254/3/check/gate-releases-docs-ubuntu-xenial/a06786d//doc/build/html/pike/schedule.html). Week zero – release week – is the week of August 28

Assistance sought for the Outreachy program

From Mahati Chamarthy an update (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112104.html) about the Outreachy (https://wiki.openstack.org/wiki/Outreachy) program – an initiative that helps folk from underrepresented groups get involved in FOSS. It's a worthy initiative if there were ever one, a lot of support was shown from it at linux.conf.au (http://linux.conf.au) recently as it happens too.

Please consider getting involved and/or supporting the programs work financially.

Session voting open for OpenStack Summit Boston

Erin Disney writes (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112247.html) to advise that voting is open for sessions in Boston until 7:59am Wednesday 22nd February (UTC) She notes that unique URLs for submissions have been returned based on community feedback.

Final Team Mascots

A slew of messages this week announcing the final versions of the team mascots that the OpenStack Foundation has been coordinating. I briefly contemplated listing them all here but that seemed a sub-optimal way to spend the next hour – so if you want to find one for your favourite project, follow this link (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112116.html) and use your browser search for "mascot" or "logo" – mostly the former. The Foundation will, I gather, be publishing a canonical list of them all shortly in any case.

In a thread (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112145.html) about licensing for the images kicked off by Graham Hayes was the clarification that they'll be CC-BY-ND (https://creativecommons.org/licenses/by-nd/2.0/au/)

End of Week Wrap-ups, Summaries and Updates

- Chef (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112194.html) by Sam Cassiba
- Horizon (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112437.html) from Richard Jones
- Ironic (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112070.html) courtesy of Ruby Loo

People and Projects

Project Team Lead Election Conclusion and Results

Kendall Nelson summarises the results of the recent PTL elections in a post (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111769.html) to the list. Most Projects had the one PTL nominee, those that went to election were Ironic, Keystone, Neutron, QA and Stable Branch Maintenance. Full details in Kendall's message.

Core nominations & changes

- [Glance] Revising the core list (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112472.html) (multiple changes) – Brian Rosmaita

- [Heat] Stable-maint additions (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112272.html) (multiple changes) – Zane Bitter
- [Ironic] Adding Vasyl Saienko (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112442.html) and Mario Villaplana (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112442.html), temporarily removing Devananda Van Der Veen (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112442.html) – Dmitry Tantsur
- [Packaging-RPM] Nominating Alberto Planas Dominguez (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112358.html) for core – Igor Yozhikov
- [Watcher] Nominating Prudhvi Rao Shedimbi (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112114.html) to core – Vincent Françoise
- [Watcher] Nominating Li Canwei (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112116.html) to Core – Alexander Chadin

Miscellanea

Further reading

Don't forget these excellent sources of OpenStack news – most recent ones linked in each case

- What's Up, Doc? (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111932.html) by Alexandra Settle
- API Working Group newsletter (http://lists.openstack.org/pipermail/openstack-dev/2017-February/112388.html) – Chris Dent and the API WG
- OpenStack Developer Mailing List Digest (http://lists.openstack.org/pipermail/openstack-dev/2017-January/111139.html) by Mike Perez & Kendall Nelson
- OpenStack news over on opensource.com (https://opensource.com/article/17/2/openstack-news-february-20) by Jason Baker
- OpenStack Foundation Events Page (https://www.openstack.org/community/events/) for a frequently updated list of events

Credits

This weeks edition of Lwood brought to you by Daft Punk (https://en.wikipedia.org/wiki/Daft_Punk) (Random Access Memories (https://en.wikipedia.org/wiki/Random_Access_Memories)) and DeeExpus (https://en.wikipedia.org/wiki/DeeExpus) (King of Number 33)

by hugh at February 20, 2017 07:30 AM (http://hugh.blemings.id.au/2017/02/20/lwood-20170219/)

Opensource.com (https://opensource.com/taxonomy/term/5126/feed/feed)

Boston summit preview, Ambassador program updates, and more OpenStack news (https://opensource.com/article/17/2/openstack-news-february-20)

Are you interested in keeping track of what is happening in the open source cloud? Opensource.com is your source for news in OpenStack (https://opensource.com/resources/what-is-openstack), the open source cloud infrastructure project.

OpenStack around the web

From news sites to developer blogs, there's a lot being written about OpenStack every week. Here are a few highlights.

by Jason Baker at February 20, 2017 06:00 AM (https://opensource.com/article/17/2/openstack-news-february-20)

February 19, 2017

Thierry Carrez (https://ttx.re/)

Using proprietary services to develop open source software (https://ttx.re/using-proprietary-to-develop-oss.html)

It is now pretty well accepted that open source is a superior way of producing software. Almost everyone is doing open source those days. In particular, the ability for users to look under the hood and make changes results in tools that are better adapted to their workflows. It reduces the cost and risk of finding yourself locked-in with a vendor in an unbalanced relationship. It contributes to a virtuous circle of continuous improvement, blurring the lines between consumers and producers. It enables everyone to remix and invent new things. It adds up to the common human knowledge.

And yet

And yet, a lot of open source software is developed on (and with the help of) proprietary services running closed-source code. Countless open source projects are developed on GitHub, or with the help of Jira for bugtracking, Slack for communications, Google docs for document authoring and sharing, Trello for status boards. That sounds a bit paradoxical and hypocritical -- a bit too much "do what I say, not what I do". Why is that ? If we agree that open source has so many tangible benefits, why are we so willing to forfeit them with the very tooling we use to produce it ?

But it's free !

The argument usually goes like this: those platforms may be proprietary, they offer great features, and they are provided free of charge to my open source project. Why on Earth would I go through the hassle of setting up, maintaining, and paying for infrastructure to run less featureful solutions ? Or why would I pay for someone to host it for me ? The trick is, as the saying goes, when the product is free, **you** are the product. In this case, your open source community is the product. In the worst case scenario, the personal data and activity patterns of your community members will be sold to 3rd parties. In the best case scenario, your open source community is recruited by force in an army that furthers the network effect and makes it even more difficult for the next open source project to not use that proprietary service. In all cases, you, as a project, decide to not bear the direct cost, but ask each and every one of your contributors to pay for it indirectly instead. You force all of your contributors to accept the ever-changing terms of use of the proprietary service in order to participate to your "open" community.

Recognizing the trade-off

It is important to recognize the situation for what it is. A trade-off. On one side, shiny features, convenience. On the other, a lock-in of your community through specific features, data formats, proprietary protocols or just plain old network effect and habit. Each situation is different. In some cases the gap between the proprietary service and the open platform will be so large that it makes sense to bear the cost. Google Docs is pretty good at what it does, and I find myself using it when collaborating on something more complex than etherpads or ethercalcs. At the opposite end of the spectrum, there is really **no** reason to use Doodle when you can use Framadate (https://framadate.org). In the same vein, Wekan (https://wekan.io) is close enough to Trello that you should really consider it as well. For Slack vs. Mattermost (https://about.mattermost.com) vs. IRC, the trade-off is more subtle. As a sidenote, the cost of lock-in is a lot reduced when the proprietary service is built on standard protocols. For example, GMail is not that much of a problem because it is easy enough to use IMAP to integrate it (and possibly move away from it in the future). If Slack was just a stellar opinionated client using IRC protocols and servers, it would also not be that much of a problem.

Part of the solution

Any simple answer to this trade-off would be dogmatic. You are not unpure if you use proprietary services, and you are not wearing blinders if you use open source software for your project infrastructure. Each community will answer that trade-off differently, based on their roots and history. The important part is to acknowledge that nothing is free. When the choice is made, we all need to be mindful of what we gain, and what we lose. To conclude, I think we can all agree that all other things being equal, when there is an open-source solution which has all the features of the proprietary offering, we all prefer to use that. The corollary is, we all benefit when those open-source solutions get better. So to be part of the solution, consider helping those open source projects build something as good as the proprietary alternative, especially when they are pretty close to it feature-wise. That will make solving that trade-off a lot easier.

by Thierry Carrez at February 19, 2017 01:00 PM (https://ttx.re/using-proprietary-to-develop-oss.html)

February 18, 2017

Clint Byrum (http://fewbar.com/)

Free and Open Source Leaders -- You need a President (http://fewbar.com/2017/02/open-source-governance-needs-presidents/)

Recently I was lucky enough to be invited to attend the Linux Foundation Open Source Leadership Summit (http://events.linuxfoundation.org/events/open-source-leadership-summit). The event was stacked with many of the people I consider mentors, friends, and definitely leaders in the various Open Source and Free Software communities that I participate in.

I was able to observe the CNCF (https://www.cncf.io/) Technical Oversight Committee meeting while there, and was impressed at the way they worked toward consensus where possible. It reminded me of the OpenStack Technical Committee (https://www.openstack.org/foundation/tech-committee/) in its make up of well spoken technical individuals who care about their users and stand up for the technical excellence of their foundations' activities.

But it struck me (and several other attendees) that this consensus building has limitations. Adam Jacob (https://twitter.com/adamhjk) noted that Linus Torvalds had given an interview on stage earlier in the day where he noted that most of his role was to listen closely for a time to differing opinions, but then stop them when it was clear there was no consensus, and select one that he felt was technically excellent, and move on. Linus, being the founder of Linux and the benevolent dictator of the project for its lifetime thus far, has earned this moral authority.

However, unlike Linux, many of the modern foundation-fostered projects lack an executive branch. The structure we see for governance is centered around ensuring corporations that want to sponsor and rely on development have influence. Foundation members pay dues to get various levels of board seats or corporate access to events and data. And this is a good thing, as it keeps people like me paid to work in these communities.

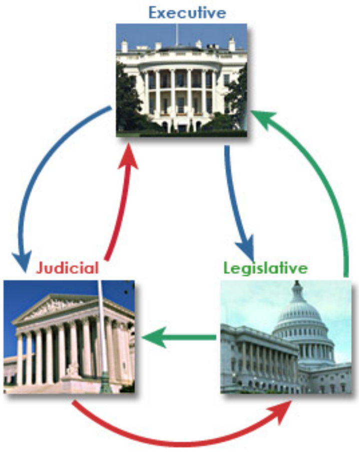
However, I believe as technical contributors, we sometimes give this too much sway in the actual governance of the community and the projects. These foundation boards know that day to day decision making should be left to those working in the project, and as such allow committees like the CNCF (https://www.cncf.io/) TOC or the OpenStack TC (https://www.openstack.org/foundation/tech-committee/) full agency over the technical aspects of the member projects.

I believe these committees operate as a legislative branch. They evaluate conditions and regulate the projects accordingly, allocating budgets for infrastructure and passing edicts to avoid chaos. Since they're not as large as political legislative bodies like the US House of Representatives & Senate, they can usually operate on a consensus basis, and not drive everything to a contentious vote. By and large, these are as nimble as a legislative body can be.

However, I believe we need an executive to be effective. At some point, we need a single person to listen to the facts, entertain theories, and then decide, and execute a plan. Some projects have natural single leaders like this. Most, however, do not.

I believe we as engineers aren't generally good at being like Linus. If you've spent any time in the corporate world you've had an executive disagree with you and run you right over. When we get the chance to distribute power evenly, we do it.

But I think that's a mistake. I think we should strive to have executives. Not just organizers like the OpenStack PTL (https://docs.openstack.org/project-team-guide/ptl.html), but more like the Debian Project Leader (https://www.debian.org/devel/leader). Empowered people with the responsibility to serve as a visionary and keep the project's decision making relevant and of high quality. This would also give the board somebody to interact with directly so that they do not have to try and convince the whole community to move in a particular direction to wield influence. In this way, I believe we'd end up with a system of checks and balances similar to the US Constitution



So here is my suggestion for how a project executive structure could work, assuming there is already a strong technical committee and a well defined voting electorate that I call the "active technical contributors".

1. The president is elected by Condorcet (https://en.wikipedia.org/wiki/Condorcet_method) vote of the active technical contributors of a project for a term of 1 year.

2. The president will have veto power over any proposed change to the project's technical assets.
3. The technical committee may override the president's veto by a super majority vote.
4. The president will inform the technical contributors of their plans for the project every 6 months.

This system only works if the project contributors expect their project president to actively drive the vision of the project. Basically, the culture has to turn to this executive for final decision making before it comes to a veto. The veto is for times when the community makes poor decisions. And this doesn't replace leaders of individual teams. Think of these like the governors of states in the US. They're running their sub-project inside the parameters set down by the technical committee and the president.

And in the case of foundations or communities with boards, I believe ultimately a board would serve as the judicial branch, checking the legality of changes made against the by-laws of the group. If there's no board of sorts, a judiciary could be appointed and confirmed, similar to the US supreme court or the Debian CTTE (https://www.debian.org/devel/tech-ctte). This would also just be necessary to ensure that the technical arm of a project doesn't get the foundation into legal trouble of any kind, which is already what foundation boards tend to do.

I'd love to hear your thoughts on this on Twitter, please tweet me @SpamapS (https://twitter.com/spamaps) with the hashtag #OpenSourcePresident to get the discussion going.

February 18, 2017 12:00 AM (http://fewbar.com/2017/02/open-source-governance-needs-presidents/)

February 17, 2017

RDO (http://rdoproject.org/blog/)

An OpenStack Distribution

OpenStack Project Team Gathering, Atlanta, 2017 (http://rdoproject.org/blog/2017/02/openstack-project-team-gathering-atlanta-2017/)

<iframe allowfullscreen="" frameborder="0" height="315" src="https://www.youtube.com/embed/oOKnJaJI7j8?list=PL27cQhFqK1QzaZL1XrX_CzT7uCOWQ64xM" width="560"></iframe>
Over the last several years, OpenStack has conducted OpenStack Summit (http://openstack.org/summit) twice a year. One of these occurs in North America, and the other one alternates between Europe and Asia/Pacific.

This year, OpenStack Summit in North America is in Boston (https://www.openstack.org/summit/boston-2017/) , and the other one will be in Sydney (https://www.openstack.org/summit/sydney-2017/).

This year, though, the OpenStack Foundation is trying something a little different. Wheras in previous years, a portion of OpenStack Summit was the developers summit, where the next version of OpenStack was planned, this year that's been split off into its own separate event called the PTG - the Project Teams Gathering (https://www.openstack.org/ptg). That's going to be happening next week in Atlanta.

Throughout the week, I'm going to be interviewing engineers who work on OpenStack. Most of these will be people from Red Hat, but I will also be interviewing people from some other organizations, and posting their thoughts about the Ocata release - what they've been working on, and what they'll be working on in the upcoming Pike release, based on their conversations in the coming week at the PTG.

So, follow this channel (https://www.youtube.com/playlist?list=PL27cQhFqK1QzaZL1XrX_CzT7uCOWQ64xM) over the next couple weeks as I start posting those interviews. It's going to take me a while to edit them after next week, of course. But you'll start seeing some of these appear in my YouTube channel over the coming few days.

Thanks, and I look forward to filling you in on what's happening in upstream OpenStack.

by Rich Bowen at February 17, 2017 07:56 PM (http://rdoproject.org/blog/2017/02/openstack-project-team-gathering-atlanta-2017/)

Rob Hirschfeld (https://robhirschfeld.com)

"Why SRE?" Discussion with Eric @Discoposse Wright (https://robhirschfeld.com/2017/02/17/sre-eric-discoposse-wright/)

My *focus on SRE series (https://robhirschfeld.com/2016/12/29/evolution-or-rebellion-the-rise-of-site-reliability-engineers-sre/) continues… At RackN (https://rackn.com/), we see a coming infrastructure explosion (https://robhirschfeld.com/2017/01/17/spiraling-ops-debt-the-sre-coding-imperative/) in both complexity and scale. Unless our industry radically rethinks operational processes, current backlogs will escalate and stability, security and sharing will suffer.*

(https://robhirschfeld.files.wordpress.com/2017/02/ericewright.jpghttp://discoposse.com/)I was a guest on Eric "@discoposse" Wright (http://discoposse.com/) of the Green Circle Community (http://greencircle.vmturbo.com/) #42 Podcast (https://itunes.apple.com/ca/podcast/gc-on-demand/id1072900664http://bit.ly/2kGuV2D)(my previous appearance (https://robhirschfeld.com/2016/05/24/open-source-as-reality-tv-and-burning-data-centers-gcondemand-podcast-notes/)).

LISTEN NOW: Podcast #42 (http://bit.ly/2kGuV2D)

In this action-packed 30 minute conversation, we discuss the industry forces putting pressure on operations teams. These pressures require operators to be investing much more heavily on reusable automation.

That leads us towards why Kubernetes is interesting and what went wrong with OpenStack (I actually use the phrase "dumpster fire"). We ultimately talk about how those lessons embedded in Digital Rebar architecture (http://rebar.digital).

Add a Comment (http://feeds.wordpress.com/1.0/gocomments/robhirschfeld.wordpress.com/6592/)

by Rob H at February 17, 2017 05:40 PM (https://robhirschfeld.com/2017/02/17/sre-eric-discoposse-wright/)

OpenStack Superuser (http://superuser.openstack.org)

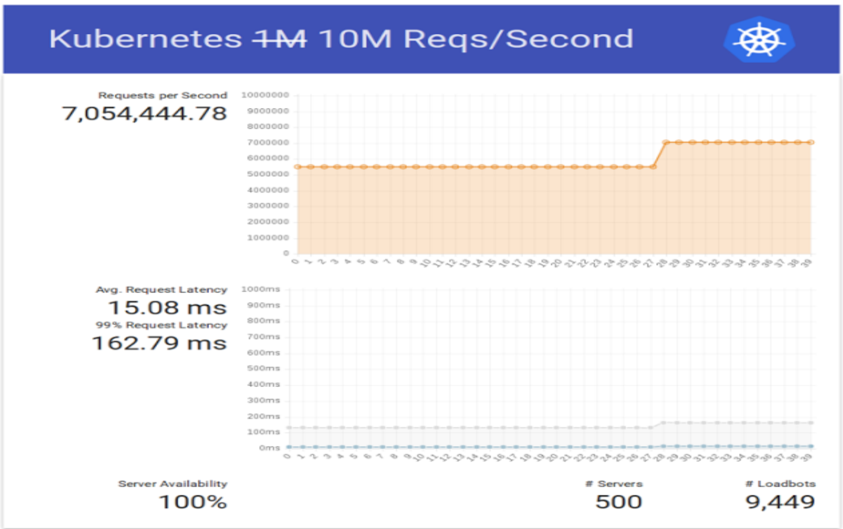
Containers on the CERN Cloud (http://superuser.openstack.org/articles/containers-cern-cloud/)

We have recently made the Container-Engine-as-a-Service (Magnum) available in production at CERN as part of the CERN IT department services for the LHC experiments and other CERN communities. This gives the OpenStack cloud users Kubernetes, Mesos and Docker Swarm on demand within the accounting, quota and project permissions structures already implemented for virtual machines.

We shared the latest news on the service with the CERN technical staff (link (https://cernbox.cern.ch/index.php/s/PixyT67aIIaHnva)). This is the follow up on the tests presented at the OpenStack Barcelona (link (https://www.openstack.org/videos/video/toward-10000-containers-on-openstack)) and covered in the blog (https://developer.ibm.com/opentech/2016/11/16/how-do-containers-scale-on-openstack/) from IBM. The work has been helped by collaborations with Rackspace in the framework of the CERN openlab (http://openlab.cern/about/industry_members/rackspace) and the European Union Horizon 2020 Indigo Datacloud (https://www.indigo-datacloud.eu/) project.

Performance

At the Barcelona summit, we presented with Rackspace and IBM regarding our additional performance tests after the previous blog post (https://openstack-in-production.blogspot.ch/2016/06/scaling-magnum-and-kubernetes-2-million.html). We expanded beyond the 2M requests/s to reach around 7M where some network infrastructure issues unrelated to OpenStack limited the scaling further.



(https://4.bp.blogspot.com/-FnyR5mHq6Yg/WFJz3AF9JYI/AAAAAAAATWc/DseQzkvPwocfG2N_J1FVUz2HzjYg1ScLQCLcB/s1600/kub7m.png)

As we created the clusters, the deployment time increased only slightly with the number of nodes as most of the work is done in parallel. But for 128 node or larger clusters, the increase in time started to scale almost linearly. At the Barcelona summit, the Heat and Magnum teams worked together to develop proposals for how to improve further in future releases, although a 1000 node cluster in 23 minutes is still a good result

Cluster Size (Nodes)	Concurrency	Deployment Time (min)
2	50	2.5
16	10	4
32	10	4
128	5	5.5
512	1	14
1000	1	23

Storage

With the LHC producing nearly 50PB this year, High Energy Physics has some custom storage technologies for specific purposes, EOS for physics data, CVMFS for read-only, highly replicated storage such as applications.

One of the features of providing a private cloud service to the CERN users is to combine the functionality of open source community software such as OpenStack with the specific needs for high energy physics. For these to work, some careful driver work is needed to ensure appropriate access while ensuring user rights. In particular,

- EOS (https://eos.web.cern.ch/content/about-eos) provides a disk-based storage system providing high-capacity and low-latency access for users at CERN. Typical use cases are where scientists are analysing data from the experiments.
- CVMFS (https://cernvm.cern.ch/portal/filesystem) is used for a scalable, reliable and low-maintenance for read-only data such as software.

There are also other storage solutions we use at CERN such as

- HDFS (http://clouddocs.web.cern.ch/clouddocs/containers/tutorials/hdfs.html) for long term archiving of data using Hadoop which uses an HDFS driver within the container. HDFS works in user space, so no particular integration was required to use it from inside (unprivileged) containers
- Cinder provides additional disk space using volumes if the basic flavor does not have sufficient. This Cinder integration is offered by upstream Magnum, and work was done in the last OpenStack cycle to improve security by adding support for Keystone trusts.

CVMFS was more straightforward as there is no need to authenticate the user. The data is read-only and can be exposed to any container. The access to the file system is provided using a driver (link (https://gitlab.cern.ch/cloud-infrastructure/docker-volume-cvmfs/tree/dockerize)) which has been adapted to run inside a

container. This saves having to run additional software inside the VM hosting the container.
EOS requires authentication through mechanisms such as Kerberos to identify the user and thus determine what files they have access to. Here a container is run per user so that there is no risk of credential sharing. The details are in the driver (link (https://gitlab.cern.ch/cloud-infrastructure/docker-volume-eos)).

Service model

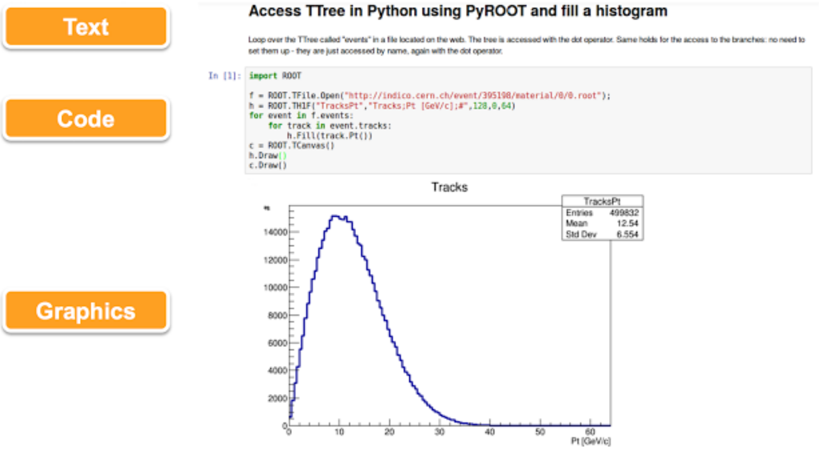
One interesting question that came up during the discussions of the container service was how to deliver the service to the end users. There are several scenarios:

1. The end user launches a container engine with their specifications but they rely on the IT department to maintain the engine availability. This implies that the VMs running the container engine are not accessible to the end user.
2. The end user launches the engine within a project that they administer. While the IT department maintains the templates and basic functions such as the Fedora Atomic images, the end user is in control of the upgrades and availability.
3. A variation of option 2., where the nodes running containers are reachable and managed by the end user, but the container engine master nodes are managed by the IT department. This is similar to the current offer from the Google Container Engine and requires some coordination and policies regarding upgrades

Currently, the default Magnum model is for the 2nd option and adding option 3 is something we could do in the near future. As users become more interested in consuming containers, we may investigate the 1st option further

Applications

Many applications at use in CERN are in the process of being reworked for a microservices based architecture. A choice of different container engines is attractive for the software developer. One example of this is the file transfer service which ensures that the network to other high energy physics sites is kept busy but not overloaded with data transfers. The work to containerise this application was described at the recent CHEP 2016 (http://chep2016.org/) FTS poster (https://indico.cern.ch/event/505613/contributions/2227329/attachments/1343962/2035344/Poster-37.pdf).
While deploying containers is an area of great interest for the software community, the key value comes from the physics applications exploiting containers to deliver a new way of working. The Swan (https://indico.cern.ch/event/517393/contributions/2150317/attachments/1269202/1879983/SWAN_SFTmeeting_090516.pdf#search=swan%20openstack) project provides a tool for running ROOT (https://root.cern.ch/), the High Energy Physics application framework, in a browser with easy access to the storage outlined above. A set of examples can be found at https://swan.web.cern.ch/notebook-galleries (https://swan.web.cern.ch/notebook-galleries). With the academic paper, the programs used and the data available from the notebook, this allows easy sharing with other physicists during the review process using CERNBox (https://indico.cern.ch/event/538540/contributions/2187138/attachments/1282513/1906054/IT-cernbox-2016-05-31.pdf), CERN's owncloud (https://owncloud.org/) based file sharing solution.



Access TTree in Python using PyROOT and fill a histogram

Loop over the TTree called "events" in a file located on the web. The tree is accessed with the dot operator. Same holds for the access to the branches: no need to set them up - they are just accessed by name, again with the dot operator.

In [1]:
```
import ROOT

f = ROOT.TFile.Open("http://indico.cern.ch/event/395198/material/0/0.root");
h = ROOT.TH1F("TracksPt","Tracks;Pt [GeV/c];#",128,0,64)
for event in f.events:
    for track in event.tracks:
        h.Fill(track.Pt())
c = ROOT.TCanvas()
h.Draw()
c.Draw()
```

(https://3.bp.blogspot.com/-rq_LuYL9fdI/WF1EdiyK3eI/AAAAAAAATXI/UHlrNC9p_xocw5Ft-e9LfehlYvdN3brjgCLcB/s1600/swangrab.png)
Another application being studied is http://opendata.cern.ch/?ln=en (http://opendata.cern.ch/?ln=en) which allows the general public to run analyses on LHC open data. Typical applications are Citizen Science (https://en.wikipedia.org/wiki/Citizen_science) and outreach for schools.

Ongoing work

There are a few major items where we are working with the upstream community:

- Cluster upgrades will allow us to upgrade the container software. Examples of this would be a new version of Fedora Atomic, Docker or the container engine. With a load balancer, this can be performed without downtime (spec (https://review.openstack.org/#/c/392193))
- Heterogeneous cluster support will allow nodes to have different flavors (cpu vs gpu, different i/o patterns, different AZs for improved failure scenarios). This is done by splitting the cluster nodes into node groups (blueprint (https://blueprints.launchpad.net/magnum/+spec/nodegroups))
- Cluster monitoring to deploy Prometheus and cAdvisor with Grafana dashboards for easy monitoring of a Magnum cluster (blueprint (https://blueprints.launchpad.net/magnum/+spec/container-monitoring)).

References

- End user documentation for containers on the CERN cloud at http://clouddocs.web.cern.ch/clouddocs/containers/index.html (http://clouddocs.web.cern.ch/clouddocs/containers/index.html)
- CERN IT department information is at http://cern.ch/it (http://cern.ch/it).
- CERN openlab (http://openlab.cern/about/industry_members/rackspace) Rackspace collaboration on container presentations are listed here (http://openlab.cern/about/industry_members/rackspace).
- Indigo Datacloud project details are here (https://www.indigo-datacloud.eu/).

This post first appeared on the OpenStack in Production (http://openstack-in-production.blogspot.com/2017/01/containers-on-cern-cloud.html) blog. Superuser is always interested in community content, email: editor@superuser.org (mailto:editor@superuser.org).

Cover Photo // CC BY NC (https://creativecommons.org/licenses/by-nc/2.0/)

The post Containers on the CERN Cloud (http://superuser.openstack.org/articles/containers-cern-cloud/) appeared first on OpenStack Superuser (http://superuser.openstack.org).

by Tim Bell at February 17, 2017 12:21 PM (http://superuser.openstack.org/articles/containers-cern-cloud/)

OpenStack Blog (https://www.openstack.org/blog)

User Group Newsletter February 2017 (https://www.openstack.org/blog/2017/02/user-group-newsletter-february-2017/)

Welcome to 2017! We hope you all had a lovely festive season. Here is our first edition of the User Group newsletter for this year.

AMBASSADOR PROGRAM NEWS

2017 sees some new arrivals and departures to our Ambassador program. Read about them here (http://superuser.openstack.org/articles/openstack-ambassadors/).

WELCOME TO OUR NEW USER GROUPS

We have some new user groups which have joined the OpenStack community.

Bangladesh (https://groups.openstack.org/groups/bangladesh)

Ireland – Cork (https://groups.openstack.org/groups/ireland-cork)

Russia – St Petersburg (https://groups.openstack.org/groups/russia-saint-petersburg)

United States – Phoenix, Arizona (https://groups.openstack.org/groups/phoenix)

Romania – Bucharest (https://groups.openstack.org/groups/romania-bucharest)

We wish them all the best with their OpenStack journey and can't wait to see what they will achieve!

Looking for a your local group? Are you thinking of starting a user group? Head to the groups portal for more information. (https://groups.openstack.org/)

MAY 2017 OPENSTACK SUMMIT

We're going to Boston for our first summit of 2017!!

You can register and stay updated here (https://www.openstack.org/summit/boston-2017/).

Consider it your pocket guide for all things Boston summit. Find out about the featured speakers (https://www.openstack.org/summit/boston-2017/travel/#travel-support), make your hotel bookings (https://www.openstack.org/summit/boston-2017/travel/#hotels), find your FAQ (https://www.openstack.org/summit/boston-2017/faq/) and read about our travel support program. (https://wiki.openstack.org/wiki/Travel_Support_Program)

NEW BOARD OF DIRECTORS
The community has spoken! A new board of directors has been elected for 2017.
Read all about it here. (http://www.businesswire.com/news/home/20170116005616/en/OpenStack-Community-Elects-2017-Board-Directors)

MAKE YOUR VOICE HEARD!
Submit your response the latest OpenStack (https://twitter.com/hashtag/OpenStack?src=hash) User Survey!
All data is completely confidential. Submissions close on the 20th of February 2017.
You can complete it here. (http://openstack.org/user-survey)

(https://t.co/Ysizr1yTNb)

CONTRIBUTING TO UG NEWSLETTER

If you'd like to contribute a news item for next edition, please submit to this etherpad (https://etherpad.openstack.org/p/User_Group_Newsletter).
Items submitted may be edited down for length, style and suitability.
This newsletter is published on a monthly basis.

by Sonia Ramza at February 17, 2017 04:47 AM (https://www.openstack.org/blog/2017/02/user-group-newsletter-february-2017/)

hastexo (https://www.hastexo.com/)

Importing an existing Ceph RBD image into Glance (https://www.hastexo.com/resources/hints-and-kinks/importing-rbd-into-glance/)

The normal process of uploading an image into Glance is straightforward: you use `glance image-create` or `openstack image create`, or the Horizon dashboard. Whichever process you choose, you select a local file, which you upload into the Glance image store.

This process can be unpleasantly time-consuming when your Glance service is backed with Ceph RBD, for a practical reason. When using the `rbd` image store, you're expected to use `raw` images, which have interesting characteristics.

Raw images and sparse files

Most people will take an existing vendor cloud image, which is typically available in the `qcow2` format, and convert it using the `qemu-img` utility, like so:

```
$ wget -O ubuntu-xenial.qcow2 \
  https://cloud-images.ubuntu.com/xenial/current/xenial-server-cloudimg-amd64-disk1.img
$ qemu-img convert -p -f qcow2 -O raw ubuntu-xenial.qcow2 ubuntu-xenial.raw
```

On face value, the result looks innocuous enough:

```
$ qemu-img info ubuntu-xenial.qcow2
image: ubuntu-xenial.qcow2
file format: qcow2
virtual size: 2.2G (2361393152 bytes)
disk size: 308M
cluster_size: 65536
Format specific information:
    compat: 0.10
    refcount bits: 16

$ qemu-img info ubuntu-xenial.raw
image: ubuntu-xenial.raw
file format: raw
virtual size: 2.2G (2361393152 bytes)
disk size: 1000M
```

As you can see, in both cases the virtual image size differs starkly from the actual file size. In `qcow2`, this is due to the copy-on-write nature of the file format and zlib compression; for the `raw` image, we're dealing with a sparse file:

```
$ ls -lh ubuntu-xenial.qcow2
-rw-rw-r-- 1 florian florian 308M Feb 17 10:05 ubuntu-xenial.qcow2
$ du -h  ubuntu-xenial.qcow2
308M    ubuntu-xenial.qcow2
$ ls -lh info ubuntu-xenial.raw
-rw-r--r-- 1 florian florian 2.2G Feb 17 10:16 ubuntu-xenial.raw
$ du -h  ubuntu-xenial.raw
1000M   ubuntu-xenial.raw
```

So, while the `qcow2` file's physical and logical sizes match, the `raw` file looks much larger in terms of filesystem metadata, as opposed to its actual storage utilization. That's because in a sparse file, "holes" (essentially, sequences of null bytes) aren't actually written to the filesystem. Instead, the filesystems just records the position and length of each "hole", and when we read from the "holes" in the file, the read would just return null bytes again.

The trouble with sparse files is that RESTful web services, like Glance, don't know too much about them. So, if we were to import that raw file with `openstack image-create --file my_cloud_image.raw`, the command line client would upload null bytes with happy abandon, which would greatly lengthen the process.

Importing images into RBD with `qemu-img convert`

Luckily for us, `qemu-img` also allows us to upload *directly* into RBD. All you need to do is make sure the image goes into the correct pool, and is reasonably named. Glance names uploaded images by there image ID, which is a universally unique identifier (UUID), so let's follow Glance's precedent.

```
export IMAGE_ID=`uuidgen`
export POOL="glance-images"  # replace with your Glance pool name

qemu-img convert \
  -f qcow2 -O raw \
  my_cloud_image.raw \
  rbd:$POOL/$IMAGE_ID
```

Creating the clone baseline snapshot

Glance expects a snapshot named `snap` to exist on any image that is subsequently cloned by Cinder or Nova, so let's create that as well:

```
rbd snap create rbd:$POOL/$IMAGE_ID@snap
rbd snap protect rbd:$POOL/$IMAGE_ID@snap
```

Making Glance aware of the image

Finally, we can let Glance know about this image. Now, there's a catch to this: this trick *only* works with the Glance v1 API, and thus you *must* use the `glance` client to do it. Your Glance is v2 only? Sorry. Insist on using the `openstack` client? Out of luck.

What's special about this invocation of the `glance` client are simply the pre-populated `location` and `id` fields. The `location` is composed of the following segments:

- the fixed string `rbd://`,
- your Ceph cluster UUID (you get this from `ceph fsid`),
- a forward slash (`/`),
- the name of your image (which you previously created with `uuidgen`),
- another forward slash (`/`, not `@` as you might expect),
- and finally, the name of your snapshot (`snap`).

Other than that, the `glance` client invocation is pretty straightforward for a v1 API call:

```
CLUSTER_ID=`ceph fsid`
glance --os-image-api-version 1 \
  image-create \
  --disk-format raw \
  --id $IMAGE_ID \
  --location rbd://$CLUSTER_ID/$IMAGE_ID/snap
```

Of course, you might add other options, like `--private` or `--protected` or `--name`, but the above options are the bare minimum.

And that's it!

Now you can happily fire up VMs, or clone your image into a volume and fire a VM up from that.

by hastexo at February 17, 2017 12:00 AM (https://www.hastexo.com/resources/hints-and-kinks/importing-rbd-into-glance/)

February 16, 2017

Ed Leafe (https://blog.leafe.com)

Interop API Requirements (https://blog.leafe.com/interop-api-requirements/)

Lately the OpenStack Board of Directors and Technical Committee has placed a lot of emphasis on making OpenStack clouds from various providers “interoperable”. This is a very positive development, after years of different deployments adding various extensions and modifications to the upstream OpenStack code, which had made it hard to define just what it means to offer an “OpenStack Cloud”. So the Interop project (https://www.openstack.org/brand/interop/) (formerly known as **DefCore**) has been working for the past few years to create a series of objective tests that cloud deployers can run to verify that their cloud meets these interoperability standards.

As a member of the OpenStack API Working Group (https://wiki.openstack.org/wiki/API_Working_Group), though, I’ve had to think a lot about what interop means for an API. I’ll sum up my thoughts, and then try to explain why.

> API Interoperability requires that all identical API calls return identical results when made to the same API version on all OpenStack clouds.

This may seem obvious enough, but it has implications that go beyond our current API guidelines. For example, we currently don’t recommend a version increase (http://specs.openstack.org/openstack/api-wg/guidelines/evaluating_api_changes.html) for changes that add things, such as an additional header or a new URL. After all, no one using the current version will be hurt by this, since they aren’t expecting those new things, and so their code cannot break. But this only considers the effect on a single cloud; when we factor in interoperability, things look very different.

Let’s consider the case where we have two OpenStack-based clouds, both running version 42 of an API. Cloud A is running the released version of the code, while Cloud B is tracking upstream master, which has recently added a new URL (which in the past we’ve said is OK). If we called that new URL on Cloud A, it will return a 404, since that URL had not been defined in the released version of the code. On Cloud B, however, since it is defined on the current code, it will return anything *except* a 404. So we have two clouds claiming to be running the same version of OpenStack, but making identical calls to them has very different results.

Note that when I say “identical” results, I mean structural things, such as response code, format of any body content, and response headers. I *don’t* mean that it will list the same resources, since it is expected that you can create different resources at will.

I’m sure this will be discussed further at next week’s PTG (https://www.openstack.org/ptg/).

by ed at February 16, 2017 11:30 PM (https://blog.leafe.com/interop-api-requirements/)

Cloudwatt (https://dev.cloudwatt.com/en/blog/index.html)

5 Minutes Stacks, épisode 53 : iceScrum (https://dev.cloudwatt.com/en/blog/5-minutes-stacks-episode-fifty-three-icescrum.html)

Episode 53 : iceScrum

iceScrum is a project management tool following “agile” method.

This tool will allow you to have a global preview of your project, and hence the analyses and the productivity.

A friendly dashboard shows useful indicators for the setting up of your project or the few last changes which were made.

iceScrum is fully available through an internet browser and it uses a MySQL database to store all its informations.

Preparations

The Versions

- CoreOS Stable 1185.5
- iceScrum R6#14.11

The prerequisites to deploy this stack

These should be routine by now:

- An Internet access

- A Linux shell
- A Cloudwatt account (https://www.cloudwatt.com/cockpit/#/create-contact) with a valid keypair (https://console.cloudwatt.com/project/access_and_security/?tab=access_security_tabs__keypairs_tab)
- The tools of the trade: OpenStack CLI (http://docs.openstack.org/cli-reference/content/install_clients.html)
- A local clone of the Cloudwatt applications (https://github.com/cloudwatt/applications) git repository (if you are creating your stack from a shell)

Size of the instance

By default, the stack deploys on an instance of type “Standard 1” (n1.cw.standard-1). A variety of other instance types exist to suit your various needs, allowing you to pay only for the services you need. Instances are charged by the minute and capped at their monthly price (you can find more details on the Pricing page (https://www.cloudwatt.com/en/pricing.html) on the Cloudwatt website).

Stack parameters, of course, are yours to tweak at your fancy.

By the way…

If you do not like command lines, you can go directly to the “run it thru the console” section by clicking here (https://dev.cloudwatt.com/en/blog/rss.xml#console)

What will you find in the repository

Once you have cloned the github, you will find in the `blueprint-coreos-icescrum/` repository:

- `blueprint-coreos-icescrum.heat.yml` : HEAT orchestration template. It will be use to deploy the necessary infrastructure.
- `stack-start.sh` : Stack launching script. This is a small script that will save you some copy-paste.
- `stack-get-url.sh` : Flotting IP recovery script.

Start-up

Initialize the environment

Have your Cloudwatt credentials in hand and click HERE (https://console.cloudwatt.com/project/access_and_security/api_access/openrc/). If you are not logged in yet, you will go thru the authentication screen then the script download will start. Thanks to it, you will be able to initiate the shell accesses towards the Cloudwatt APIs.

Source the downloaded file in your shell. Your password will be requested.

```
$ source COMPUTE-[...]-openrc.sh
Please enter your OpenStack Password:
```

Once this done, the Openstack command line tools can interact with your Cloudwatt user account.

Adjust the parameters

With the `blueprint-coreos-icescrum.heat.yml` file, you will find at the top a section named `parameters` . The sole mandatory parameter to adjust is the one called `keypair_name` . Its `default` value must contain a valid keypair with regards to your Cloudwatt user account. This is within this same file that you can adjust the instance size by playing with the `flavor` parameter.

```
heat_template_version: 2015-04-30


description: Blueprint iceScrum


parameters:
  keypair_name:
    description: Keypair to inject in instance
    label: SSH Keypair
    type: string

  flavor_name:
    default: n1.cw.standard-1
    description: Flavor to use for the deployed instance
    type: string
    label: Instance Type (Flavor)
    constraints:
      - allowed_values:
          - n1.cw.standard-1
          - n1.cw.standard-2
          - n1.cw.standard-4
          - n1.cw.standard-8
          - n1.cw.standard-12
          - n1.cw.standard-16

  sqlpass:
    description: password root sql
    type: string
    hidden: true

[...]
```

Start stack

In a shell, run the script `stack-start.sh` with his name in parameter:

```
./stack-start.sh iceScrum
+--------------------------------------+-----------------+--------------------+----------------------+
| id                                   | stack_name      | stack_status       | creation_time        |
+--------------------------------------+-----------------+--------------------+----------------------+
| ee873a3a-a306-4127-8647-4bc80469cec4 | iceScrum        | CREATE_IN_PROGRESS | 2015-11-25T11:03:51Z |
+--------------------------------------+-----------------+--------------------+----------------------+
```

Within **5 minutes** the stack will be fully operational. (Use `watch` to see the status in real-time)

```
 $ watch heat resource-list iceScrum
 +------------------+-----------------------------------------------------+----------------------------------+-----------------+----------------
------+
 | resource_name    | physical_resource_id                                | resource_type                    | resource_status | updated_time
      |
 +------------------+-----------------------------------------------------+----------------------------------+-----------------+----------------
------+
 | floating_ip      | 44dd841f-8570-4f02-a8cc-f21a125cc8aa                | OS::Neutron::FloatingIP          | CREATE_COMPLETE | 2015-11-25T11:0
3:51Z |
 | security_group   | efead2a2-c91b-470e-a234-58746da6ac22                | OS::Neutron::SecurityGroup       | CREATE_COMPLETE | 2015-11-25T11:0
3:52Z |
 | network          | 7e142d1b-f660-498d-961a-b03d0aee5cff                | OS::Neutron::Net                 | CREATE_COMPLETE | 2015-11-25T11:0
3:56Z |
 | subnet           | 442b31bf-0d3e-406b-8d5f-7b1b6181a381                | OS::Neutron::Subnet              | CREATE_COMPLETE | 2015-11-25T11:0
3:57Z |
 | server           | f5b22d22-1cfe-41bb-9e30-4d089285e5e5                | OS::Nova::Server                 | CREATE_COMPLETE | 2015-11-25T11:0
4:00Z |
 | floating_ip_link | 44dd841f-8570-4f02-a8cc-f21a125cc8aa-`floating IP`  | OS::Nova::FloatingIPAssociation  | CREATE_COMPLETE | 2015-11-25T11:0
4:30Z |
   +------------------+-----------------------------------------------------+--------------------------------+-----------------+----------------
------
```

The `start-stack.sh` script takes care of running the API necessary requests to execute the normal heat template which:

- Starts an CoreOS based instance with the docker container *iceScrum* and the container *MySQL*
- Expose it on the Internet via a floating IP.

All of this is fine, but…

You do not have a way to create the stack from the console?

We do indeed! Using the console, you can deploy iceScrum:

1. Go the Cloudwatt Github in the applications/blueprint-coreos-icescrum (https://github.com/cloudwatt/applications/tree/master/blueprint-coreos-icescrum) repository
2. Click on the file named `blueprint-coreos-icescrum.heat.yml`
3. Click on RAW, a web page will appear containing purely the template
4. Save the file to your PC. You can use the default name proposed by your browser (just remove the .txt)
5. Go to the « Stacks (https://console.cloudwatt.com/project/stacks/) » section of the console
6. Click on « Launch stack », then « Template file » and select the file you just saved to your PC, and finally click on « NEXT »
7. Name your stack in the « Stack name » field
8. Enter the name of your keypair in the « SSH Keypair » field
9. Write a passphrase that will be used for the database icescrum user
10. Choose your instance size using the « Instance Type » dropdown and click on « LAUNCH »

The stack will be automatically generated (you can see its progress by clicking on its name). When all modules become green, the creation will be complete. You have to wait 5 minutes to the softwares be ready. You can then go to the "Instances" menu to find the floating IP, or simply refresh the current page and check the Overview tab for a handy link.

If you've reached this point, you're already done! Go enjoy iceScrum!

A one-click deployment sounds really nice…

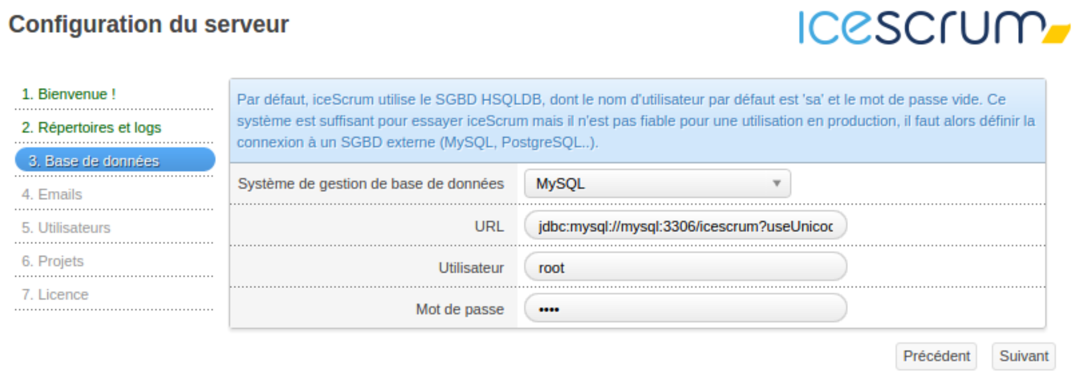
… Good! Go to the Apps page (https://www.cloudwatt.com/en/apps/) on the Cloudwatt website, choose the apps, press **DEPLOY** and follow the simple steps… 2 minutes later, a green button appears… **ACCESS**: you have your e-commerce platform.

Enjoy

Once all this makes you can connect on your server in SSH by using your keypair beforehand downloaded on your compute,

You are now in possession of iceScrum, you can enter via the URL `http://ip-floatingip`. Your full URL will be present in your stack overview in horizon Cloudwatt console.

At your first connexion you will ask to give some information and how to access to the database. Complete the fields as below (MySQL's URL is `jdbc:mysql://mysql:3306/icescrum?useUnicode=true&characterEncoding=utf8`), the password is which one you chose when you created the stack.



Configuration du serveur

Par défaut, iceScrum utilise le SGBD HSQLDB, dont le nom d'utilisateur par défaut est 'sa' et le mot de passe vide. Ce système est suffisant pour essayer iceScrum mais il n'est pas fiable pour une utilisation en production, il faut alors définir la connexion à un SGBD externe (MySQL, PostgreSQL..).

Système de gestion de base de données	MySQL
URL	jdbc:mysql://mysql:3306/icescrum?useUnicoc
Utilisateur	root
Mot de passe	••••

When setup is completed, you have to restart iceScrum with the following command: `docker restart icescrum`

So watt?

The goal of this tutorial is to accelerate your start. At this point **you** are the master of the stack.

You now have an SSH access point on your virtual machine through the floating-IP and your private keypair (default userusername `core`).

- You have access to the web interface via the address specified in your output stack in horizon console.
- Here are some news sites to learn more:
  - https://www.icescrum.com/
  - https://www.icescrum.com/documentation/

Have fun. Hack in peace.

by Simon Decaestecker at February 16, 2017 11:00 PM (https://dev.cloudwatt.com/en/blog/5-minutes-stacks-episode-fifty-three-icescrum.html)

NFVPE @ Red Hat (https://blog.nfvpe.site)

Let's spin up k8s 1.5 on CentOS (with CNI pod networking, too!) (http://dougbtv.com//nfvpe/2017/02/16/kubernetes-1.5-centos/)

Alright, so you've seen my blog post about installing Kubernetes by hand (http://dougbtv.com/nfvpe/2017/02/09/kubernetes-on-centos/) on CentOS, now… Let's make that easier and do that with an Ansible playbook, specifically my kube-centos-ansible (https://github.com/dougbtv/kube-centos-ansible) playbook. This time we'll have Kubernetes 1.5 running on a cluster of 3 VMs, and we'll use weave as a CNI plugin to handle our pod network. And to make it more fun, we'll even expose some pods to the 'outside world', so we can actually (kinda) do something with them. Ready? Let's go!

by Doug Smith at February 16, 2017 08:00 PM (http://dougbtv.com//nfvpe/2017/02/16/kubernetes-1.5-centos/)

OpenStack Superuser (http://superuser.openstack.org)

Why commercial open source is like a voyage to mars: The Kubernetes story (http://superuser.openstack.org/articles/mars-open-source-kubernetes/)

LAKE TAHOE, CALIF. — Commercial open source is like the planet Mars: a fascinating environment that holds tremendous promise. It can also be harsh, sterile and difficult to navigate, say two veteran explorers.

Craig McLuckie (https://twitter.com/@cmcluck), founder of Kubernetes and now CEO of Heptio (https://www.heptio.com/), and Sarah Novotny (https://twitter.com/@sarahnovotny) currently leads the Kubernetes Community Program (https://kubernetes.io/community/) for Google. Kubernetes, an open source container cluster manager originally designed by Google in 2014, has been called a "secret weapon" in cloud computing.

The pair told a story about the emergence of "open source as a force in the world" as well as a cautionary tale for those embarking on a journey into the far reaches of commercial open source at the Linux Foundation's Open Source leadership summit. (http://events.linuxfoundation.org/events/open-source-leadership-summit)

<figure class="wp-caption alignnone" id="attachment_5487" style="width: 605px">

Why this matters
(and what is with the Mars theme?)
Software is increasingly central to success of any enterprise
OSS is changing the relationship between enterprises and technology
Engagement level
Buying criteria
Cloud is changing the way that OSS is monetized
Nowhere more true that infrastructure software
This has potentially significant impact and is worth understanding
Particularly if you are running a community
And doubly so if you are building a company around an OSS technology
Monetization of OSS is not without its challenges
(like going to Mars, and staying alive when you get there)
OPEN SOURCE
LEADERSHIP SUMMIT

(http://superuser.openstack.org/wp-content/uploads/2017/02/saraj.jpeg)<figcaption class="wp-caption-text"> Craig McLuckie and Sarah Novotny at the Linux Leadership Summit. </figcaption>
</figure>

They first took a look at the current landscape: these days, software is increasingly central to the success of any business and open source software is changing the relationship between enterprises and technology. More progressive companies — including banks – are changing the way they engage with software, McLuckie says. They want to put resources into it to make it better, to make it theirs, behave the way they need to behave and that ripples into commercial buying decisions.

"You go to a lot of these organizations and increasingly they start to say, 'Hey, this is open source' and if it's not, they're not interested," McLuckie says. "If you're not an open source company, it's hard times."

Open source has also been the infrastructure that the internet has used for years and years but as the cloud has changed the infrastructure, and everything becomes a service, infrastructure software is being built in open source and deployed and monetized through cloud providers. "That really and fundamentally has changed how you engage with open source software and how you engage as open source developers," Novotny says.

> Sinha: Ownership in Kubernetes means we encourage everyone to take on a role, and you must respect whoever has the role. #lfosls (https://twitter.com/hashtag/lfosls?src=hash)
>
> — APESOL (@APESOL) February 15, 2017 (https://twitter.com/APESOL/status/831937460208857088)

<script async="async" charset="utf-8" src="http://platform.twitter.com/widgets.js"></script>

Cloud has changed the way open source software is monetized. When people ask McLuckie what was Google's intent with Kubernetes, he answers, "We're going to build something awesome. We're going to monetize the heck out of hosting it. That's it. That was the plan. It actually worked out really well." The result was a strong community and a quickly growing product.

That impact is worth understanding, particularly if you're running a community — doubly so if you're building a company around an open source technology.

"We're all in the open source world together," he says, adding that there is "no finer mechanism for the manifestation of OS than Puppet cloud right now" and citing the example of RDS which is effectively a strategy to monetize MySQL. "It's very difficult to point to something more successful than Amazon's ability to mark up the cost of its infrastructure, overlay a technology like MySQL's technology and then deliver a premium on it. This is incredibly powerful."

Monetization is not without its challenges — "like going to Mars and staying alive when you get there," McLuckie says. There's an obvious tension in commercial open source between the need for control and the need to build and sustain an open ecosystem. "If you're thinking about building a technology and building a business where your business is value extraction from the open source community, it's going to be interesting. You're going to have some interesting problems."

McLuckie's admittedly "non-scientific anecdata" theory is that the reality on return on effort for (distros, training, consulting) for many startups in open source can be "bleak." The first way that startups tend to think about monetization is to create a distro and sell support. This is an easy way to get things rolling: the community finds it, they want to use it, very few people have the expertise to run it and you can make good money packaging it and selling it. But it doesn't take long until another distro — almost identical — comes out at half the price. It's easy for your customers to switch and the rest of your value proposition (getting the technology working, getting it through the community) is now making it easier for your competitor to undercut you. This is also true of licensing — where if your fees are high enough, companies start thinking internal engineers are cheaper and can customize more — professional services and training.

Survival is about being lean, McLuckie says. Smart startups can eke out a living if they don't burn, say 50 percent of operating budget on customer aquisition and "look hard" for economies of scale.

How did Kubernetes navigate this territory? "We put together something that's special. We created this back pressure, that will fight back from monetization," McLuckie says . "The most focused way to do this is lock down a community. All organizations belong to us. If you want a fix, it's coming through our computer base."

Leadership matters, it's a powerful and healthy form of control. The pair said that when they look at the biggest gap in their communities it's not the contributors but leaders who can help contributors succeed. Leadership is also about the workaday tasks of managing releases, paying down the "community tax," and being part of the team first.

"It's easy to get on the wrong side of the community. It's hard to get the unit economics right. Make sure you really think through the future of where this is going. And designing an organization, designing a good market strategy that's grounded in the economics of the day," he says.

"It's also about being really smart," Novotny says. "If you are a business, and you think you your business is value extraction from an open-source community. You're going to have hard times. You cannot take more out of an open-source community that you put in."

> Let’s be smart. Keep moving forward. Keep collaborating. #lfosls (https://twitter.com/hashtag/lfosls?src=hash) pic.twitter.com/jpOskkSHo4 (https://t.co/jpOskkSHo4)
>
> — Rochelle Grober (@GroberRocky) February 14, 2017 (https://twitter.com/GroberRocky/status/831591661993996288)

<script async="async" charset="utf-8" src="http://platform.twitter.com/widgets.js"></script>

Be adaptable, she adds, ss the technology changes, as the landscape changes, as the cultural changes. “We’ve seen all of these cultural shifts, they all have threads that carry that, and now one of our favorite cultural shifts, of course, is Cloud Native. And that has such a strong expectation of being mobile. Mobile in the sense of not locked into any particular vendor, while still being able to get your- any you-cases service to the best possible execution of your engine. So my hope is that out of this, we will see open-source into across all of the very work that we need in our communities.

Above all, the key is to sell your vision of the future – new territories, unexplored lands.

“The technology is a tool,” McLuckie says. “If you want to create a business, the business has to be about how you use it. You have to sell the dream. You have to think about ways in which that technology is transforming in other people’s businesses.”

Cover photo by: Pascal (http://superuser.openstack.org/feed/%26quot%3Bhttps%3A//www.flickr.com/photos/pasukaru76/4999924988//%E2%80%9C)

The post Why commercial open source is like a voyage to mars: The Kubernetes story (http://superuser.openstack.org/articles/mars-open-source-kubernetes/) appeared first on OpenStack Superuser (http://superuser.openstack.org).

by Nicole Martinelli at February 16, 2017 01:12 PM (http://superuser.openstack.org/articles/mars-open-source-kubernetes/)

Daniel P. Berrangé (https://www.berrange.com)

Setting up a nested KVM guest for developing & testing PCI device assignment with NUMA (https://www.berrange.com/posts/2017/02/16/setting-up-a-nested-kvm-guest-for-developing-testing-pci-device-assignment-with-numa/)

Over the past few years OpenStack Nova project has gained support for managing VM usage of NUMA, huge pages and PCI device assignment. One of the more challenging aspects of this is availability of hardware to develop and test against. In the ideal world it would be possible to emulate everything we need using KVM, enabling developers / test infrastructure to exercise the code without needing access to bare metal hardware supporting these features. KVM has long has support for emulating NUMA topology in guests, and guest OS can use huge pages inside the guest. What was missing were pieces around PCI device assignment, namely IOMMU support and the ability to associate NUMA nodes with PCI devices. Co-incidentally a QEMU community member was already working on providing emulation of the Intel IOMMU. I made a request to the Red Hat KVM team to fill in the other missing gap related to NUMA / PCI device association. To do this required writing code to emulate a PCI/PCI-E Expander Bridge (PXB) device, which provides a light weight host bridge that can be associated with a NUMA node. Individual PCI devices are then attached to this PXB instead of the main PCI host bridge, thus gaining affinity with a NUMA node. With this, it is now possible to configure a KVM guest such that it can be used as a virtual host to test NUMA, huge page and PCI device assignment integration. The only real outstanding gap is support for emulating some kind of SRIOV network device, but even without this, it is still possible to test most of the Nova PCI device assignment logic – we’re merely restricted to using physical functions, no virtual functions. This blog posts will describe how to configure such a virtual host.

First of all, this requires very new libvirt & QEMU to work, specifically you’ll want libvirt >= 2.3.0 and QEMU 2.7.0. We could technically support earlier QEMU versions too, but that’s pending on a patch to libvirt to deal with some command line syntax differences in QEMU for older versions. No currently released Fedora has new enough packages available, so even on Fedora 25, you must enable the “Virtualization Preview (https://fedoraproject.org/wiki/Virtualization_Preview_Repository)” repository on the physical host to try this out – F25 has new enough QEMU, so you just need a libvirt update.

```
# curl --output /etc/yum.repos.d/fedora-virt-preview.repo https://fedorapeople.org/groups/virt/virt-preview/fedora-virt-preview.repo (https://fed
# dnf upgrade
```

For sake of illustration I’m using Fedora 25 as the OS inside the virtual guest, but any other Linux OS will do just fine. The initial task is to install guest with 8 GB of RAM & 8 CPUs using virt-install

```
# cd /var/lib/libvirt/images
# wget -O f25x86_64-boot.iso https://download.fedoraproject.org/pub/fedora/linux/releases/25/Server/x86_64/os/images/boot.iso
# virt-install --name f25x86_64  \
    --file /var/lib/libvirt/images/f25x86_64.img --file-size 20 \
    --cdrom f25x86_64-boot.iso --os-type fedora23 \
    --ram 8000 --vcpus 8 \
    ...
```

The guest needs to use host CPU passthrough to ensure the guest gets to see VMX, as well as other modern instructions and have 3 virtual NUMA nodes. The first guest NUMA node will have 4 CPUs and 4 GB of RAM, while the second and third NUMA nodes will each have 2 CPUs and 2 GB of RAM. We are just going to let the guest float freely across host NUMA nodes since we don’t care about performance for dev/test, but in production you would certainly pin each guest NUMA node to a distinct host NUMA node.

```
    ...
    --cpu host,cell0.id=0,cell0.cpus=0-3,cell0.memory=4096000,\
               cell1.id=1,cell1.cpus=4-5,cell1.memory=2048000,\
               cell2.id=2,cell2.cpus=6-7,cell2.memory=2048000 \
    ...
```

QEMU emulates various different chipsets and historically for x86, the default has been to emulate the ancient PIIX4 (it is 20+ years old dating from circa 1995). Unfortunately this is too ancient to be able to use the Intel IOMMU emulation with, so it is neccessary to tell QEMU to emulate the marginally less ancient chipset Q35 (it is only 9 years old, dating from 2007).

```
    ...
    --machine q35
```

The complete virt-install command line thus looks like

```
# virt-install --name f25x86_64  \
    --file /var/lib/libvirt/images/f25x86_64.img --file-size 20 \
    --cdrom f25x86_64-boot.iso --os-type fedora23 \
    --ram 8000 --vcpus 8 \
    --cpu host,cell0.id=0,cell0.cpus=0-3,cell0.memory=4096000,\
               cell1.id=1,cell1.cpus=4-5,cell1.memory=2048000,\
               cell2.id=2,cell2.cpus=6-7,cell2.memory=2048000 \
    --machine q35
```

Once the installation is completed, shut down this guest since it will be necessary to make a number of changes to the guest XML configuration to enable features that virt-install does not know about, using “ `virsh edit` “. With the use of Q35, the guest XML should initially show three PCI controllers present, a “pcie-root”, a “dmi-to-pci-bridge” and a “pci-bridge”

```
<controller type='pci' index='0' model='pcie-root'/>
<controller type='pci' index='1' model='dmi-to-pci-bridge'>
  <model name='i82801b11-bridge'/>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x1e' function='0x0'/>
</controller>
<controller type='pci' index='2' model='pci-bridge'>
  <model name='pci-bridge'/>
  <target chassisNr='2'/>
  <address type='pci' domain='0x0000' bus='0x01' slot='0x00' function='0x0'/>
</controller>
```

PCI endpoint devices are not themselves associated with NUMA nodes, rather the bus they are connected to has affinity. The default pcie-root is not associated with any NUMA node, but extra PCI-E Expander Bridge controllers can be added and associated with a NUMA node. So while in edit mode, add the following to the XML config

```
<controller type='pci' index='3' model='pcie-expander-bus'>
  <target busNr='180'>
    <node>0</node>
  </target>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x0'/>
</controller>
<controller type='pci' index='4' model='pcie-expander-bus'>
  <target busNr='200'>
    <node>1</node>
  </target>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0'/>
</controller>
<controller type='pci' index='5' model='pcie-expander-bus'>
  <target busNr='220'>
    <node>2</node>
  </target>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x04' function='0x0'/>
</controller>
```

It is not possible to plug PCI endpoint devices directly into the PXB, so the next step is to add PCI-E root ports into each PXB – we’ll need one port per device to be added, so 9 ports in total. This is where the requirement for libvirt >= 2.3.0 – earlier versions mistakenly prevented you adding more than one root port to the PXB

```
<controller type='pci' index='6' model='pcie-root-port'>
  <model name='ioh3420'/>
  <target chassis='6' port='0x0'/>
  <alias name='pci.6'/>
  <address type='pci' domain='0x0000' bus='0x03' slot='0x00' function='0x0'/>
</controller>
<controller type='pci' index='7' model='pcie-root-port'>
  <model name='ioh3420'/>
  <target chassis='7' port='0x8'/>
  <alias name='pci.7'/>
  <address type='pci' domain='0x0000' bus='0x03' slot='0x01' function='0x0'/>
</controller>
<controller type='pci' index='8' model='pcie-root-port'>
  <model name='ioh3420'/>
  <target chassis='8' port='0x10'/>
  <alias name='pci.8'/>
  <address type='pci' domain='0x0000' bus='0x03' slot='0x02' function='0x0'/>
</controller>
<controller type='pci' index='9' model='pcie-root-port'>
  <model name='ioh3420'/>
  <target chassis='9' port='0x0'/>
  <alias name='pci.9'/>
  <address type='pci' domain='0x0000' bus='0x04' slot='0x00' function='0x0'/>
</controller>
<controller type='pci' index='10' model='pcie-root-port'>
  <model name='ioh3420'/>
  <target chassis='10' port='0x8'/>
  <alias name='pci.10'/>
  <address type='pci' domain='0x0000' bus='0x04' slot='0x01' function='0x0'/>
</controller>
<controller type='pci' index='11' model='pcie-root-port'>
  <model name='ioh3420'/>
  <target chassis='11' port='0x10'/>
  <alias name='pci.11'/>
  <address type='pci' domain='0x0000' bus='0x04' slot='0x02' function='0x0'/>
</controller>
<controller type='pci' index='12' model='pcie-root-port'>
  <model name='ioh3420'/>
  <target chassis='12' port='0x0'/>
  <alias name='pci.12'/>
  <address type='pci' domain='0x0000' bus='0x05' slot='0x00' function='0x0'/>
</controller>
<controller type='pci' index='13' model='pcie-root-port'>
  <model name='ioh3420'/>
  <target chassis='13' port='0x8'/>
  <alias name='pci.13'/>
  <address type='pci' domain='0x0000' bus='0x05' slot='0x01' function='0x0'/>
</controller>
<controller type='pci' index='14' model='pcie-root-port'>
  <model name='ioh3420'/>
  <target chassis='14' port='0x10'/>
  <alias name='pci.14'/>
  <address type='pci' domain='0x0000' bus='0x05' slot='0x02' function='0x0'/>|
</controller>
```

Notice that the values in ' `bus` ' attribute on the `<address>` element is matching the value of the ' `index` ' attribute on the `<controller>` element of the parent device in the topology. The PCI controller topology now looks like this

```
pcie-root (index == 0)
  |
  +- dmi-to-pci-bridge (index == 1)
  |    |
  |    +- pci-bridge (index == 2)
  |
  +- pcie-expander-bus (index == 3, numa node == 0)
  |    |
  |    +- pcie-root-port (index == 6)
  |    +- pcie-root-port (index == 7)
  |    +- pcie-root-port (index == 8)
  |
  +- pcie-expander-bus (index == 4, numa node == 1)
  |    |
  |    +- pcie-root-port (index == 9)
  |    +- pcie-root-port (index == 10)
  |    +- pcie-root-port (index == 11)
  |
  +- pcie-expander-bus (index == 5, numa node == 2)
       |
       +- pcie-root-port (index == 12)
       +- pcie-root-port (index == 13)
       +- pcie-root-port (index == 14)
```

All the existing devices are attached to the " `pci-bridge` " (the controller with `index` == 2). The devices we intend to use for PCI device assignment inside the virtual host will be attached to the new " `pcie-root-port` " controllers. We will provide 3 e1000 per NUMA node, so that's 9 devices in total to add

```
<interface type='user'>
  <mac address='52:54:00:7e:6e:c6'/>
  <model type='e1000e'/>
  <address type='pci' domain='0x0000' bus='0x06' slot='0x00' function='0x0'/>
</interface>
<interface type='user'>
  <mac address='52:54:00:7e:6e:c7'/>
  <model type='e1000e'/>
  <address type='pci' domain='0x0000' bus='0x07' slot='0x00' function='0x0'/>
</interface>
<interface type='user'>
  <mac address='52:54:00:7e:6e:c8'/>
  <model type='e1000e'/>
  <address type='pci' domain='0x0000' bus='0x08' slot='0x00' function='0x0'/>
</interface>
<interface type='user'>
  <mac address='52:54:00:7e:6e:d6'/>
  <model type='e1000e'/>
  <address type='pci' domain='0x0000' bus='0x09' slot='0x00' function='0x0'/>
</interface>
<interface type='user'>
  <mac address='52:54:00:7e:6e:d7'/>
  <model type='e1000e'/>
  <address type='pci' domain='0x0000' bus='0x0a' slot='0x00' function='0x0'/>
</interface>
<interface type='user'>
  <mac address='52:54:00:7e:6e:d8'/>
  <model type='e1000e'/>
  <address type='pci' domain='0x0000' bus='0x0b' slot='0x00' function='0x0'/>
</interface>
<interface type='user'>
  <mac address='52:54:00:7e:6e:e6'/>
  <model type='e1000e'/>
  <address type='pci' domain='0x0000' bus='0x0c' slot='0x00' function='0x0'/>
</interface>
<interface type='user'>
  <mac address='52:54:00:7e:6e:e7'/>
  <model type='e1000e'/>
  <address type='pci' domain='0x0000' bus='0x0d' slot='0x00' function='0x0'/>
</interface>
<interface type='user'>
  <mac address='52:54:00:7e:6e:e8'/>
  <model type='e1000e'/>
  <address type='pci' domain='0x0000' bus='0x0e' slot='0x00' function='0x0'/>
</interface>
```

Note that we're using the "user" networking, aka SLIRP. Normally one would never want to use SLIRP but we don't care about actually sending traffic over these NICs, and so using SLIRP avoids polluting our real host with countless TAP devices.

The final configuration change is to simply add the Intel IOMMU device

```
<iommu model='intel'/>
```

It is a capability integrated into the chipset, so it does not need any `<address>` element of its own. At this point, save the config and start the guest once more. Use the "virsh domifaddrs" command to discover the IP address of the guest's primary NIC and ssh into it.

```
# virsh domifaddr f25x86_64
 Name       MAC address          Protocol     Address
-------------------------------------------------------------------------------
 vnet0      52:54:00:10:26:7e    ipv4         192.168.122.3/24

# ssh root@192.168.122.3
```

We can now do some sanity check that everything visible in the guest matches what was enabled in the libvirt XML config in the host. For example, confirm the NUMA topology shows 3 nodes

```
# dnf install numactl
# numactl --hardware
available: 3 nodes (0-2)
node 0 cpus: 0 1 2 3
node 0 size: 3856 MB
node 0 free: 3730 MB
node 1 cpus: 4 5
node 1 size: 1969 MB
node 1 free: 1813 MB
node 2 cpus: 6 7
node 2 size: 1967 MB
node 2 free: 1832 MB
node distances:
node   0   1   2
  0:  10  20  20
  1:  20  10  20
  2:  20  20  10
```

Confirm that the PCI topology shows the three PCI-E Expander Bridge devices, each with three NICs attached

```
# lspci -t -v
-+-[0000:dc]-+-00.0-[dd]----00.0  Intel Corporation 82574L Gigabit Network Connection
 |           +-01.0-[de]----00.0  Intel Corporation 82574L Gigabit Network Connection
 |           \-02.0-[df]----00.0  Intel Corporation 82574L Gigabit Network Connection
 +-[0000:c8]-+-00.0-[c9]----00.0  Intel Corporation 82574L Gigabit Network Connection
 |           +-01.0-[ca]----00.0  Intel Corporation 82574L Gigabit Network Connection
 |           \-02.0-[cb]----00.0  Intel Corporation 82574L Gigabit Network Connection
 +-[0000:b4]-+-00.0-[b5]----00.0  Intel Corporation 82574L Gigabit Network Connection
 |           +-01.0-[b6]----00.0  Intel Corporation 82574L Gigabit Network Connection
 |           \-02.0-[b7]----00.0  Intel Corporation 82574L Gigabit Network Connection
 \-[0000:00]-+-00.0  Intel Corporation 82G33/G31/P35/P31 Express DRAM Controller
             +-01.0  Red Hat, Inc. QXL paravirtual graphic card
             +-02.0  Red Hat, Inc. Device 000b
             +-03.0  Red Hat, Inc. Device 000b
             +-04.0  Red Hat, Inc. Device 000b
             +-1d.0  Intel Corporation 82801I (ICH9 Family) USB UHCI Controller #1
             +-1d.1  Intel Corporation 82801I (ICH9 Family) USB UHCI Controller #2
             +-1d.2  Intel Corporation 82801I (ICH9 Family) USB UHCI Controller #3
             +-1d.7  Intel Corporation 82801I (ICH9 Family) USB2 EHCI Controller #1
             +-1e.0-[01-02]----01.0-[02]--+-01.0  Red Hat, Inc Virtio network device
             |                            +-02.0  Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) High Definition Audio Controller
             |                            +-03.0  Red Hat, Inc Virtio console
             |                            +-04.0  Red Hat, Inc Virtio block device
             |                            \-05.0  Red Hat, Inc Virtio memory balloon
             +-1f.0  Intel Corporation 82801IB (ICH9) LPC Interface Controller
             +-1f.2  Intel Corporation 82801IR/IO/IH (ICH9R/DO/DH) 6 port SATA Controller [AHCI mode]
             \-1f.3  Intel Corporation 82801I (ICH9 Family) SMBus Controller
```

The IOMMU support will not be enabled yet as the kernel defaults to leaving it off. To enable it, we must update the kernel command line parameters with grub.

```
# vi /etc/default/grub
....add "intel_iommu=on"...
# grub2-mkconfig > /etc/grub2.cfg
```

While intel-iommu device in QEMU can do interrupt remapping, there is no way enable that feature via libvirt at this time. So we need to set a hack for vfio

```
echo "options vfio_iommu_type1 allow_unsafe_interrupts=1" > \
  /etc/modprobe.d/vfio.conf
```

This is also a good time to install libvirt and KVM inside the guest

```
# dnf groupinstall "Virtualization"
# dnf install libvirt-client
# rm -f /etc/libvirt/qemu/networks/autostart/default.xml
```

Note we're disabling the default libvirt network, since it'll clash with the IP address range used by this guest. An alternative would be to edit the default.xml to change the IP subnet.

Now reboot the guest. When it comes back up, there should be a `/dev/kvm` device present in the guest.

```
# ls -al /dev/kvm
crw-rw-rw-. 1 root kvm 10, 232 Oct  4 12:14 /dev/kvm
```

If this is not the case, make sure the physical host has nested virtualization enabled for the "kvm-intel" or "kvm-amd" kernel modules.

The IOMMU should have been detected and activated

```
# dmesg  | grep -i DMAR
[    0.000000] ACPI: DMAR 0x000000007FFE2541 000048 (v01 BOCHS  BXPCDMAR 00000001 BXPC 00000001)
[    0.000000] DMAR: IOMMU enabled
[    0.203737] DMAR: Host address width 39
[    0.203739] DMAR: DRHD base: 0x000000fed90000 flags: 0x1
[    0.203776] DMAR: dmar0: reg_base_addr fed90000 ver 1:0 cap 12008c22260206 ecap f02
[    2.910862] DMAR: No RMRR found
[    2.910863] DMAR: No ATSR found
[    2.914870] DMAR: dmar0: Using Queued invalidation
[    2.914924] DMAR: Setting RMRR:
[    2.914926] DMAR: Prepare 0-16MiB unity mapping for LPC
[    2.915039] DMAR: Setting identity map for device 0000:00:1f.0 [0x0 - 0xffffff]
[    2.915140] DMAR: Intel(R) Virtualization Technology for Directed I/O
```

The key message confirming everything is good is the last line there – if that's missing something went wrong – don't be mislead by the earlier "DMAR: IOMMU enabled" line which merely says the kernel saw the "intel_iommu=on" command line option.

The IOMMU should also have registered the PCI devices into various groups

```
# dmesg  | grep -i iommu  |grep device
[    2.915212] iommu: Adding device 0000:00:00.0 to group 0
[    2.915226] iommu: Adding device 0000:00:01.0 to group 1
...snip...
[    5.588723] iommu: Adding device 0000:b5:00.0 to group 14
[    5.588737] iommu: Adding device 0000:b6:00.0 to group 15
[    5.588751] iommu: Adding device 0000:b7:00.0 to group 16
```

Libvirt meanwhile should have detected all the PCI controllers/devices

```
# virsh nodedev-list --tree
computer
  |
  +- net_lo_00_00_00_00_00_00
  +- pci_0000_00_00_0
  +- pci_0000_00_01_0
  +- pci_0000_00_02_0
  +- pci_0000_00_03_0
  +- pci_0000_00_04_0
  +- pci_0000_00_1d_0
  |   |
  |   +- usb_usb2
  |       |
  |       +- usb_2_0_1_0
  |
  +- pci_0000_00_1d_1
  |   |
  |   +- usb_usb3
  |       |
  |       +- usb_3_0_1_0
  |
  +- pci_0000_00_1d_2
  |   |
  |   +- usb_usb4
  |       |
  |       +- usb_4_0_1_0
  |
  +- pci_0000_00_1d_7
  |   |
  |   +- usb_usb1
  |       |
  |       +- usb_1_0_1_0
  |       +- usb_1_1
  |           |
  |           +- usb_1_1_1_0
  |
  +- pci_0000_00_1e_0
  |   |
  |   +- pci_0000_01_01_0
  |       |
  |       +- pci_0000_02_01_0
  |       |   |
  |       |   +- net_enp2s1_52_54_00_10_26_7e
  |       |
  |       +- pci_0000_02_02_0
  |       +- pci_0000_02_03_0
  |       +- pci_0000_02_04_0
  |       +- pci_0000_02_05_0
  |
  +- pci_0000_00_1f_0
  +- pci_0000_00_1f_2
  |   |
  |   +- scsi_host0
  |   +- scsi_host1
  |   +- scsi_host2
  |   +- scsi_host3
  |   +- scsi_host4
  |   +- scsi_host5
  |
  +- pci_0000_00_1f_3
  +- pci_0000_b4_00_0
  |   |
  |   +- pci_0000_b5_00_0
  |       |
  |       +- net_enp181s0_52_54_00_7e_6e_c6
  |
  +- pci_0000_b4_01_0
  |   |
  |   +- pci_0000_b6_00_0
  |       |
  |       +- net_enp182s0_52_54_00_7e_6e_c7
  |
  +- pci_0000_b4_02_0
  |   |
  |   +- pci_0000_b7_00_0
  |       |
  |       +- net_enp183s0_52_54_00_7e_6e_c8
  |
  +- pci_0000_c8_00_0
  |   |
  |   +- pci_0000_c9_00_0
  |       |
  |       +- net_enp201s0_52_54_00_7e_6e_d6
  |
  +- pci_0000_c8_01_0
  |   |
  |   +- pci_0000_ca_00_0
  |       |
  |       +- net_enp202s0_52_54_00_7e_6e_d7
  |
  +- pci_0000_c8_02_0
  |   |
  |   +- pci_0000_cb_00_0
  |       |
  |       +- net_enp203s0_52_54_00_7e_6e_d8
  |
  +- pci_0000_dc_00_0
  |   |
  |   +- pci_0000_dd_00_0
  |       |
  |       +- net_enp221s0_52_54_00_7e_6e_e6
  |
  +- pci_0000_dc_01_0
  |   |
  |   +- pci_0000_de_00_0
  |       |
  |       +- net_enp222s0_52_54_00_7e_6e_e7
  |
  +- pci_0000_dc_02_0
      |
      +- pci_0000_df_00_0
          |
          +- net_enp223s0_52_54_00_7e_6e_e8
```

And if you look at at specific PCI device, it should report the NUMA node it is associated with and the IOMMU group it is part of

```
# virsh nodedev-dumpxml pci_0000_df_00_0
<device>
  <name>pci_0000_df_00_0</name>
  <path>/sys/devices/pci0000:dc/0000:dc:02.0/0000:df:00.0</path>
  <parent>pci_0000_dc_02_0</parent>
  <driver>
    <name>e1000e</name>
  </driver>
  <capability type='pci'>
    <domain>0</domain>
    <bus>223</bus>
    <slot>0</slot>
    <function>0</function>
    <product id='0x10d3'>82574L Gigabit Network Connection</product>
    <vendor id='0x8086'>Intel Corporation</vendor>
    <iommuGroup number='10'>
      <address domain='0x0000' bus='0xdc' slot='0x02' function='0x0'/>
      <address domain='0x0000' bus='0xdf' slot='0x00' function='0x0'/>
    </iommuGroup>
    <numa node='2'/>
    <pci-express>
      <link validity='cap' port='0' speed='2.5' width='1'/>
      <link validity='sta' speed='2.5' width='1'/>
    </pci-express>
  </capability>
</device>
```

Finally, libvirt should also be reporting the NUMA topology

```
# virsh capabilities
...snip...
<topology>
  <cells num='3'>
    <cell id='0'>
      <memory unit='KiB'>4014464</memory>
      <pages unit='KiB' size='4'>1003616</pages>
      <pages unit='KiB' size='2048'>0</pages>
      <pages unit='KiB' size='1048576'>0</pages>
      <distances>
        <sibling id='0' value='10'/>
        <sibling id='1' value='20'/>
        <sibling id='2' value='20'/>
      </distances>
      <cpus num='4'>
        <cpu id='0' socket_id='0' core_id='0' siblings='0'/>
        <cpu id='1' socket_id='1' core_id='0' siblings='1'/>
        <cpu id='2' socket_id='2' core_id='0' siblings='2'/>
        <cpu id='3' socket_id='3' core_id='0' siblings='3'/>
      </cpus>
    </cell>
    <cell id='1'>
      <memory unit='KiB'>2016808</memory>
      <pages unit='KiB' size='4'>504202</pages>
      <pages unit='KiB' size='2048'>0</pages>
      <pages unit='KiB' size='1048576'>0</pages>
      <distances>
        <sibling id='0' value='20'/>
        <sibling id='1' value='10'/>
        <sibling id='2' value='20'/>
      </distances>
      <cpus num='2'>
        <cpu id='4' socket_id='4' core_id='0' siblings='4'/>
        <cpu id='5' socket_id='5' core_id='0' siblings='5'/>
      </cpus>
    </cell>
    <cell id='2'>
      <memory unit='KiB'>2014644</memory>
      <pages unit='KiB' size='4'>503661</pages>
      <pages unit='KiB' size='2048'>0</pages>
      <pages unit='KiB' size='1048576'>0</pages>
      <distances>
        <sibling id='0' value='20'/>
        <sibling id='1' value='20'/>
        <sibling id='2' value='10'/>
      </distances>
      <cpus num='2'>
        <cpu id='6' socket_id='6' core_id='0' siblings='6'/>
        <cpu id='7' socket_id='7' core_id='0' siblings='7'/>
      </cpus>
    </cell>
  </cells>
</topology>
...snip...
```

Everything should be ready and working at this point, so lets try and install a nested guest, and assign it one of the e1000e PCI devices. For simplicity we'll just do the exact same install for the nested guest, as we used for the top level guest we're currently running in. The only difference is that we'll assign it a PCI device

```
# cd /var/lib/libvirt/images
# wget -O f25x86_64-boot.iso https://download.fedoraproject.org/pub/fedora/linux/releases/25/Server/x86_64/os/images/boot.iso
# virt-install --name f25x86_64 --ram 2000 --vcpus 8 \
    --file /var/lib/libvirt/images/f25x86_64.img --file-size 10 \
    --cdrom f25x86_64-boot.iso --os-type fedora23 \
    --hostdev pci_0000_df_00_0 --network none
```

If everything went well, you should now have a nested guest with an assigned PCI device attached to it.

This turned out to be a rather long blog posting, but this is not surprising as we're experimenting with some cutting edge KVM features trying to emulate quite a complicated hardware setup, that deviates from normal KVM guest setup quite a way. Perhaps in the future virt-install will be able to simplify some of this, but at least for the short-medium term there'll be a fair bit of work required. The positive thing though is that this has clearly demonstrated that KVM is now advanced enough that you can now reasonably expect to do development and testing of features like NUMA and PCI device assignment inside nested guests.

The next step is to convince someone to add QEMU emulation of an Intel SRIOV network device....volunteers please :-)

by Daniel Berrange at February 16, 2017 12:44 PM (https://www.berrange.com/posts/2017/02/16/setting-up-a-nested-kvm-guest-for-developing-testing-pci-device-assignment-with-numa/)

ANNOUNCE: libosinfo 1.0.0 release (https://www.berrange.com/posts/2017/02/16/announce-libosinfo-1-0-0-release/)

NB, this blog post was intended to be published back in November last year, but got forgotten in draft stage. Publishing now in case anyone missed the release…

I am happy to announce a new release of libosinfo, version 1.0.0 (https://fedorahosted.org/releases/l/i/libosinfo/libosinfo-1.0.0.tar.gz) is now available, signed (https://fedorahosted.org/releases/l/i/libosinfo/libosinfo-1.0.0.tar.gz.asc) with key DAF3 A6FD B26B 6291 2D0E 8E3F BE86 EBB4 1510 4FDF (4096R). All historical releases are available from the project download page (http://libosinfo.org/download/).

Changes in this release include:

- Update loader to follow new layout for external database

- Move all database files into separate osinfo-db package
- Move osinfo-db-validate into osinfo-db-tools package

As promised, this release of libosinfo has completed the separation of the library code from the database files. There are now three independently released artefacts:

- libosinfo – provides the libosinfo shared library and most associated command line tools
- osinfo-db – contains only the database XML files and RNG schema, no code at all.
- osinfo-db-tools – a set of command line tools for managing deployment of osinfo-db archives for vendors & users.

Before installing the 1.0.0 release of libosinfo it is necessary to install osinfo-db-tools, followed by osinfo-db. The download (https://libosinfo.org/download/) page has instructions for how to deploy the three components. In particular note that 'osinfo-db' does **NOT** contain any traditional build system, as the only files it contains are XML database files. So instead of unpacking the osinfo-db archive, use the osinfo-db-import tool to deploy it.

by Daniel Berrange at February 16, 2017 11:19 AM (https://www.berrange.com/posts/2017/02/16/announce-libosinfo-1-0-0-release/)

Mirantis (https://www.mirantis.com)

Planning for OpenStack Summit Boston begins (https://www.mirantis.com/blog/planning-for-openstack-summit-boston-begins/)

The post Planning for OpenStack Summit Boston begins (https://www.mirantis.com/blog/planning-for-openstack-summit-boston-begins/) appeared first on Mirantis | Pure Play Open Cloud (https://www.mirantis.com).

The next OpenStack summit will be held in Boston May 8 through May 11, 2017, and the agenda is in progress. Mirantis folks, as well as some of our customers, have submitted talks, and we'd like to invite you to take a look, and perhaps to vote to show your support in this process. The talks include:

- From Point and Click to CI/CD: A real world look at accelerating OpenStack deployment, improving sustainability, and painless upgrades! (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17508) (Bruce Mathews, Ryan Day, Amit Tank (AT&T))
- Terraforming the OpenStack Landscape (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17752) (Mykyta Gubenko)
- Virtualized services delivery using SDN/NFV: from end-to-end in a brownfield MSO environment (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/18005) (Bill Coward (Cox Business Services))
- Operational automation of elements, api calls, integrations, and other pieces of MSO SDN/NFV cloud (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/18000) (Bill Coward (Cox Business Services))
- The final word on Availability Zones (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17659) (Craig Anderson)
- m1.Boaty.McBoatface: The joys of flavor planning by popular vote (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17658) (Craig Anderson)
- Proactive support and Customer care (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17649) (Anton Tarasov)
- OpenStack with SaltStack for complete deployment automation (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/18297) (Ales Komarek)
- Resilient RabbitMQ cluster automation with Kubernetes (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17706) (Alexey Lebedev, Michael Klishin (Pivotal))
- How fast is fast enough? The science behind bottlenecks (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17991) (Christian Huebner)
- Approaches for cloud transformation of Big Data use case (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17995) (Christian Huebner)
- Workload Onboarding and Lifecycle Management with Heat (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17804) (Florin Stingaciu)
- Preventing Nightmares: Data Protection for OpenStack environments (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17997) (Christian Huebner)
- Deploy a Distributed Containerized OpenStack Control Plane Infrastructure (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17642) (Rama Darbha (Cumulus), Stanley Karunditu)
- Saving one cloud at a time with tenant care (https://www.openstack.org/summit/boston-2017/vote-for-speakers/#/18310) (Bryan Langston, Holly Bazemore (Comcast), Shilla Saebi (Comcast))
- CI/CD in Documentation (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17888) (Alexandra Settle (Rackspace), Olga Gusarenko)
- Kuryr-Kubernetes: The seamless path to adding Pods to your datacenter networking (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17700) (Antoni Segura Puimedon (RedHat), Irena Berezovsky (Huawei), Ilya Chukhnakov)
- Cinder Stands Alone (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17740) (Scott DAngelo (IBM), Ivan Kolodyazhny, Walter A. Boring IV (IBM))
- NVMe-over-Fabrics and Openstack (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/18114) (Tushar Gohad (Intel), Michał Dulko (Intel), Ivan Kolodyazhny)
- Episode 2: Log Book: VW Ops team's adventurous journey to the land of OpenStack – Go Global (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/18261) (Gerd Pruessmann, Tilman Schulz (Volkswagen))
- OpenStack: pushing to 5000 nodes and beyond (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/18414) (Dina Belova, Georgy Okrokvertskhov)
- Turbo Charged VNFs at 40 gbit/s. Approaches to deliver fast, low latency networking using OpenStack (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/18294) (Greg Elkinbard)
- Using Top of the Rack Switch as a fast L2 and L3 Gateway on OpenStack (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/18329) (Greg Elkinbard)
- Deploy a Distributed Containerized OpenStack Control Plane Infrastructure (https://www.openstack.org/summit/boston-2017/vote-for-speakers#/17642) (Stanley Karunditu)

While you're in Boston, consider taking a little extra time in Beantown to take advantage of Mirantis Training's special Pre-Summit training (https://training.mirantis.com/class/openstack-courses/openstack-accelerated-bootcamp-with-exam-os250/boston-ma-regus-3/2017-05-01), which includes a bonus introduction module on the Mirantis Cloud Platform (MCP). You'll get to the summit up to speed with the technology, and even (if you pass the exam) the OCM100 OpenStack certification (https://training.mirantis.com/instructor-led-training/openstack-exam-ocm100). Can't make it to Boston? You can also take the pre-summit class live from the comfort of your own home (or office) (https://training.mirantis.com/class/openstack-courses/openstack-accelerated-bootcamp-with-exam-os250/virtual---americas/2017-05-01).

The post Planning for OpenStack Summit Boston begins (https://www.mirantis.com/blog/planning-for-openstack-summit-boston-begins/) appeared first on Mirantis | Pure Play Open Cloud (https://www.mirantis.com).

by Nick Chase at February 16, 2017 03:25 AM (https://www.mirantis.com/blog/planning-for-openstack-summit-boston-begins/)

February 15, 2017

OpenStack Superuser (http://superuser.openstack.org)

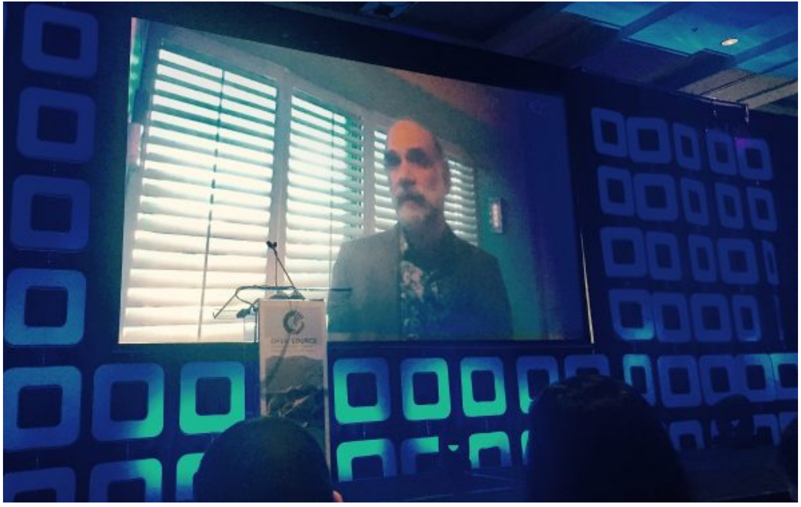
Security expert: open source must embrace working with the government or else (http://superuser.openstack.org/articles/bruce-schneier-open-source-policy/)

LAKE TAHOE, CALIF. — At a time when tech companies are locked in an awkward dance with the government, one security expert says that the open-source community must embrace working with lawmakers or face death by regulation.

"We've had this special right to code the world as we see fit," says security guru Bruce Schneier (https://www.schneier.com/), speaking at the Linux Foundation's Open Source leadership summit. (http://events.linuxfoundation.org/events/open-source-leadership-summit) "My guess is that we're going to lose this right because it's too dangerous to give it to a bunch of techies."

Up until now, he noted that the industry has left security to the market with decent results, but the tune has changed with the internet of things (IoT). Your connected car, toaster and thermostat and medical devices are turning the world into what amounts to a robot, says Schneier who appeared via Skype from a darkened room while attending the RSA (https://www.rsaconference.com/), making his predictions about the future even more ominous. This "world robot," is not the Terminator-type sci-fi fans expect, but rather one without either a single goal or one creator.

<figure class="wp-caption alignnone" id="attachment_5449" style="width: 605px">



(http://superuser.openstack.org/wp-content/uploads/2017/02/bruce.jpg)<figcaption class="wp-caption-text"> Bruce Schneier speaking via Skype at the Open Source leadership summit. // </figcaption> </figure>

"As everything becomes a computer, computer security becomes everything security," he says. With iOT, the traditional paradigms of security are out of synch, sometimes with disastrous results. The paradigm where things are done right and properly first time (buildings, cars, medical devices) and the other (software) where the goal is to be agile and developers can always add patches and updates as vulnerabilities arise. "These two worlds are colliding (literally) now in things like automobiles, medical devices and e-voting."

> RT linuxfoundation: Schneier: We'll never get policy right if policymakers get the technology wrong. #lfosls (https://twitter.com/hashtag/lfosls?src=hash)
>
> — Adil Mishra (@AdilMishra1) February 14, 2017 (https://twitter.com/AdilMishra1/status/831562782193618948)

<script async="async" charset="utf-8" src="http://platform.twitter.com/widgets.js"></script>

Your computer and phone are secure because there are teams of engineers at companies like Apple and Google working to make them secure, he said, holding up his own iPhone. With "smart" devices, there are often external teams who build libraries on the fly and then disband. You also replace your phone every two years which ensures updated security, but replace your car every 10 years, your refrigerator 25 years and your thermostat, well, never.

The effect is colossal: there is a fundamental difference between what happens when a spreadsheet crashes and a car or pacemaker crashes. From the standpoint of security professionals "it's the same thing, for the rest of the world it's not."

> #lfosls (https://twitter.com/hashtag/lfosls?src=hash) Bruce Schneier – 5.5m new devices connect to the Internet every day, most with poorly written, insecure, non-updatable software
>
> — Rod Cope (@RodCope) February 14, 2017 (https://twitter.com/RodCope/status/831561192619380737)

<script async="async" charset="utf-8" src="http://platform.twitter.com/widgets.js"></script>

That's where he expects the government to come in. He predicts that the first line of intervention will be through the courts — most likely liabilities and tort law — with congress following.

"Nothing motivates the U.S. government like fear," he says. So the open-source community must connect with lawmakers because there's "smart government involvement and stupid government involvement. You can imagine a liability regime that would kill open source."

His talk was in step with the earlier keynote by Jim Zemlin, the Linux Foundation's executive director, who said that the cyber security should be at the forefront of everyone's agenda.

> Bruce Schneier: "We have prioritized features, speed, and price over security." Oops! #lfosls (https://twitter.com/hashtag/lfosls?src=hash)
>
> — Yev the dev (@YevTheDev) February 14, 2017 (https://twitter.com/YevTheDev/status/831559577623678976)

<script async="async" charset="utf-8" src="http://platform.twitter.com/widgets.js"></script>

Schneier made a plea for the open-source community to get involved with policy before it's too late. He pitched the idea of an iOT security regulatory agency in the hopes of getting new expertise and control over the ever-shifting tech landscape.

"We build tech because it's cool. We don't design our future, we just see what happens. We need to make moral and ethical decisions about how we want to work."

"This is a horribly contentious idea but my worry is that the alternatives aren't viable any longer," he said.

Cover photo: Chris Isherwood
(https://www.flickr.com/photos/isherwoodchris/6774558732/)

The post Security expert: open source must embrace working with the government or else (http://superuser.openstack.org/articles/bruce-schneier-open-source-policy/) appeared first on OpenStack Superuser (http://superuser.openstack.org).

by Nicole Martinelli at February 15, 2017 01:19 PM (http://superuser.openstack.org/articles/bruce-schneier-open-source-policy/)

February 14, 2017

The Official Rackspace Blog (https://blog.rackspace.com)

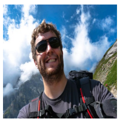
What is OpenStack? The Basics, Part 1 (https://blog.rackspace.com/what-is-openstack-the-basics-part-1)

OpenStack. In an increasingly cloud-obsessed world, you've probably heard of it. Maybe you've read it's "one of the fastest growing open source communities in the world," but you're still not sure what all the hype is about. The aim of this post is to get you from zero to 60 on the basics of OpenStack,

The post What is OpenStack? The Basics, Part 1 (https://blog.rackspace.com/what-is-openstack-the-basics-part-1) appeared first on The Official Rackspace Blog (https://blog.rackspace.com).

by Walter Bentley at February 14, 2017 06:58 PM (https://blog.rackspace.com/what-is-openstack-the-basics-part-1)

Dougal Matthews (http://www.dougalmatthews.com/)



Mistral on-success, on-error and on-complete (http://www.dougalmatthews.com/2017/Feb/14/mistral-on-success-on-error-and-on-complete/)

I spent a bit of time today looking into the subtleties of the Mistral task properties `on-success`, `on-complete` and `on-error` when used with the `fail` engine commands.

As an upcoming docs patch (https://review.openstack.org/#/c/433557/) explains, these are similar to the Python try, except, finally blocks. Meaning, that it would look like the following.

```
try:
    action()
    # on-success
except:
    # on-error
finally:
    # on-complete
```

I was looking to see how the Mistral engine command would work in combination with these. In TripleO (https://github.com/openstack/tripleo-common/blob/master/workbooks/baremetal.yaml) we want to mark a workflow as failed if it sends a Zaqar message with the value `{"status": "FAILED"}`. So our task would look a bit like this...

```
      send_message:
        action: zaqar.queue_post
        input:
          queue_name: <% $.queue_name %>
          messages:
            body:
              status: <% $.get('status', 'SUCCESS') %>
        on-complete:
          - fail: <% $.get('status') = "FAILED" %>
```

This task uses the `zaqar.queue_post` action to send a message containing the status. Once it is complete it will fail the workflow if the status is equal to `"FAILED"`. Then in the `mistral execution-list` the workflow will show as failed. This is good, because we want to surface the best error in the execution list.

However, if the `zaqar.queue_post` action fails then we want to surface that error instead. At the moment it will still be possible to see it in the list of action executions. However, looking at the workflow executions it isn't obvious where the problem was.

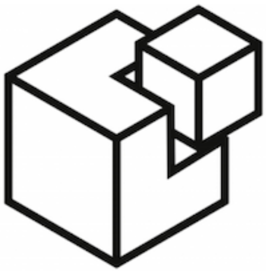
Changing the above example to on-success solves that. We only want to manually mark the workflow as having failed if the Zaqar message was sent with the FAILED status. Otherwise if the message fails to send, the workflow will error anyway with a more detailed error.

by Dougal Matthews at February 14, 2017 04:35 PM (http://www.dougalmatthews.com/2017/Feb/14/mistral-on-success-on-error-and-on-complete/)

Mirantis (https://www.mirantis.com)

Introduction to Salt and SaltStack (https://www.mirantis.com/blog/introduction-to-salt-and-saltstack/)

The post Introduction to Salt and SaltStack (https://www.mirantis.com/blog/introduction-to-salt-and-saltstack/) appeared first on Mirantis | Pure Play Open Cloud (https://www.mirantis.com).



(https://cdn.mirantis.com/wp-content/uploads/2017/02/image01.png)The amazing world of configuration management software is really well populated these days. You may already have looked at *Puppet* (https://puppet.com/), *Chef* (https://www.chef.io/) or *Ansible* (https://www.ansible.com/) but today we focus on *SaltStack* (https://saltstack.com/). Simplicity is at its core, without any compromise on speed or scalability. In fact, some users have up to 10,000 minions or more. In this article, we're going to give you a look at what Salt is and how it works.

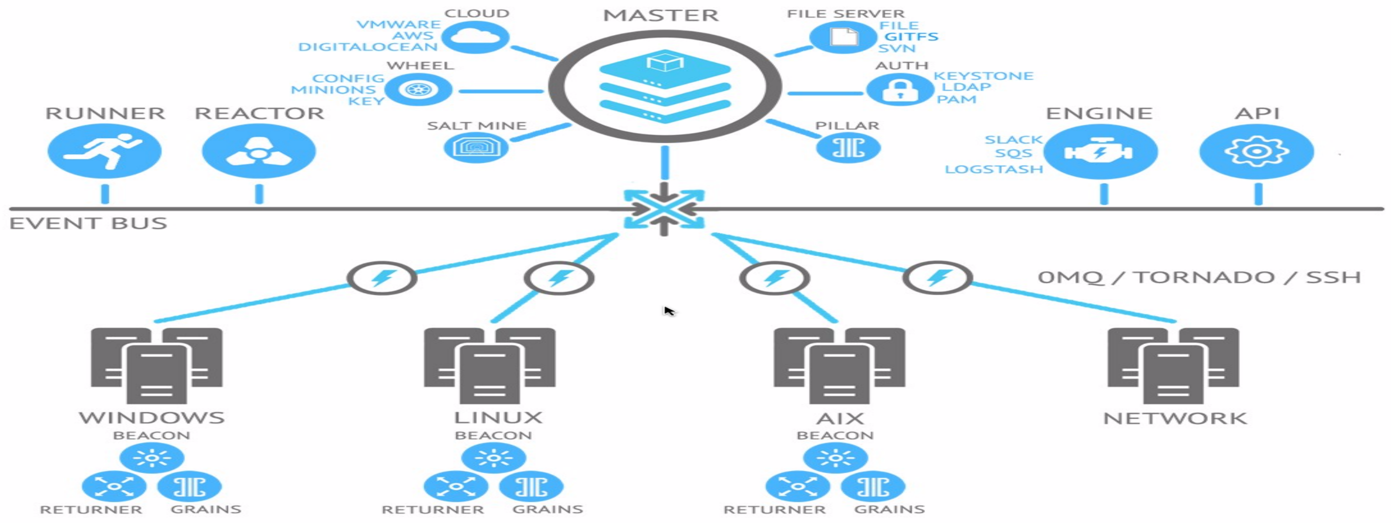
Salt architecture

Salt remote execution is built on top of an event bus, which makes it unique. It uses a server-agent communication model where the server is called the salt master and the agents the salt minions.

Salt minions receive commands simultaneously from the master and contain everything required to execute commands locally and report back to salt master. Communication between master and minions happens over a high-performance data pipe that use ZeroMQ or raw TCP, and messages are serialized using MessagePack to enable fast and light network traffic. Salt uses public keys for authentication with the master daemon, then uses faster AES encryption for payload communication.

State description is done using YAML and remote execution is possible over a CLI, and programming or extending Salt isn't a must.

Salt is heavily pluggable; each function can be replaced by a plugin implemented as a Python module. For example, you can replace the data store, the file server, authentication mechanism, even the state representation. So when I said state representation is done using YAML, I'm talking about the Salt default, which can be replaced by JSON, Jinja, Wempy, Mako, or Py Objects. But don't freak out. Salt comes with default options for all these things, which enables you to jumpstart the system and customize it when the need arises.



(https://cdn.mirantis.com/wp-content/uploads/2017/02/image00.png)Terminology

It's easy to be overwhelmed by the obscure vocabulary that Salt introduces, so here are the main salt concepts which make it unique.

- *salt master* – sends commands to minions
- *salt minions* – receives commands from master
- *execution modules* (https://docs.saltstack.com/en/latest/topics/execution/index.html) – ad hoc commands
- *grains* (https://docs.saltstack.com/en/latest/topics/grains/) – static information about minions
- *pillar* (https://docs.saltstack.com/en/latest/topics/pillar/) – secure user-defined variables stored on master and assigned to minions (equivalent to data bags in Chef or Hiera in Puppet)
- *formulas* (https://docs.saltstack.com/en/latest/topics/development/conventions/formulas.html) (states) – representation of a system configuration, a grouping of one or more state files, possibly with pillar data and configuration files or anything else which defines a neat package for a particular application.
- *mine* (https://docs.saltstack.com/en/latest/topics/mine/) – area on the master where results from minion executed commands can be stored, such as the IP address of a backend webserver, which can then be used to configure a load balancer

- *top file* (https://docs.saltstack.com/en/latest/ref/states/top.html) – matches formulas and pillar data to minions
- *runners* (https://docs.saltstack.com/en/latest/ref/runners/) – modules executed on the master
- *returners* (https://docs.saltstack.com/en/latest/ref/returners/) – components that inject minion data to another system
- *renderers* (https://docs.saltstack.com/en/latest/ref/renderers/) – components that run the template to produce the valid state of configuration files. The default renderer uses Jinja2 syntax and outputs YAML files.
- *reactor* (https://docs.saltstack.com/en/latest/topics/reactor/) – component that triggers reactions on events
- *thorium* (https://docs.saltstack.com/en/latest/topics/thorium/index.html) – a new kind of reactor, which is still experimental.
- *beacons* (https://docs.saltstack.com/en/latest/topics/beacons/) – a little piece of code on the minion that listens for events such as server failure or file changes. When it registers on of these events, it informs the master. Reactors are often used to do self healing.
- *proxy minions* (https://docs.saltstack.com/en/latest/topics/proxyminion/index.html) – components that translate Salt Language to device specific instructions in order to bring the device to the desired state using its API, or over SSH.
- *salt cloud* (https://docs.saltstack.com/en/latest/topics/cloud/) – command to bootstrap cloud nodes
- *salt ssh* (https://docs.saltstack.com/en/latest/topics/ssh/) – command to run commands on systems without minions

You'll find a great overview of all of this on the official docs (https://docs.saltstack.com/en/getstarted/overview.html).

Installation

Salt is built on top of lots of Python modules. Msgpack (https://pypi.python.org/pypi/msgpack-python), YAML (http://pyyaml.org/wiki/PyYAMLDocumentation), Jinja2 (https://pypi.python.org/pypi/Jinja2/), MarkupSafe (https://pypi.python.org/pypi/MarkupSafe/), ZeroMQ, Tornado (https://pypi.python.org/pypi/tornado/4.4.1), PyCrypto (https://pypi.python.org/pypi/pycrypto/) and M2Crypto (https://pypi.python.org/pypi/M2Crypto/0.25.1) are all required. To keep your system clean, easily upgradable and to avoid conflicts, the easiest installation workflow is to use system packages.

Salt is operating system specific; in the examples in this article, I'll be using Ubuntu 16.04 [Xenial Xerus]; for other Operating Systems consult the salt repo page (https://repo.saltstack.com/). For simplicity's sake, you can install the master and the minion on a single machine, and that's what we'll be doing here. Later, we'll talk about how you can add additional minions.

1. To install the master and the minion, execute the following commands:

```
$ sudo su
# apt-get update
# apt-get upgrade
# apt-get install curl wget
# echo "deb [arch=amd64] http://apt.tcpcloud.eu/nightly xenial tcp-salt" > /etc/apt/sources.list
# wget -O - http://apt.tcpcloud.eu/public.gpg | sudo apt-key add -
# apt-get clean
# apt-get update
# apt-get install -y salt-master salt-minion reclass
```

2. Finally, create the directory where you'll store your state files.

```
# mkdir -p /srv/salt
```

3. You should now have Salt installed on your system, so check to see if everything looks good:

```
# salt --version
```

You should see a result something like this:

```
salt 2016.3.4 (Boron)
```

Alternative installations

If you can't find packages for your distribution, you can rely on Salt Bootstrap (https://github.com/saltstack/salt-bootstrap), which is an alternative installation method, look below for further details.

Configuration

To finish your configuration, you'll need to execute a few more steps:

1. If you have firewalls in the way, make sure you open up both port 4505 (the publish port) and 4506 (the return port) to the Salt master to let the minions talk to it.
2. Now you need to configure your Minion to connect to your master. Edit the file /etc/salt/minion.d/minion.conf and Change the following lines as indicated below:

```
...

# Set the location of the salt master server. If the master server cannot be
# resolved, then the minion will fail to start.
master: localhost

# If multiple masters are specified in the 'master' setting, the default behavior
# is to always try to connect to them in the order they are listed. If random_master is
# set to True, the order will be randomized instead. This can be helpful in distributing

...

# Explicitly declare the id for this minion to use, if left commented the id
# will be the hostname as returned by the python call: socket.getfqdn()
# Since salt uses detached ids it is possible to run multiple minions on the
# same machine but with different ids, this can be useful for salt compute
# clusters.
id: saltstack-m01

# Append a domain to a hostname in the event that it does not exist.  This is
# useful for systems where socket.getfqdn() does not actually result in a
# FQDN (for instance, Solaris).
#append_domain:
...
```

As you can see, we're telling the minion where to find the master so it can connect — in this case, it's just localhost, but if that's not the case for you, you'll want to change it. We've also given this particular minion an id of saltstack-m01; that's a completely arbitrary name, so you can use whatever you want. Just make sure to substitute in the examples!

3. Before being able you can play around, you'll need to restart the required Salt services to pick up the changes:

```
# service salt-minion restart
# service salt-master restart
```

4. Make sure services are also started at boot time:

```
# systemctl enable salt-master.service
# systemctl enable salt-minion.service
```

5. Before the master can do anything on the minion, the master needs to trust it, so accept the corresponding key of each of your minion as follows:

```
# salt-key
 Accepted Keys:
 Denied Keys:
 Unaccepted Keys:
 saltstack-m01
 Rejected Keys:
```

6. Before accepting it, you can validate it looks good. First inspect it:

```
# salt-key -f saltstack-m01
 Unaccepted Keys:
 saltstack-m01:  98:f2:e1:9f:b2:b6:0e:fe:cb:70:cd:96:b0:37:51:d0
```

7. Then compare it with the minion key:

```
# salt-call --local key.finger
 local:
 98:f2:e1:9f:b2:b6:0e:fe:cb:70:cd:96:b0:37:51:d0
```

8. It looks the same, so go ahead and accept it:/span>

```
salt-key -a saltstack-m01
```

Repeat this process of installing salt-minion and accepting the keys to add new minions to your environment. Consult the documentation (https://docs.saltstack.com/en/latest/ref/configuration/minion.html) to get more details regarding the configuration of minions or more generally this documentation (https://docs.saltstack.com/en/latest/topics/configuration/index.html) for all salt configuration options.

Remote execution

Now that everything's installed and configured, let's make sure it's actually working. The first, most obvious thing we could do with our master/minion infrastructure is to run a command remotely. For example we can test whether the minion is alive by using the test.ping command:

```
# salt 'saltstack-m01' test.ping
 saltstack-m01:
     True
```

As you can see here, we're calling salt, and we're feeding it a specific minion, and a command to run on that minion. We could, if we wanted to, send this command to more than one minion. For example, we could send it to **all** minions:

```
# salt '*' test.ping
 saltstack-m01:
     True
```

In this case, we have only one, but if there were more, salt would cycle through all of them giving you the appropriate response.

So that should get you started. Next time, we'll look at some of the more complicated things you can do with Salt.

The post Introduction to Salt and SaltStack (https://www.mirantis.com/blog/introduction-to-salt-and-saltstack/) appeared first on Mirantis | Pure Play Open Cloud (https://www.mirantis.com).

by Sebastian Braun at February 14, 2017 01:25 PM (https://www.mirantis.com/blog/introduction-to-salt-and-saltstack/)

OpenStack Superuser (http://superuser.openstack.org)

Getting started with Kolla (http://superuser.openstack.org/articles/getting-started-with-kolla/)

I've been playing with Kolla for about a week, so I thought it'd be good to share my notes with the OpenStack operator community. (Kolla (https://wiki.openstack.org/wiki/Kolla) provides production-ready containers and deployment tools for operating OpenStack clouds, including Docker and Ansible.)

Up to `stable/newton` , Kolla was a single project that lives in the git repository:

- https://github.com/openstack/kolla (https://github.com/openstack/kolla)

In the current master (Ocata not yet released), Kolla is split into two repositories:

- https://github.com/openstack/kolla (https://github.com/openstack/kolla)
- https://github.com/openstack/kolla-ansible (https://github.com/openstack/kolla-ansible)

So in the current master, you won't find the directory with the Ansible roles, because that directory is now in the new repository.

There is also a `kolla-kubernetes` repo, but I haven't had the chance to look at that yet. I'll work up a second part to this tutorial about that soon.

My first goal was to deploy OpenStack on top of OpenStack with Kolla. I will use SWITCHengines (https://www.switch.ch/engines/) that is OpenStack Mitaka and I'll try to deploy OpenStack Newton.

To get started, you need an Operator Seed node, the machine where you actually install Kolla, and from where you can run the `kolla-ansible` command.

I used Ubuntu Xenial for all my testing. Ubuntu does not yet have packages for Kolla. Instead of just installing with `pip` a lot of Python stuff and coming up with a deployment that is hard to reproduce, on `#ubuntu-server` I got this tip to use https://snapcraft.io (https://snapcraft.io).

There are already some OpenStack tools packaged with snapcraft:

- https://github.com/openstack-snaps (https://github.com/openstack-snaps)

I looked at what was already done, then I tried to package a snap for Kolla myself:

- https://github.com/zioproto/snap-kolla (https://github.com/zioproto/snap-kolla)

It worked quite fast, but I needed to write a couple of Kolla patches:

- https://review.openstack.org/#/c/426703/ (https://review.openstack.org/#/c/426703/)
- https://review.openstack.org/#/c/427139/ (https://review.openstack.org/#/c/427139/)

Also, because I had a lot of permission issues, I had to introduce this ugly patch to run all Ansible things as `sudo` :

- https://github.com/zioproto/kolla/commit/51cc9ce1cc9d1c17410ef813930a65ac8b3dce4b (https://github.com/zioproto/kolla/commit/51cc9ce1cc9d1c17410ef813930a65ac8b3dce4b)

In the beginning, I tried to fix it in a elegant way and add only where necessary the `become: true` , but my work collided with some one who was already working on that:

- https://blueprints.launchpad.net/kolla-ansible/+spec/ansible-specific-task-become (https://blueprints.launchpad.net/kolla-ansible/+spec/ansible-specific-task-become)

I hope that all these dirty workarounds will be gone by `stable/ocata` . Apart from these small glitches everything worked pretty well.

For Docker, I used this repo on Xenial: `deb http://apt.dockerproject.org/repo ubuntu-xenial main`

Understanding high availability

Kolla comes with HA built in. The key idea is to have as a front end two servers sharing a public VIP with VRRP protocol. These front-end nodes run HAProxy in active-backup mode. HAProxy then load-balances the requests for the API services and for DB and RabbitMQ to two or more controller nodes in the back end.

In the standard setup the front-end nodes are called `network` because they act also as Neutron network nodes. The nodes in the back end are called `controllers` .

(https://github.com/zioproto/kolla-on-SWITCHengines/#run-the-playbook)Run the playbook

To get started, source your OpenStack config and get a tenant with enough quota and run this Ansible playbook:

```
cp vars.yaml.template vars.yaml
vim vars.yaml # add your custom config
export ANSIBLE_HOST_KEY_CHECKING=False
source ~/opestack-config
ansible-playbook main.yaml
```

The Ansible playbook will create the necessary VMs, will hack the `/etc/hosts` of all VMs so that they look all reachable to each other with names, and it will install Kolla on the operator-seed node using my snap package.

To have the frontend VMs share a VIP, I used the approach I found on this blog:

- https://blog.codecentric.de/en/2016/11/highly-available-vips-openstack-vms-vrrp/ (https://blog.codecentric.de/en/2016/11/highly-available-vips-openstack-vms-vrrp/)

The playbook will configure all the OpenStack networking needed for our tests, and will configure Kolla on the operator node.

Now you can ssh to the operator node and start configuring Kolla. For this easy example, make sure that on `/etc/kolla/passwords.yaml` you have at least something written for the following values:

```
database_password:
rabbitmq_password:
rabbitmq_cluster_cookie:
haproxy_password:
```

If you want, you can also just type `kolla-genpwd` and this will enter passwords for all the fields in the file.

Now let's get ready to run Ansible:

```
export ANSIBLE_HOST_KEY_CHECKING=False
kolla-ansible -i inventory/mariadb bootstrap-servers
kolla-ansible -i inventory/mariadb pull
kolla-ansible -i inventory/mariadb deploy
```

This example inventory that I have put at the path `/home/ubuntu/inventory/mariadb` is a very simplified inventory that will just deploy `mariadb` and `rabbitmq`. Check what I disabled in `/etc/kolla/globals.yml`

(https://github.com/zioproto/kolla-on-SWITCHengines/#check-what-is-working)Check what is working

With the command:

```
openstack floating ip list | grep 192.168.22.22
```

You can check the public floating IP applied to the VIP. Check the OpenStack security groups applied to the front-end VMs. If the necessary ports are open you should be able to access the MySQL service on port 3306, and the HAProxy admin panel on port 1984. The passwords are the ones in the `password.yml` file and the username for HAProxy is `openstack`.

(https://github.com/zioproto/kolla-on-SWITCHengines/#todo)TODO

I will update this file (https://github.com/zioproto/kolla-on-SWITCHengines/) with more steps 🙂 Pull requests are welcome!

Saverio Proto (https://github.com/zioproto)is a cloud engineer at SWITCH, (http://cloudblog.switch.ch/) a national research and education network in Switzerland, which runs a public cloud for national universities.

Superuser is always interested in community content, email: editor@superuser.org.

The post Getting started with Kolla (http://superuser.openstack.org/articles/getting-started-with-kolla/) appeared first on OpenStack Superuser (http://superuser.openstack.org).

by Saverio Proto at February 14, 2017 12:52 PM (http://superuser.openstack.org/articles/getting-started-with-kolla/)

February 13, 2017

David Moreau Simard (https://dmsimard.com/)

Announcing the ARA 0.11 release (https://dmsimard.com/2017/02/13/announcing-the-ara-0.11-release/)

We're on the road to version 1.0.0 and we're getting closer: introducing the release of version 0.11!

Four new contributors (https://github.com/openstack/ara/graphs/contributors) (!), 55 commits (https://github.com/openstack/ara/compare/0.10.0...0.11.0) since 0.10 and 112 files changed for a total of 2,247 additions and 939 deletions.

New features, more stability, better documentation and better test coverage.

The changelog since 0.10.5

- New feature: ARA UI and Ansible version (ARA UI is running with) are now shown at the top right
- New feature: The Ansible version a playbook was run is now stored and displayed in the playbook reports
- New feature: New command: "ara generate junit": generates a junit xml stream of all task results
- New feature: ara_record now supports two new types: "list" and "dict", each rendered appropriately in the UI
- UI: Add ARA logo and favicon
- UI: Left navigation bar was removed (top navigation bar will be further improved in future versions)
- Bugfix: CLI commands could sometimes fail when trying to format as JSON or YAML
- Bugfix: Database and logs now properly default to ARA_DIR if ARA_DIR is changed
- Bugfix: When using non-ascii characters (ex: äëö) in playbook files, web application or static generation could fail
- Bugfix: Trying to use ara_record to record non strings (ex: lists or dicts) could fail
- Bugfix: Ansible config: 'tmppath' is now a 'type_value' instead of a boolean

- Deprecation: The “ara generate” command was deprecated and moved to “ara generate html”
- Deprecation: The deprecated callback location, ara/callback has been removed. Use ara/plugins/callbacks.
- Misc: Various unit and integration testing coverage improvements and optimization
- Misc: Slowly started working on full python 3 compatibility
- Misc: ARA now has a logo

ARA now has a logo !

Thanks Jason Rist (https://twitter.com/knowncitizen) for the contribution, really appreciate it !

With the icon (https://github.com/openstack/ara/blob/master/doc/source/_static/ara-with-icon.png):

Without the icon (https://github.com/openstack/ara/blob/master/doc/source/_static/ara.png):

Taking the newest version of ARA out for a spin

Want to give this new version a try ? It’s out on pypi!

Install dependencies and ARA (http://ara.readthedocs.io/en/latest/installation.html), configure the Ansible callback location (http://ara.readthedocs.io/en/latest/configuration.html#ansible) and ansible-playbook your stuff !

Once ARA has recorded your playbook, you’ll be able to fire off and browse the embedded server (http://ara.readthedocs.io/en/latest/usage.html#browsing-the-web-interface) or generate a static version (http://ara.readthedocs.io/en/latest/usage.html#generating-a-static-html-version-of-the-web-application) of the report.

The road ahead: version 1.0

What is coming in version 1.0 ? Let me ask you this question: what would *you* like in 1.0 ? The development of ARA has mostly been driven by it’s user’s needs and I’m really excited with what we already have.

I’d like to finish a few things before releasing 1.0… let’s take a sneak peek.

New web user interface

I’ve been working slowly but surely on a complete UI refactor, you can look at an early prototype preview here (https://www.youtube.com/watch?v=h3vY87_EWHw).

<iframe allowfullscreen="allowfullscreen" frameborder="0" height="480" src="https://www.youtube.com/embed/h3vY87_EWHw" width="853"></iframe> Some ideas and concepts have evolved since then but the general idea is to try and display more information in less pages, while not going overboard and have your browser throw up due to the weight of the pages.

Some ARA users are running playbooks involving hundreds of hosts or thousands of tasks and it makes the static generation very slow, large and heavy. While I don’t think I’ll be able to make the static generation work well at any kind of scale, I think we can make this better.

There will have to be a certain point in terms of scale where users will be encouraged to leverage the dynamic web application instead.

Python 3 support

ARA isn't gating against python3 right now and is actually failing unit tests when running python3.

As Ansible is working towards python3 support, ARA needs to be there too.

More complex use case support (stability/maturity)

There are some cases where it's unclear if ARA works well or works at all. This is probably a matter of stability and maturity.

For example, ARA currently might not behave well when running concurrent ansible-playbook runs from the same node or if a remote database server happens to be on vacation.

More complex use case support might also mean providing users documentation on how to best leverage all the data that ARA records and provides: elasticsearch implementation, junit reports and so on.

If ARA is useful to you, I'd be happy to learn about your use case. Get in touch and let's chat.

Implement support for ad-hoc ansible run logging

ARA will by default record anything and everything related to ansible-playbook runs. It also needs to support ad-hoc ansible commands as well. I want this before tagging 1.0.

Other features

There's some other features I'd like to see make the cut for version 1.0:

- Fully featured Ansible role (https://github.com/openstack/ansible-role-ara) for ARA
- Store variables and extra variables
- Provide some level of support for data on a role basis (filter tasks by role, metrics, duration, etc.)
- Support generating a html or junit report for a specific playbook (rather than the whole thing)
- Packaging for Debian/Ubuntu and Fedora/CentOS/RHEL

A stretch goal would be to re-write ARA to be properly split between client, server, UI and API — however I'm okay to let that slip for 2.0!

What else would you like to see in ARA ? Let me know in the comments, on IRC in #ara on freenode or on twitter (https://twitter.com/dmsimard)!

by dmsimard at February 13, 2017 04:00 PM (https://dmsimard.com/2017/02/13/announcing-the-ara-0.11-release/)

RDO (http://rdoproject.org/blog/)

RDO
An OpenStack Distribution

RDO blogs, week of Feb 13 (http://rdoproject.org/blog/2017/02/rdo-blogs-week-of-feb-13/)

Here's what RDO enthusiasts have been blogging about in the last few weeks. If you blog about RDO, please let me know (rbowen@redhat.com) so I can add you to my list.

TripleO: Debugging Overcloud Deployment Failure by bregman

> You run 'openstack overcloud deploy' and after a couple of minutes you find out it failed and if that's not enough, then you open the deployment log just to find a very (very!) long output that doesn't give you an clue as to why the deployment failed. In the following sections we'll see how can […]

Read more at http://tm3.org/dv (http://tm3.org/dv)

RDO @ DevConf by Rich Bowen

> It's been a very busy few weeks in the RDO travel schedule, and we wanted to share some photos with you from RDO's booth at DevConf.cz.

Read more at http://tm3.org/dw (http://tm3.org/dw)

The surprisingly complicated world of disk image sizes by Daniel Berrange

> When managing virtual machines one of the key tasks is to understand the utilization of resources being consumed, whether RAM, CPU, network or storage. This post will examine different aspects of managing storage when using file based disk images, as opposed to block storage. When provisioning a virtual machine the tenant user will have an idea of the amount of storage they wish the guest operating system to see for their virtual disks. This is the easy part. It is simply a matter of telling 'qemu-img' (or a similar tool) '40GB' and it will create a virtual disk image that is visible to the guest OS as a 40GB volume. The virtualization host administrator, however, doesn't particularly care about what size the guest OS sees. They are instead interested in how much space is (or will be) consumed in the host filesystem storing the image. With this in mind, there are four key figures to consider when managing storage:

Read more at http://tm3.org/dx (http://tm3.org/dx)

Project Leader by rbowen

> I was recently asked to write something about the project that I work on – RDO – and one of the questions that was asked was:

Read more at http://tm3.org/dy (http://tm3.org/dy)

os_type property for Windows images on KVM by Tim Bell

> The OpenStack images have a long list of properties which can set to describe the image meta data. The full list is described in the documentation. This blog reviews some of these settings for Windows guests running on KVM, in particular for Windows 7 and Windows 2008R2.

Read more at http://tm3.org/dz (http://tm3.org/dz)

Commenting out XML snippets in libvirt guest config by stashing it as metadata by Daniel Berrange

> Libvirt uses XML as the format for configuring objects it manages, including virtual machines. Sometimes when debugging / developing it is desirable to comment out sections of the virtual machine configuration to test some idea. For example, one might want to temporarily remove a secondary disk. It is not always desirable to just delete the configuration entirely, as it may need to be re-added immediately after. XML has support for comments which one might try to use to achieve this. Using comments in XML fed into libvirt, however, will result in an unwelcome suprise – the commented out text is thrown into /dev/null by libvirt.

Read more at http://tm3.org/d- (http://tm3.org/d-)

Videos from the CentOS Dojo, Brussels, 2017 by Rich Bowen

> Last Friday in Brussels, CentOS enthusiasts gathered for the annual CentOS Dojo, right before FOSDEM.

Read more at http://tm3.org/dp (http://tm3.org/dp)

FOSDEM Day 0 - CentOS Dojo by Rich Bowen

> FOSDEM starts tomorrow in Brussels, but there's always a number of events the day before.

Read more at http://tm3.org/dq (http://tm3.org/dq)

Gnocchi 3.1 unleashed by Julien Danjou

> It's always difficult to know when to release, and we really wanted to do it earlier. But it seems that each week more awesome work was being done in Gnocchi, so we kept delaying it while having no pressure to push it out.

Read more at http://tm3.org/dr (http://tm3.org/dr)

Testing RDO with Tempest: new features in Ocata by ltoscano

> The release of Ocata, with its shorter release cycle, is close and it is time to start a broader testing (even if one could argue that it is always time for testing!).

Read more at http://tm3.org/ds (http://tm3.org/ds)

Barely Functional Keystone Deployment with Docker by Adam Young

> My eventual goal is to deploy Keystone using Kubernetes. However, I want to understand things from the lowest level on up. Since Kubernetes will be driving Docker for my deployment, I wanted to get things working for a single node Docker deployment before I move on to Kubernetes. As such, you'll notice I took a few short cuts. Mostly, these involve configuration changes. Since I will need to use Kubernetes for deployment and configuration, I'll postpone doing it right until I get to that layer. With that caveat, let's begin.

Read more at http://tm3.org/dt (http://tm3.org/dt)

by Rich Bowen at February 13, 2017 03:35 PM (http://rdoproject.org/blog/2017/02/rdo-blogs-week-of-feb-13/)

Gorka Eguileor (https://gorka.eguileor.com)

iSCSI multipath issues in OpenStack (https://gorka.eguileor.com/iscsi-multipath/)

Multipathing is a technique frequently used in enterprise deployments to increase throughput and reliability on external storage connections, and it's been a little bit of a pain in the neck for OpenStack users. If you've nodded while reading the previous statement, then this post will probably be of interest to you, as we'll be going […]

by geguileo at February 13, 2017 03:05 PM (https://gorka.eguileor.com/iscsi-multipath/)

OpenStack Superuser (http://superuser.openstack.org)

What's new in the world of OpenStack Ambassadors (http://superuser.openstack.org/articles/openstack-ambassadors/)

The OpenStack Ambassador Program is excited to welcome two new volunteers, Lisa-Marie Namphy and Ilya Alekseyev.

Ambassadors act as liaisons between multiple User Groups (https://groups.openstack.org/), the Foundation and the community in their regions. Launched in 2013, the OpenStack Ambassador program (https://www.openstack.org/blog/2013/08/ambassador-program/)aims to create a framework of community leaders to sustainably expand the reach of OpenStack around the world.

Namphy (https://twitter.com/SWDevAngel) will be looking after the United States. She first joined the OpenStack community in 2012, while leading the product marketing team for the OpenStack technology initiative at Hewlett-Packard. Later, she became the San Francisco Bay Area OpenStack User Group organizer and has been running it for the past three years.

Prior to becoming an Ambassador, she has made considerable contributions to the OpenStack community with a book published (https://www.linkedin.com/in/lnamphy) on OpenStack, speaking sessions (https://www.openstack.org/videos/vancouver-2015/what-your-customers-dont-know-about-openstack-might-hurt-you) at seven Summits, five OpenStack Days and recorded dozens of video interviews (https://www.youtube.com/results?search_query=Lisa-Marie+Namphy) on OpenStack. She have also taken part in building the SF Bay Area OpenStack community to nearly 6,000 members.

Namphy tells Superuser that she believes 2017 will be the year of OpenStack adoption. Her goal is to encourage user groups and their organizers to make this a priority initiative as they build their communities. Furthermore, she hopes to mentor fellow community leaders in creating robust communities like the San Francisco Bay chapter.

Alekseyev (https://www.linkedin.com/in/ilya-alekseyev-7a29b310) will be looking after the Russia and the Commonwealth of Independent States. He started working with OpenStack in December 2010, when he made proof-of-concepts for potential customers at Grid Dynamics, also working with a team to contribute to Nova. In addition, he has coordinated the Russian translation team. Alekseyev also helped launch and organize meetups and conferences devoted to OpenStack in Russia and creating User Groups in Moscow, St. Petersburg, Kazan and Kazakhstan.

Alekseyev's goals include to further develop user groups in his region and help them meet the official User Group requirements. In addition, he hopes to organize OpenStack days in his region and facilitate relationships with local universities, other open source and cloud communities, which he hopes will engage new users. Lastly, he will continue his work on promoting OpenStack resources in Russian for the Russian-speaking community.

We're very excited to have them on board. We also bid a farewell to two valued members of our Ambassador team, Kenneth Hui and Sean Roberts. Both of them achieved fantastic things while with us.

Kenneth initiated and mentored the Philadelphia, Maryland and Florida user groups, while also growing the New York City user group to become one of the longest running. In addition, Kenneth contributed to organising the OpenStack Architecture Guide book sprint as well as representing the OpenStack project at various conferences and meet-ups.

Roberts has long been a vocal participant in our meetings and along with Namphy stewarded SFBay, one of the largest OpenStack user groups.

We are grateful for their work and the legacy they will leave for our newcomers to build upon.

If you're interested in becoming an ambassador, you can apply here: https://docs.google.com/forms/d/e/1FAIpQLSc0TWAPr32S5CNh0mvYyYH4vZY-rEF6bWUZ9KhwMbXjq3jovw/viewform (https://docs.google.com/forms/d/e/1FAIpQLSc0TWAPr32S5CNh0mvYyYH4vZY-rEF6bWUZ9KhwMbXjq3jovw/viewform)

Sonia Ramez recently joined the OpenStack Foundation on the community management team as an intern. She's working on the user group process and the Ambassador Program.

The post What's new in the world of OpenStack Ambassadors (http://superuser.openstack.org/articles/openstack-ambassadors/) appeared first on OpenStack Superuser (http://superuser.openstack.org).

by Sonia Ramza at February 13, 2017 12:30 PM (http://superuser.openstack.org/articles/openstack-ambassadors/)

Hugh Blemings (http://hugh.blemings.id.au)

Lwood-20170212 (http://hugh.blemings.id.au/2017/02/13/lwood-20170212/)

Introduction

Welcome to Last week on OpenStack Dev ("Lwood") for the week just past. For more background on Lwood, please refer here (http://hugh.blemings.id.au/openstack/lwood/).

Basic Stats for the week 6 to 12 February for openstack-dev:

~348 Messages (down about 39% relative to the long term average)

~124 Unique threads (down about 31% relative to the long term average)

One of those weeks where I wonder if should ever *speculate what is going to happen with traffic on the list! Much quieter this week relative to average – there does seem to be a trend where traffic falls away a bit around a PTG or Summit so perhaps just a side effect of the proximity to next weeks PTG in Boston. Bit of a shorter Lwood as a result*

Notable Discussions – openstack-dev

New OpenStack Security Notices

Users of Glance may be able to replace active image data [OSSN-0065]

From the summary: "When Glance has been configured with the "show_multiple_locations" option enabled with default policy for set and delete locations, it is possible for a non-admin user having write access to the image metadata to replace active image data."

What is your favourite/most embarrassing IRC gaffe ?

So asks Kendall Nelson in his email (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111816.html) – he's gathering stories from the community as part of an article he's writing. In fairness I won't risk inadvertently stealing his thunder by repeating or summarising the stories here but if you want something to brighten your morning/afternoon, have a quick peek at the thread (http://lists.openstack.org/pipermail/openstack-dev/2017-

February/111816.html) :)

End of Week Wrap-ups, Summaries and Updates

- Horizon (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111925.html) from Richard Jones
- Ironic (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111698.html) courtesy of Ruby Loo
- Nova (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111630.html) by Ed Leafe

People and Projects

Project Team Lead Election Conclusion and Results

Kendall Nelson summarises the results of the recent PTL elections in a post (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111769.html) to the list. Most Projects had the one PTL nominee, those that went to election were Ironic, Keyston, Neutron, QA and Stable Branch Maintenance. Full details in Kendall's message.

Core nominations & changes

A quiet week this week other than the PTL elections winding up

- [Dragonflow] Nominating Xiao Hong Hu (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111841.html)i for core of Dragonflow – Omer Anson

Miscellanea

Further reading

Don't forget these excellent sources of OpenStack news – most recent ones linked in each case

- What's Up, Doc? (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111932.html) by Alexandra Settle
- API Working Group newsletter (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111882.html) – Michael McCune and the API WG
- OpenStack Developer Mailing List Digest (http://lists.openstack.org/pipermail/openstack-dev/2017-January/111139.html) by Mike Perez & Kendall Nelson
- OpenStack news over on opensource.com (https://opensource.com/article/17/2/openstack-news-february-13) by Jason Baker
- OpenStack Foundation Events Page (https://www.openstack.org/community/events/) for a frequently updated list of events

Credits

This weeks edition of Lwood brought to you by Bruce Hornsby (https://en.wikipedia.org/wiki/Bruce_Hornsby) (Scenes from the Southside (https://en.wikipedia.org/wiki/Scenes_from_the_Southside)) and Bruce Springsteen (https://en.wikipedia.org/wiki/Bruce_Springsteen) (Greatest Hits (https://en.wikipedia.org/wiki/Greatest_Hits_(Bruce_Springsteen_album))).

In this my first Lwood post Rackspace (http://www.rackspace.com) I place on record my thanks to the Rack for a great few years and, of course, for supporting producing Lwood as part of my role there. I intend continuing to write Lwood for the foreseeable future modulo what my new (yet to be determined) gig might entail :)

by hugh at February 13, 2017 08:16 AM (http://hugh.blemings.id.au/2017/02/13/lwood-20170212/)

Opensource.com (https://opensource.com/taxonomy/term/5126/feed/feed)

A getting started guide for contributors, Designate's future, and more OpenStack news (https://opensource.com/article/17/2/openstack-news-february-13)

Explore what's happening this week in OpenStack, the open source cloud computing project.

by Jason Baker at February 13, 2017 06:00 AM (https://opensource.com/article/17/2/openstack-news-february-13)

February 12, 2017

Flavio Percoco (https://blog.flaper87.com/)

On communities: When should change happen? (https://blog.flaper87.com/on-communities-when-should-change-happen.html)

One common rule of engineering (and not only engineering, really) is that you don't change something that is not broken. In this context, broken doesn't only refer to totally broken things. It could refer to a piece of code becoming obsolete, or a part of the software not performing well anymore, etc. The point is that it doesn't matter how sexy the change you want to make is, if there's no good reason to make it, then don't. Because the moment you do, you'll break what isn't broken (or known to be broken, at the very least).

Good practices are good for some things, not everything and even the one mentioned above is not an exception. Trying to apply this practice to everything in our lives and everywhere in our jobs is not going to bring the results one would expect. We will soon end up with stalled processes or even worse, as it's the case for communities, we may be dictating the death of the thing we are applying this practice on.

When it comes to communities, I am a strong believer that the sooner we try to improve things, the more we will avoid future issues that could damage our community. If we know there are things that can be improved and we don't do it because there are no signs of the community being broken, we will, in most cases, be damaging the community. Hopefully the example below will help understanding the point I'm making.

Take OpenStack as an example. It's a fully distributed community with people from pretty much everywhere in the world. What this really means is that there are people from different cultures, whose first language is not English, that live in different timezones. One common issues with every new team in OpenStack is finding the best way to communicate across the team. Should the team use IRC? Should the team try video first? Should the team do both? What time is the best time to meet? etc.

The defacto standard mean of communication for instant messaging in OpenStack is IRC. It's accessible from everywhere, it's written, it's logged and it's open. It has been around for ages and it has been used by the community since the very beginning. Some teams, however, have chosen video over IRC because it's just faster. The amount of things that can be covered in a 1h-long call are normally more than the ones covered in a 1h-long IRC meeting. For **some** people it's just easier and faster to talk. For some people. Not everyone, just some people. The community is distributed and diverse, remember?

Now, without getting into the details of whether IRC is better than video calls, let's assume a new (or existing team) decides to start doing video calls. Let's also assume that the technology used is accessible everywhere (no G+ because it is blocked in China, for example) and that the video calls are recorded and made public. For the current size and members of the hypothetical team, video calls are ok. Members feel comfortable and they can all attend at a reasonable time. Technically, there's nothing broken with this setup. Technically, the team could keep using video calls until something happens, until someone actually complains, until something breaks.

This is exactly where problems begin. In a multi-cultural environment we ought to consider that not everyone is used to speaking up and complaining. While I agree the best way to improve a community is by people speaking up, we also have to take into account those who don't do it because they are just not used to it. Based on the scenario described above, these folks are still not part of the project's team and they likely won't be because in order for them to participate in the community, they would have to give up part of who they are.

For the sake of discussion, let's assume that these folks can attend the call but they are not native English speakers. At this point the problem becomes the language barrier. The language barrier is always higher than your level of extroversion. Meaning, you can be a very extrovert person but not being able to speak the language fluently will leave you off of some discussions, which will likely end up in frustration. Written forms of expression are easier than spoken ones. Our brain has more time to process them, reason about them and apply/correct the login before it even tries to come out of our fingers. The same is not true for spoken communication.

I don't want to get too hung up on the video vs IRC discussion, to be honest. The point made is that, when it comes to communities, waiting for people to complain, or for things to be broken, is the wrong approach. Sit down and reflect how you can make the community better, what things are slowing down its growth and what changes would help you be more inclusive. Waiting until there is an actual problem may be the death of your community. The last thing you want to do is to drive the wrong people away.

If you liked this post, you may also like:

- On communities: Empower humans to be amazing (https://blog.flaper87.com/on-communities-empower-humans-to-be-amazing.html)
- Keeping up with the pace of a fast growing community without dying (https://www.youtube.com/watch?v=bW_AEmKbB_o)

by Flavio Percoco at February 12, 2017 11:00 PM (https://blog.flaper87.com/on-communities-when-should-change-happen.html)

Arie Bregman (http://abregman.com)

TripleO: Debugging Overcloud Deployment Failure (http://abregman.com/2017/02/12/tripleo-debugging-overcloud-deployment-failure/)

You run 'openstack overcloud deploy' and after a couple of minutes you find out it failed and if that's not enough, then you open the deployment log just to find a very (very!) long output that doesn't give you an clue as to why the deployment failed. In the following sections we'll see how can […]

by bregman at February 12, 2017 08:45 PM (http://abregman.com/2017/02/12/tripleo-debugging-overcloud-deployment-failure/)

February 10, 2017

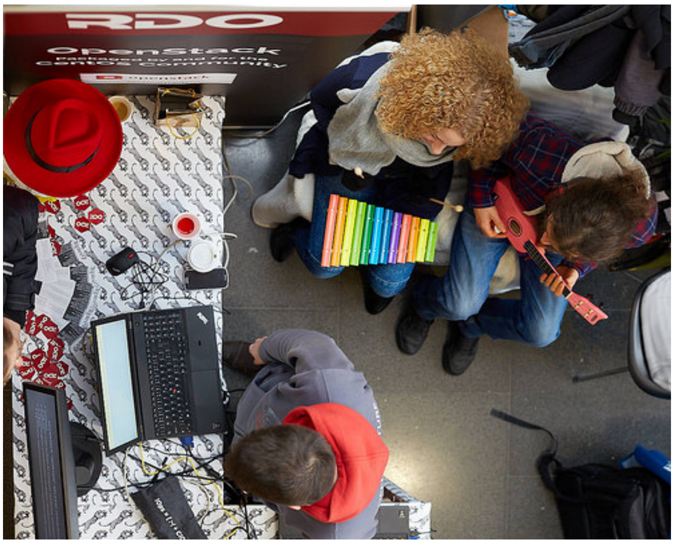
RDO (http://rdoproject.org/blog/)

RDO @ DevConf (http://rdoproject.org/blog/2017/02/rdo-devconf/)

It's been a very busy few weeks in the RDO travel schedule, and we wanted to share some photos with you from RDO's booth at DevConf.cz (http://http//devconf.cz/).

(https://www.flickr.com/photos/rbowen/32444478880/in/album-72157678032428192/)<script async="" charset="utf-8" src="https://embedr.flickr.com/assets/client-code.js"></script>

Led by Eliska Malikova, and supported by our team of RDO engineers, we provided information about RDO and OpenStack, as well as a few impromptu musical performances.


(https://www.flickr.com/photos/rbowen/32444478420/in/album-72157678032428192/)<script async="" charset="utf-8" src="https://embedr.flickr.com/assets/client-code.js"></script>

RDO engineers spun up a small RDO cloud, and later in the day, the people from the Manage IQ (http://manageiq.org/) booth next door set up an instance of their software to manage that cloud, showing that RDO and Manage IQ are better together.

You can see the full album of photos on Flickr (https://www.flickr.com/photos/rbowen/albums/72157678032428192/with/32444478830/).

If you have photos or stories from DevConf, please share them with us on rdo-list. Thanks!

by Rich Bowen at February 10, 2017 09:10 PM (http://rdoproject.org/blog/2017/02/rdo-devconf/)

Daniel P. Berrangé (https://www.berrange.com)

The surprisingly complicated world of disk image sizes (https://www.berrange.com/posts/2017/02/10/the-surprisingly-complicated-world-of-disk-image-sizes/)

When managing virtual machines one of the key tasks is to understand the utilization of resources being consumed, whether RAM, CPU, network or storage. This post will examine different aspects of managing storage when using file based disk images, as opposed to block storage. When provisioning a virtual machine the tenant user will have an idea of the amount of storage they wish the guest operating system to see for their virtual disks. This is the easy part. It is simply a matter of telling 'qemu-img' (or a similar tool) '40GB' and it will create a virtual disk image that is visible to the guest OS as a 40GB volume. The virtualization host administrator, however, doesn't particularly care about what size the guest OS sees. They are instead interested in how much space is (or will be) consumed in the host filesystem storing the image. With this in mind, there are four key figures to consider when managing storage:

- Capacity – the size that is visible to the guest OS
- Length – the current highest byte offset in the file.
- Allocation – the amount of storage that is currently consumed.
- Commitment – the amount of storage that could be consumed in the future.

The relationship between these figures will vary according to the format of the disk image file being used. For the sake of illustration, raw and qcow2 files will be compared since they provide an examples of the simplest file format and the most complicated file format used for virtual machines.

Raw files

In a raw file, the sectors visible to the guest are mapped 1-2-1 onto sectors in the host file. Thus the capacity and length values will always be identical for raw files – the length dictates the capacity and vica-verca. The allocation value is slightly more complicated. Most filesystems do lazy allocation on blocks, so even if a file is 10 GB in length it is entirely possible for it to consume 0 bytes of physical storage, if nothing has been written to the file yet. Such a file is known as "sparse" or is said to have "holes" in its allocation. To maximize guest performance, it is common to tell the operating system to fully allocate a file at time of creation, either by writing zeros to every block (very slow) or via a special system call to instruct it to immediately allocate all blocks (very fast). So immediately after creating a new raw file, the allocation would typically either match the length, or be zero. In the latter case, as the guest writes to various disk sectors, the allocation of the raw file will grow. The commitment value refers the upper bound for the allocation value, and for raw files, this will match the length of the file.

While raw files look reasonably straightforward, some filesystems can create surprises. XFS has a concept of "*speculative preallocation*" where it may allocate more blocks than are actually needed to satisfy the current I/O operation. This is useful for files which are progressively growing, since it is faster to allocate 10 blocks all at once, than to allocate 10 blocks individually. So while a raw file's allocation will usually never exceed the length, if XFS has speculatively preallocated extra blocks, it is possible for the allocation to exceed the length. The excess is usually pretty small though – bytes or KBs, not MBs. Btrfs meanwhile has a concept of "*copy on write*" whereby multiple files can initially share allocated blocks and when one file is written, it will take a private copy of the blocks written. IOW, to determine the usage of a set of files it is not sufficient sum the allocation for each file as that would over-count the true allocation due to block sharing.

QCow2 files

In a qcow2 file, the sectors visible to the guest are indirectly mapped to sectors in the host file via a number of lookup tables. A sector at offset 4096 in the guest, may be stored at offset 65536 in the host. In order to perform this mapping, there are various auxiliary data structures stored in the qcow2 file. Describing all of these structures is beyond the scope of this, read the specification (https://github.com/qemu/qemu/blob/master/docs/specs/qcow2.txt) instead. The key point is that, unlike raw files, the length of the file in the host has no relation to the capacity seen in the guest. The capacity is determined by a value stored in the file header metadata. By default, the qcow2 file will grow on demand, so the length of the file will gradually grow as more data is stored. It is possible to request preallocation, either just of file metadata, or of the full file payload too. Since the file grows on demand as data is written, traditionally it would never have any holes in it, so the allocation would always match the length (the previous caveat wrt to XFS speculative preallocation still applies though). Since the introduction of SSDs, however, the notion of explicitly cutting holes in files has become commonplace. When this is plumbed through from the guest, a guest initiated TRIM request, will in turn create a hole in the qcow2 file, which will also issue a TRIM to the underlying host storage. Thus even though qcow2 files are grow on demand, they may also become sparse over time, thus allocation may be less than the length. The maximum commitment for a qcow2 file is surprisingly hard to get an accurate answer to. To calculate it requires intimate knowledge of the qcow2 file format and even the type of data stored in it. There is allocation overhead from the data structures used to map guest sectors to host file offsets, which is directly proportional to the capacity and the qcow2 cluster size (a cluster is the qcow2 equivalent "sector" concept, except much bigger – 65536 bytes by default). Over time qcow2 has grown other data structures though, such as various bitmap tables tracking cluster allocation and recent writes. With the addition of LUKS support, there will be key data tables. Most significantly though is that qcow2 can internally store entire VM snapshots containing the virtual device state, guest RAM and copy-on-write disk sectors. If snapshots are ignored, it is possible to calculate a value for the commitment, and it will be proportional to the capacity. If snapshots are used, however, all bets are off – the amount of storage that can be consumed is unbounded, so there is no commitment value that can be accurately calculated.

Summary

Considering the above information, for a newly created file the four size values would look like

Format	Capacity	Length	Allocation	Commitment
raw (sparse)	40GB	40GB	0	40GB [1]</a.> (https://www.berrange.com/topics/openstack/feed/#sup1)
raw (prealloc)	40GB	40GB	40GB [1] (https://www.berrange.com/topics/openstack/feed/#sup1)	40GB [1] (https://www.berrange.com/topics/openstack/feed/#sup1)
qcow2 (grow on demand)	40GB	193KB	196KB	41GB [2] (https://www.berrange.com/topics/openstack/feed/#sup2)
qcow2 (prealloc metadata)	40GB	41GB	6.5MB	41GB [2] (https://www.berrange.com/topics/openstack/feed/#sup2)
qcow2 (prealloc all)	40GB	41GB	41GB	41GB [2] (https://www.berrange.com/topics/openstack/feed/#sup2)

[1] *XFS speculative preallocation may cause allocation/commitment to be very slightly higher than 40GB*

[2] *use of internal snapshots may massively increase allocation/commitment*

For an application attempting to manage filesystem storage to ensure any future guest OS write will always succeed without triggering ENOSPC (out of space) in the host, the commitment value is critical to understand. If the length/allocation values are initially less than the commitment, they will grow towards it as the guest writes data. For raw files it is easy to determine commitment (XFS preallocation aside), but for qcow2 files it is unreasonably hard. Even ignoring internal snapshots, there is no API provided by libvirt that reports this value, nor is it exposed by QEMU or its tools. Determining the commitment for a qcow2 file requires the application to not only understand the qcow2 file format, but also directly query the header metadata to read internal parameters such as "cluster size" to be able to then calculate the required value. Without this, the best an application can do is to guess – e.g. add 2% to the capacity of the qcow2 file to determine likely commitment. Snapshots may life even harder, but to be fair, qcow2 internal snapshots are best avoided regardless in favour of external snapshots. The lack of information around file commitment is a clear gap that needs addressing in both libvirt and QEMU.

That all said, ensuring the sum of commitment values across disk images is within the filesystem free space is only one approach to managing storage. These days QEMU has the ability to live migrate virtual machines even when their disks are on host-local storage – it simply copies across the disk image contents too. So a valid approach is to mostly ignore future commitment implied by disk images, and instead just focus on the near term usage. For example, regularly monitor filesystem usage and if free space drops below some threshold, then migrate one or more VMs (and their disk images) off to another host to free up space for remaining VMs.

by Daniel Berrange at February 10, 2017 03:58 PM (https://www.berrange.com/posts/2017/02/10/the-surprisingly-complicated-world-of-disk-image-sizes/)

OpenStack Superuser (http://superuser.openstack.org)

From zero to hero: Your first week as an OpenStack contributor (http://superuser.openstack.org/articles/new-openstack-contributor/)

Each year, hundreds of new contributors start working on OpenStack. Most OpenStack projects have mature code bases and contributors who have been developing the code for several years. Ensuring that a new contributor is pointed in the right direction can often be hard work and a little time consuming.

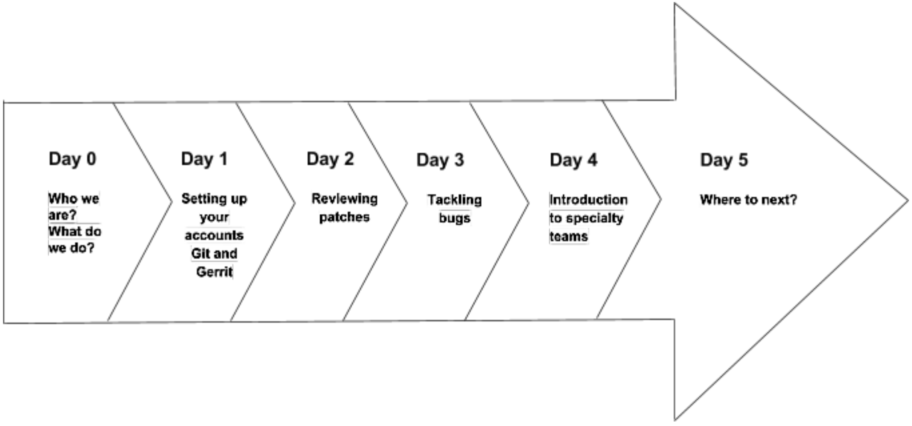
When a newbie asks (a project team lead, jumps in on the mailing list, pipes up over IRC) about how to contribute, the seasoned Stacker will often send them straight to OpenStack Manuals. Why? Because the documentation contribution process is identical to the code contribution process, making it an ideal place to start.

The OpenStack manuals project develops key introductory installation, operation and administration documentation for OpenStack projects. The manuals are a great place to start and provide an invaluable window into each project and how they are operated. This enables the contributor to become familiar with the Git and Gerrit workflow and to feel confident reviewing, responding and reacting to patches and bugs without feeling like they are breaking code lines.

So, from the documentation team to you, here are the Day 0 to Day 5 tips (okay, we'll be honest, this might work out to more than five days, so take your time!) and links to get you set up during your first week. They'll help to ensure that by the end of the week, you can feel (and tell your boss!) that you are an OpenStack contribution guru.

Scenario: You've been told to get started working on OpenStack, ramp up your contributions and start the journey to becoming a core in a project.

- If you have no idea what this means, or entails, start at Day 0.
- If you understand the concepts, but want to know how to get more involved, start at Day 1.


Day 0
Who we are? What do we do?
Day 1
Setting up your accounts Git and Gerrit
Day 2
Reviewing patches
Day 3
Tackling bugs
Day 4
Introduction to specialty teams
Day 5
Where to next?

(http://superuser.openstack.org/wp-content/uploads/2017/02/image00.png)

Day 0:

The OpenStack manuals project provides documentation for various OpenStack projects to promote OpenStack and to develop and maintain tools and processes to ensure the quality and accuracy of documentation.

Our team structure is the same as any other OpenStack project. There is a Project Technical Lead (PTL) (http://docs.openstack.org/project-team-guide/open-development.html#project-team-lead) who ensures that projects and individual tasks are completed and looks after the individual contributor's requirements, if any. The PTL is the community manager for this project.

A team of core reviewers work with the PTL. Core reviewers (http://docs.openstack.org/contributor-guide/docs-review.html) are able to +2 and merge content into the projects for which they have core status. Core status is granted to those who have not only shown care and wisdom in their reviews but have also done a sufficient quantity of reviews and commits.

The OpenStack manuals team looks after the repositories listed here (https://governance.openstack.org/tc/reference/projects/documentation.html#deliverables).

There are no restrictions on who can submit a patch to the OpenStack manuals. If you are looking to get started, the Contributor Guide (http://docs.openstack.org/contributor-guide/index.html) is your source for all information.

To begin contributing, use the First Timers (http://docs.openstack.org/contributor-guide/quickstart/first-timers.html)section to set up your accounts, Git and Gerrit. We treat documentation like code, so we have the same process as development. We also recommend that you join the documentation mailing list (http://lists.openstack.org/cgi-bin/mailman/listinfo/openstack-docs) (and introduce yourself with a few lines) and join the #openstack-doc IRC channel on Freenode. (https://freenode.net/kb/answer/chat)

Day 1:

You have successfully setup your Git and Gerrit workflow and you know what commands to execute to submit a patch for review.

We recommend tackling a low-hanging-fruit bug (https://bugs.launchpad.net/openstack-manuals/+bugs?field.tag=low-hanging-fruit). These bugs have been triaged (http://docs.openstack.org/contributor-guide/doc-bugs.html) by the documentation team and have been designated a "quick fix." The issue should be described within the bug report or within the comments. However, if you do not believe you understand the bug you can do one of the following:

- First, join the OpenStack Documentation Bug (https://launchpad.net/~openstack-doc-bugs)team. Set the status of the bug as "Incomplete" and ask the reporter to provide more information.
- If the bug addresses a problem with project specific documentation, contact the Documentation Liaison (https://wiki.openstack.org/wiki/CrossProjectLiaisons#Documentation) for the specific project.

Our documentation is written in RST (http://docs.openstack.org/contributor-guide/rst-conv.html), so ensure that you have an appropriate text editor to assist with your formatting and syntax. You can also find the guidelines (http://docs.openstack.org/contributor-guide/writing-style.html) for the general writing style that all documentation contributors should follow to ensure consistency throughout all technical publications.

From here, you can either patch the bug and apply the fix, based on the workflow described in the First Timers (http://docs.openstack.org/contributor-guide/quickstart/first-timers.html) section, or you can review some of the patches from other people. Reviewing documentation (https://review.openstack.org/#/q/status:open+(project:%20openstack/openstack-manuals+OR+project:openstack/api-site+OR+project:openstack/sec-guide),n,z) patches is one of the best ways to learn what's in the guides.

Day 2:

Reviewing documentation can be confusing – people are replying to the patch with requests, bug reports and maybe even content specifications.

At the beginning of each release cycle, the project teams work out their key deliverables at the Project Team Gathering (PTG) (https://www.openstack.org/ptg/). This immediately impacts the documentation – what changes upstream **must** change in documentation. This usually comes to the documentation in the form of a bug report (https://bugs.launchpad.net/openstack-manuals/+filebug). The project will report a bug to the OpenStack manuals team by either tagging DocImpact (https://wiki.openstack.org/wiki/Documentation/DocImpact) in the commit message of the original development patch or by filing an entirely new bug with a request for documentation to be updated.

While the project teams work out their key deliverables, the documentation team also has a chance to decide on what deliverables need to be met within the guides. This might relate to technical debt, archiving, or perhaps even a mass change that must occur across all of the guides. This work is tracked through a specification (http://docs.openstack.org/contributor-guide/blueprints-and-specs.html).

All patches up for review (https://review.openstack.org/#/q/status:open+(project:%20openstack/openstack-manuals+OR+project:openstack/api-site+OR+project:openstack/sec-guide),n,z)should link to a specification, bug report, or, at the very least, have a detailed commit message following our Good Practice (https://wiki.openstack.org/wiki/GitCommitMessages) guidelines.

When reviewing the patch, ensure that the committer has explained why they fixed the problem and ensure that what they say matches the output. If you need to build the documentation to review properly, you can use our build tools (http://docs.openstack.org/contributor-guide/docs-builds.html), or you can use the gate jobs, *gate-[REPO-NAME]-tox-doc-publish-checkbuild,* to check the build in your browser.

Here are some guidelines to remember when reviewing someone else's patch: http://docs.openstack.org/contributor-guide/docs-review.html (http://docs.openstack.org/contributor-guide/docs-review.html)

Day 3:

On Day 1, you pushed up your first patch. You made iterations based off requests from other individuals and now, according to the guidelines, your patch can merge with the required +2, +1 and +2a. Do not be concerned if your patch is not ready and merged by Day 3, however. Getting a patch reviewed and then merged can often take time.

Safely merged, you need to know where to go next. If you would like to work on a specific guide or guides and you don't know how to get involved, see Day 4.

If you are interested in staying involved but really don't know what you want to do, we recommend you continue fixing bugs. You can find the list of all bugs that have been confirmed or triaged by the OpenStack manuals team here (https://bugs.launchpad.net/openstack-manuals/+bugs?field.searchtext=&orderby=-importance&field.status%3Alist=CONFIRMED&field.status%3Alist=TRIAGED&field.importance%3Alist=HIGH&field.importance%3Alist=MEDIUM&field.importance%3

Do not work on any bugs that are labeled "New" and do not have "Confirmed" or "Triaged" in the Status column or any bugs that already have an assignee.

Day 4:

One of the things you might come across on the mailing list and in the Contributor Guide (http://docs.openstack.org/contributor-guide/team-structure.html), is the mention of specialty teams. To ensure that each of our guides are looked after and the bugs against the guides are dealt with, the documentation team has assigned specialty team leads. You can find the list of each specialty team lead here (http://docs.openstack.org/contributor-guide/team-structure.html).

To get more involved in an individual guide, contact the relevant individual listed. Each team often has projects happening that require new contributors. You do not have to specialize in only one guide.

Day 5:

Now that you've spent your first week working within the manuals, you have several possible routes to take from here. You can:

1. Continue working with the documentation team and gain insight into how OpenStack is installed, used, and administered by fixing bugs and working with the specialty teams.

2. Find more documentation outlets. Each development project has their own developer-tailored documentation. You can find that, and more information, at: docs.openstack.org/develop/PROJECTNAME
3. Start working on your project of interest! All you need to do is clone the relevant repository (https://github.com/openstack/) and get started!

Good luck!

Join us in #openstack-doc on Freenode (https://freenode.net/kb/answer/chat) to say “hi” and have a chat!

If you choose to contribute to another project, please always come back and document the new changes so that the code can be used by admins.

Cover Photo (https://flic.kr/p/oaeu8j) // CC BY NC (https://creativecommons.org/licenses/by-nc/2.0/)

The post From zero to hero: Your first week as an OpenStack contributor (http://superuser.openstack.org/articles/new-openstack-contributor/) appeared first on OpenStack Superuser (http://superuser.openstack.org).

by Alexandra Settle at February 10, 2017 01:00 PM (http://superuser.openstack.org/articles/new-openstack-contributor/)

OpenStack in Production (http://openstack-in-production.blogspot.com/)

os_type property for Windows images on KVM (http://openstack-in-production.blogspot.com/2017/02/ostype-property-for-windows-images-on.html)

The OpenStack images have a long list of properties which can set to describe the image meta data. The full list is described in the documentation (http://docs.openstack.org/cli-reference/glance-property-keys.html). This blog reviews some of these settings for Windows guests running on KVM, in particular for Windows 7 and Windows 2008R2.

At CERN, we've used a number of these properties to help users filter images such as the OS distribution and version but also added some additional properties for specific purposes such as

- when the image was released (so the images can be sorted by date)
- whether the image is the latest recommended one (such as setting the CentOS 7.2 image to not recommended when CentOS 7.3 comes out)
- which CERN support team provided the image

For a typical Windows image, we have

```
$ glance image-show 9e194003-4608-4fe3-b073-00bd2a774a57
+-------------------+----------------------------------------------------------------+
| Property          | Value                                                          |
+-------------------+----------------------------------------------------------------+
| architecture      | x86_64                                                         |
| checksum          | 27f9cf3e1c7342671a7a0978f5ff288d                               |
| container_format  | bare                                                           |
| created_at        | 2017-01-27T16:08:46Z                                           |
| direct_url        | rbd://b4f463a0-c671-43a8-bd36-e40ab8d233d2/images/9e194003-4   |
| disk_format       | raw                                                            |
| hypervisor_type   | qemu                                                           |
| id                | 9e194003-4608-4fe3-b073-00bd2a774a57                           |
| min_disk          | 40                                                             |
| min_ram           | 0                                                              |
| name              | Windows 10 - With Apps [2017-01-27]                            |
| os                | WINDOWS                                                        |
| os_distro         | Windows                                                        |
| os_distro_major   | w10entx64                                                      |
| os_edition        | DESKTOP                                                        |
| os_version        | UNKNOWN                                                        |
| owner             | 7380e730-d36c-44dc-aa87-a2522ac5345d                           |
| protected         | False                                                          |
| recommended       | true                                                           |
| release_date      | 2017-01-27                                                     |
| size              | 37580963840                                                    |
| status            | active                                                         |
| tags              | []                                                             |
| updated_at        | 2017-01-30T13:56:48Z                                           |
| upstream_provider | https://cern.service-now.com/service-portal/function.do?name   |
| virtual_size      | None                                                           |
| visibility        | public                                                         |
+-------------------+----------------------------------------------------------------+
```

Recently, we have seen some cases of Windows guests becoming unavailable with the BSOD (https://en.wikipedia.org/wiki/Blue_Screen_of_Death) error "CLOCK_WATCHDOG_TIMEOUT (101)". On further investigation, these tended to occur around times of heavy load on the hypervisors such as another guest doing CPU intensive work.

Windows 7 and Windows Server 2008 R2 were the guest OSes where these problems were observed. Later OS levels did not seem to show the same problem.

We followed the standard processes to make sure the drivers were all updated but the problem still occurred.

Looking into the root cause, the Red Hat support articles (https://access.redhat.com/solutions/755943) were a significant help.

```
"In the environment described above, it is possible that 'CLOCK_WATCHDOG_TIMEOUT (101)' BSOD errors could be due to high load
within the guest itself. With virtual guests, tasks may take more time that expected on a physical host. If Windows guests are
```

aware that they are running on top of a Microsoft Hyper-V host, additional measures are taken to ensure that the guest takes this into account, reducing the likelihood of the guest producing a BSOD due to time-outs being triggered."

These suggested to use the `os_type` parameter to help inform the hypervisor to use some additional flags. However, the OpenStack documentation (http://docs.openstack.org/cli-reference/glance-property-keys.html) explained this was a XenAPI only setting (which would not therefore apply for KVM hypervisors).

It is not always clear which parameters to set for an OpenStack image. Setting `os_distro` has a value such as 'windows' or 'ubuntu'. While the flavor of the OS could be determined, the setting of `os_type` is needed to be used by the code.

Thus, in order to get the best behaviour for Windows guests, from our experience, we would recommend setting both the `os_distro` and `os_type` as follows.

- os_distro = 'windows'
- os_type = 'windows'

When the `os_type` parameter is set, some additional XML is added to the KVM configuration following the Kilo (https://bugs.launchpad.net/nova/+bug/1400315) enhancement.

```
<features>
    <acpi/>
    <apic/>
    <hyperv>
      <relaxed state='on'/>
      <vapic state='on'/>
      <spinlocks state='on' retries='8191'/>
    </hyperv>
  </features>
  ....
  <clock offset='localtime'>
    <timer name='pit' tickpolicy='delay'/>
    <timer name='rtc' tickpolicy='catchup'/>
    <timer name='hpet' present='no'/>
    <timer name='hypervclock' present='yes'/>
  </clock>
```

These changes have led to an improvement when running on a loaded hypervisors, especially for Windows 7 and 2008R2 guests. A bug (https://bugs.launchpad.net/openstack-manuals/+bug/1662991) has been opened for the documentation to explain the setting is not Xen only.

Acknowledgements

- Jose Castro Leon performed all of the analysis and testing of the various solutions.

References

- Property keys documentation for OpenStack at http://docs.openstack.org/cli-reference/glance-property-keys.html (http://docs.openstack.org/cli-reference/glance-property-keys.html)
- Additional functionality added to Nova to add hyper-v timer enlightenments for windows guests at https://bugs.launchpad.net/nova/+bug/1400315 (https://bugs.launchpad.net/nova/+bug/1400315)
- Documentation bug report at https://bugs.launchpad.net/openstack-manuals/+bug/1662991 (https://bugs.launchpad.net/openstack-manuals/+bug/1662991)
- Red Hat articles at https://access.redhat.com/solutions/755943 (https://access.redhat.com/solutions/755943) and https://access.redhat.com/node/1471613 (https://access.redhat.com/node/1471613)

<style></style>

by Tim Bell (noreply@blogger.com) at February 10, 2017 07:29 AM (http://openstack-in-production.blogspot.com/2017/02/ostype-property-for-windows-images-on.html)

February 09, 2017

Lance Bragstad (http://lbragstad.com)

Using OpenStack-Ansible to performance test rolling upgrades (http://lbragstad.com/using-openstack-ansible-to-performance-test-rolling-upgrades/)

I've spent the last year or so dabbling with ways to provide consistent performance results for keystone. In addition to that, the keystone community has been trying to implement rolling upgrades support (https://governance.openstack.org/tc/reference/tags/assert_supports-rolling-upgrade.html). Getting both of these tested and in the gate would be a huge step forward for developers and deployers.

Today I hopped into IRC and Jesse (https://twitter.com/odyssey4me) from the OpenStack-Ansible team passed me a review (https://review.openstack.org/#/c/431225) that used OpenStack-Ansible to performance test keystone *during* a rolling upgrade… Since that pretty much qualifies as one of the coolest reviews someone has ever handed me, I couldn't wait to test it out. I was able to do everything on a fresh Ubuntu 16.04 virtual machine with 8 GB of memory, 8 VCPUs and I brought it up to speed following the initial steps provided in the OpenStack-Ansible AIO Guide (http://docs.openstack.org/developer/openstack-ansible/developer-docs/quickstart-aio.html).

<script src="https://gist.github.com/lbragstad/7e83a983fd83f63074feb600910de2f8.js?file=basic-commands.sh"></script>

Next I made sure I had `pip` and `tox` available, as well as `tmux` for my own personal preference. Luckily the OpenStack-Ansible team does a good job of managing binary dependencies in tree (https://github.com/openstack/openstack-ansible-os_keystone/blob/d7141eec5261930ad4991307e1893e800e2ab75c/bindep.txt), which makes getting fresh installs up and off the ground virtually headache-free. Since the patch was still in review at the time of this writing, I went ahead and checked that out of Gerrit.

<script src="https://gist.github.com/lbragstad/7e83a983fd83f63074feb600910de2f8.js?file=checkout-change.sh"></script>

From here, the `os_keystone` role should be able to setup the infrastructure and environment. Another nice thing about the various roles in OpenStack-Ansible is that they isolate `tox` environments much like you would for building docs, syntax linting, or running tests using a specific version of python. In this case, there happens to be one dedicated to upgrades. Behind the scenes this is going to prepare the infrastructure, install lxc, orchestrate multiple installations of the most recent stable keystone release isolated into separate containers (which plays a crucial role in achieving rolling upgrades), install the latest keystone source code from master, and perform a rolling upgrade (whew!). Lucky for us, we only have to run one command.

<script src="https://gist.github.com/lbragstad/7e83a983fd83f63074feb600910de2f8.js?file=tox.sh"></script>

The first time I ran `tox` locally I did get one failure related to the absence of `libpq-dev` while installing requirements for `os_tempest` :

<script src="https://gist.github.com/lbragstad/7e83a983fd83f63074feb600910de2f8.js?file=libpq-dev-failure.output"></script>

Other folks were seeing the same thing, but only locally. For some reason the gate was not hitting this specific issue (maybe it was using wheels?). There is a patch (https://review.openstack.org/#/c/431656) up for review to fix this. After that I reran tox and was rewarded with:

<script src="https://gist.github.com/lbragstad/7e83a983fd83f63074feb600910de2f8.js?file=success.output"></script>

Not only do we see that the rolling upgrade succeeded according to `os_keystone` 's functional tests, but we also see the output from the performance tests. There were 2527 total requests during the execution of the upgrade, 10 of which resulted in an error (could probably use some tweaking here to see if node rotation timing using HAProxy mitigates those?).

Next Steps

Propose a rolling upgrade keystone gate job

Now that we have a consistent way to test rolling upgrades while running a performance script, we can start looping this into other gate jobs. It would be awesome to be able to leverage this work to test every patch proposed to ensure it is not only performant, but also maintains our commitment to delivering rolling upgrades.

Build out the performance script

The performance script is just python that gets fed into Locust (http://locust.io/). The current version (https://github.com/lbragstad/keystone-performance-upgrade/blob/c9b827a5e5398d2783297e1d318adb1a04e1e141/locustfile.py) is really simple and only focuses on authenticating for a token and validating it. Locust has some flexibility that allows writers to add new test cases and even assign different call percentages to different operations (i.e. authenticate for a token 30% of the time and validate 70% of the time). Since it's all python making API calls, Locust test cases are really just functional API tests. This makes it easy to propose patches that add more scenarios as we move forward, increasing our rolling upgrade test coverage. From the output we should be able to inspect which calls failed, just like today when we saw we had 10 authentication/validation failures.

Publish performance results

With running this as part of the gate, it would be a waste to not stash or archive the results from each run (especially if two separate projects are running it). We could even look into running it on dedicated hardware somewhere, similar to the performance testing project (https://github.com/lbragstad/keystone-performance) I was experimenting with last year. The OSIC Performance Bot would technically be a first-class citizen gate job (and we could retire the first iteration of it!). All the results could be stuffed away somewhere and made available for people to write tools that analyze it. I'd personally like to revamp our keystone performance site (http://keystone-performance.lbragstad.com/) to continuously update according to the performance results from the latest master patch. Maybe we could even work some sort of performance view into OpenStack Health (http://status.openstack.org/openstack-health/).

The final bit that helps seal the deal is that we get this at the expense of a single virtual machine. Since OpenStack-Ansible uses containers to isolate services we can feel confident in testing rolling upgrades while only consuming minimal gate resources.

I'm look forward to doing a follow up post as we hopefully start incorporating this into our gate.

by lbragstad at February 09, 2017 10:45 PM (http://lbragstad.com/using-openstack-ansible-to-performance-test-rolling-upgrades/)

Rich Bowen (http://drbacchus.com)

Project Leader (http://drbacchus.com/project-leader/)

I was recently asked to write something about the project that I work on – RDO (http://rdoproject.org/) – and one of the questions that was asked was:

> A healthy project has a visible lead(s). Who is the project lead(s) for this project?

This struck me as a strange question because, for the most part, the open source projects that I choose to work on don't have a project lead, but are, rather, led by community consensus, as well as a healthy dose of "Just Do It". This is also the case with RDO, where decisions are discussed in public on the mailing list, and on IRC meetings, and those that step up to do the work have more practical influence than those that just talk about it.

Now, this isn't to say that nobody takes leadership or ownership of the projects. In many senses, everyone does. But, of course, certain people do rise to prominence from time to time, just based on the volume of work that they do, and these people are the de facto leaders for that moment.

There's a lot of different leadership styles in open source, and a lot of projects do in fact choose to have one technical leader who has the final say on all contributions. That model can work well, and does in many cases. But I think it's important for a project to ask itself a few questions:

- What do we do when a significant number of the community disagrees with the direction that this leader is taking things?
- What happens when the leader leaves? This can happen for many different reasons, from vacation time, to losing interest in the project, to death.
- What do we do when the project grows in scope to the point that a single leader can no longer be an expert on everything?

A strong leader who cares about their project and community will be able to delegate, and designate replacements, to address these concerns. A leader who is more concerned with power or ego than with the needs of their community is likely to fail on one or more of these tests.

But, I find that I greatly prefer projects where project governance is of the people, by the people, and for the people.

by rbowen at February 09, 2017 10:01 PM (http://drbacchus.com/project-leader/)

Maish Saidel-Keesing (http://technodrone.blogspot.com/search/label/OpenStack)

I am Running for the OpenStack User Committee (http://technodrone.blogspot.com/2017/02/i-am-running-for-openstack-user.html)

Two days ago I decided to submit my candidacy for one of the two spots up for election (for the first time!) on the OpenStack User committee.

I am pasting my proposal verbatim (original email link here (http://lists.openstack.org/pipermail/user-committee/2017-February/001680.html))...

Good evening to you all.

As others have so kindly stepped up - I would also like to self-nominate myself for as candidate for the User committee.

I have been involved in the OpenStack community since the Icehouse release.

From day 1, I felt that the user community was not completely accepted as a part of the OpenStack community and that there was a clear and broad disconnect between the two parts of OpenStack.

Instead of going all the way back - and stepping through time to explain who I am and what I have done - I have chosen a few significant points along the way - of where I think I made an impact - sometimes small - but also sometimes a lot bigger.

- The OpenStack Architecture Design Guide [1]. This was my first Opensource project and it was an honor to participate and help the community to produce such a valuable resource.
- Running for the TC for the first time [2]. I was not elected.
- Running for the TC for the second time [3]. Again I was not elected.

  (There has never been a member of the User community elected to a TC seat - AFAIK)

In my original candidacy [2] proposal - I mentioned the inclusion of others.

Which is why I so proud of the achievement of the definition of the AUC from the last cycle and the workgroup [3] that Shamail Tahir and I co-chaired
(Needless to say that a **huge** amount of the credit goes also to all the other members of the WG that were involved!!) in making this happen.

Over the years I think I have tried to make difference (perhaps not always in the right way) - maybe the developer community was not ready for such a drastic change - and I still think that they are not.

Now is a time for change.

I think that the User Committee and these upcoming election (which are the first ever) are a critical time for all of us that are part of the OpenStack community - who contribute in numerous ways - **but do not contribute code**.

The User Committee is now becoming what it should have been from the start, an equal participant in the 3 pillars of OpenStack.

I would like to be a part, actually I would be honored to be a part, of ensuring that this comes to fruition and would like to request your vote for the User Committee.

Now down to the nitty gritty. If elected I would like to focus on the following (but not only):

1. Establishing the User committee as significant part of OpenStack - and continue the amazing collaboration that has been forged over the past two years. The tangible feedback to the OpenStack community provided by the Working Groups have defined clear requirements coming from the trenches and need to be addressed throughout the community as a whole.
2. Expand the AUC constituency - both by adding additional criteria and by encouraging more participation in the community according to the initial defined criteria.
3. Establish a clear and fruitful working relationship with Technical committee - enabling the whole of OpenStack to continue to evolve, produce features and functionality that is not only cutting edge but also fundamental and crucial to anyone and everyone using OpenStack today.

Last but not least - I would like to point you to a blog post I wrote almost a year ago [5].

My views have not changed. OpenStack is evolving and needs participation not only from the developer community (which by the way is facing more than enough of its own challenges) but also from us who use, and operate OpenStack.

For me - we are already in a better place - and things will only get better - regardless of who leads the User committee.

Thank you for your consideration - and I would like to wish the best of luck to all the other candidates.

--
Best Regards,
Maish Saidel-Keesing

[1] http://technodrone.blogspot.com/2014/08/the-openstack-architecture-design-guide.html
(http://technodrone.blogspot.com/2014/08/the-openstack-architecture-design-guide.html)

> ***[2] http://lists.openstack.org/pipermail/openstack-dev/2015-April/062372.html*** (http://lists.openstack.org/pipermail/openstack-dev/2015-April/062372.html)
>
> ***[3] http://lists.openstack.org/pipermail/openstack-dev/2015-September/075773.html*** (http://lists.openstack.org/pipermail/openstack-dev/2015-September/075773.html)
>
> ***[4] https://wiki.openstack.org/wiki/AUCRecognition*** (https://wiki.openstack.org/wiki/AUCRecognition)
>
> ***[5] http://technodrone.blogspot.com/2016/03/we-are-all-openstack-are-we-really.html*** (http://technodrone.blogspot.com/2016/03/we-are-all-openstack-are-we-really.html)

Elections open up on February 13th (https://wiki.openstack.org/wiki/Governance/Foundation/UserCommittee/UC-Election-Feb17) and only those who have been recognized as AUC (Active User Contributors) are eligible to vote.

Don't forget to vote!

by Maish Saidel-Keesing (noreply@blogger.com) at February 09, 2017 09:41 PM (http://technodrone.blogspot.com/2017/02/i-am-running-for-openstack-user.html)

NFVPE @ Red Hat (https://blog.nfvpe.site)

Let's (manually) run k8s on CentOS! (http://dougbtv.com//nfvpe/2017/02/09/kubernetes-on-centos/)

So sometimes it's handy to have a plain-old-Kubernetes running on CentOS 7. Either for development purposes, or to check out something new.
Our goal today is to install Kubernetes by hand on a small cluster of 3 CentOS 7 boxen. We'll spin up some libvirt VMs running CentOS generic cloud images, get Kubernetes spun up on those, and then we'll run a test pod to prove it works. Also, this gives you some exposure to some of the components that are running 'under the hood'.

by Doug Smith at February 09, 2017 08:10 PM (http://dougbtv.com//nfvpe/2017/02/09/kubernetes-on-centos/)

Graham Hayes (http://graham.hayes.ie/)

OpenStack Designate - Where we are. (http://graham.hayes.ie/posts/openstack-designate-where-we-are/)

<figure> </figure>

I have been asked a few times recently "What is the state of the Designate project?", "How is Designate getting on?", and by people who know what is happening "What are you going to do about Designate?".

Needless to say, all of this is depressing to me, and the people that I have worked with for the last number of years to make Designate a truly useful, feature rich project.

> Note
>
> *TL;DR;* for this - Designate is not in a sustainable place.

To start out - Designate has always been a small project. DNS does not have massive *cool* appeal - its not shiny, pretty, or something you see on the front page of HackerNews (unless it breaks - then oh boy do people become DNS experts).

A line a previous PTL for the project used to use, and I have happily robbed is "DNS is like plumbing, no one cares about it until it breaks, and then you are standing knee deep in $expletive". (As an aside, that was the reason we chose the crocodile as our mascot - its basically a dinosaur, old as dirt, and when it bites it causes some serious complications).

Unfortunately that comes over into the development of DNS products sometimes. DNSaaS is a check box on a tender response, an assumption.

We were lucky in the beginning - we had 2 large(ish) public clouds that needed DNS services, and nothing currently existed in the eco-system, so we got funding for a team from a few sources.

We got a ton done in that period - we moved from a v1 API which was synchronous to a new v2 async API, we massively increased the amount of DNS servers we supported, and added new features.

Unfortunately, this didn't last. Internal priorities within companies sponsoring the development changed, and we started to shed contributors, which happens, however disappointing. Usually when this happens if a project is important enough the community will pick up where the previous group left off.

We have yet to see many (meaningful) commits from the community though. We have some great deployers who will file bugs, and if they can put up patch sets - but they are (incredibly valuable and appreciated) tactical contributions. A project cannot survive on them, and we are no exception.

So where does that leave us? Let have a look at how many actual commits we have had:

Commits per cycle

Havana	172
Icehouse	165
Juno	254
Kilo	340
Liberty	327
Mitaka	246
Newton	299
Ocata	98

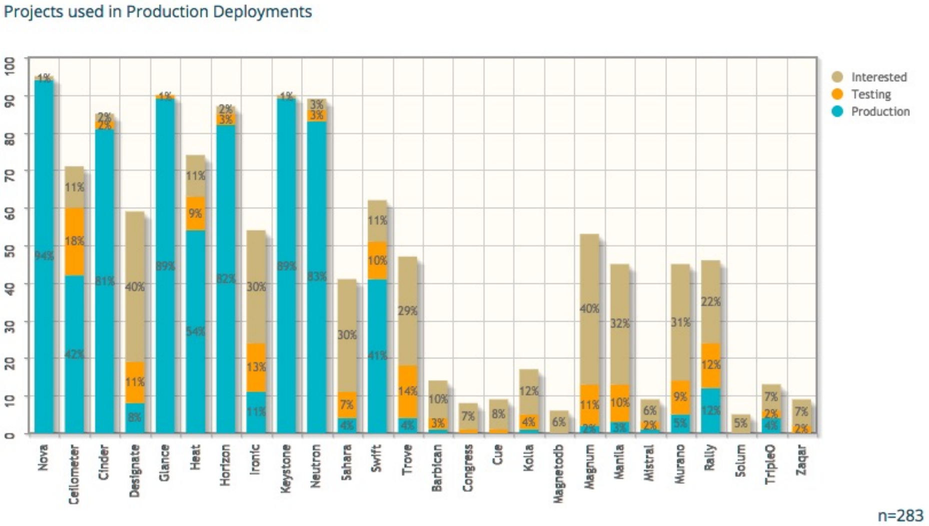
Next cycle, we are going to have 2 community goals:

- Control Plane API endpoints deployment via WSGI
- Python 3.5 functional testing

We would have been actually OK for the tempest one - we were one of the first external repo based plug-ins with designate-tempest-plugin (https://github.com/openstack/designate-tempest-plugin)

For WSGI based APIs, this will be a chunk of work - due to our internal code structure splitting out the API is going to be ... an issue. (and I think it will be harder than most people expect - anyone using olso.service has eventlet imported - I am not sure how that affects running in a WSGI server)

Python 3.5 - I have no idea. We can't even run all our unit tests on python 3.5, so I suspect getting functional testing may be an issue. And, convincing management that re-factoring parts of the code base due to "community goals" or a future potential pay-off can be more difficult than it should.



We now have a situation where the largest "non-core" project [1] (http://graham.hayes.ie/posts/openstack-designate-where-we-are/#id6) in the tent has a tiny number of developers working on it. 42% of deployers are evaluating Designate, so we should see this start to increase.

How did this happen?

Like most situations, there is no single cause.

Certainly there may have been fault on the side of the Designate leadership. We had started out as a small team, and had built a huge amount of trust and respect based on in person interactions over a few years, which meant that there was a fair bit of "tribal knowledge" in the heads of a few people, and that new people had a hard time becoming part of the group.

Also, due to volume of work done by this small group, a lot of users / distros were OK leaving us work - some of us were also running a production designate service during this time, so we knew what we needed to develop, and we had pretty quick feedback when we made a mistake, or caused a bug. All of this resulted in the major development cost being funded by two companies, which left us vulnerable to changes in direction from those companies. Then that shoe dropped. We are now one corporate change of direction from having no cores on the project being paid to work on the project. [2] (http://graham.hayes.ie/posts/openstack-designate-where-we-are/#id7)

Preceding this, the governance of OpenStack changed to the Big Tent (https://governance.openstack.org/tc/resolutions/20141202-project-structure-reform-spec.html) While this change was a good thing for the OpenStack project as a whole it had quite a bad impact on us.

Pre Big Tent, you got integrated. This was at least a cycle, where you moved docs to docs.openstack.org, integrated with QA testing tooling, got packaged by Linux distros, and build cross project features.

When this was a selective thing, there was teams available to help with that, docs teams would help with content (and tooling - docs was a mass of XML back then), QA would help with tempest and devstack, horizon would help with panels.

In Big Tent, there just wasn't resources to do this - the scope of the project expansion was huge. However the big tent happened (in my opinion - I have written about this before) before the horizontal / cross project teams were ready. They stuck to covering the "integrated" projects, which was all they could do at the time.

This left us in a position of having to reimplement tooling, figure out what tooling we did have access to, and migrate everything we had on our own. And, as a project that (at our peak level of contribution) only ever had 5% of the number of contributors compared to a project like nova, this put quite a load on our developers. Things like grenade, tempest and horizon plug-ins, took weeks to figure out all of which took time from other vital things like docs, functional tests and getting designate into other tools.

One of the companies who invested in designate had a QE engineer that used to contribute, and I can honestly say that the quality of our testing improved 10 fold during the time he worked with us. Not just from in repo tests, but from standing up full deployment stacks, and trying to break them - we learned a lot about how we could improve things from his expertise.

Which is kind of the point I think. Nobody is amazing at everything. You need people with domain knowledge to work on these areas. If you asked me to do a multi-node grenade job, I would either start drinking, throw my laptop at you or do both.

We still have some of these problems to this day - most of our docs are in a messy pile in docs.openstack.org/developer/designate (http://docs.openstack.org/developer/designate) while we still have a small amount of old functional tests that are not ported from our old non plug-in style.

All of this adds up to make projects like Designate much less attractive to users - we just need to look at the project navigator (https://www.openstack.org/software/releases/newton/components/designate) to see what a bad image potential users get of us. [3] (http://graham.hayes.ie/posts/openstack-designate-where-we-are/#id8) This is for a project that was ran as a full (non beta) service in a public cloud. [4] (http://graham.hayes.ie/posts/openstack-designate-where-we-are/#id9)

Where too now then?

Well, this is where I call out to people who actually use the project - don't jump ship and use something else because of the picture I have painted. We are a dedicated team, who cares about the project. We just need some help.

I know there are large telcos who use Designate. I am sure there is tooling, or docs build up in these companies that could be very useful to the project.

Nearly every commercial OpenStack distro has Designate. Some have had it since the beginning. Again, developers, docs, tooling, testers, anything and everything is welcome. We don't need a massive amount of resources - we are a small ish, stable, project.

We need developers with upstream time allocated, and the budget to go to events like the PTG - for cross project work, and internal designate road map, these events form the core of how we work.

We also need help from cross project teams - the work done by them is brilliant but it can be hard for smaller projects to consume. We have had a lot of progress since the Leveller Playing Field (http://graham.hayes.ie/posts/openstack-a-leveler-playing-field/) debate, but a lot of work is still optimised for the larger teams who get direct support, or well resourced teams who can dedicate people to the implementation of plugins / code.

As someone I was talking to recently said - AWS is not winning public cloud because of commodity compute (that does help - a lot), but because of the added services that make using the cloud, well, cloud like. OpenStack needs to decide that either it is just compute, or if it wants the eco-system. [5] (http://graham.hayes.ie/posts/openstack-designate-where-we-are/#id10) Designate is far from alone in this.

I am happy to talk to anyone about helping to fill in the needed resources - Designate is a project that started in the very office I am writing this blog post in, and something I want to last.

For a visual this is Designate team in Atlanta, just before we got incubated.

and this was our last mid cycle:

and in Atlanta at the PTG, there will be two of us.

[1] (http://graham.hayes.ie/posts/openstack-designate-where-we-are/#id1) In the Oct-2016 (https://www.openstack.org/analytics) User Survey Designate was deployed in 23% of clouds

[2] (http://graham.hayes.ie/posts/openstack-designate-where-we-are/#id2) I have been lucky to have a management chain that is OK with me spending some time on Designate, and have not asked me to take time off for Summits or Gatherings, but my day job is working on a completely different project.

[3] (http://graham.hayes.ie/posts/openstack-designate-where-we-are/#id3) I do have other issues with the metrics - mainly that we existed before leaving stackforge, and some of the other stats are set so high, that non "core" projects will probably never meet them.

[4] (http://graham.hayes.ie/posts/openstack-designate-where-we-are/#id4) I recently went to an internal training talk, where they were talking about new features in Newton. There was a whole slide about how projects had improved, or gotten worse on these scores. A whole slide. With tables of scores, and I think there may have even been a graph.

[5] (http://graham.hayes.ie/posts/openstack-designate-where-we-are/#id5) Now, I am slightly biased, but I would argue that DNS is needed in commodity compute, but again, that is my view.

by Graham Hayes at February 09, 2017 06:38 PM (http://graham.hayes.ie/posts/openstack-designate-where-we-are/)

OpenStack Superuser (http://superuser.openstack.org)

CERN'S expanding cloud universe (http://superuser.openstack.org/articles/cern-expanding-cloud-universe/)

CERN is rapidly expanding OpenStack cores in production as it accelerates work on understanding the mysteries of the universe.

The European Organization for Nuclear Research currently has over 190,000 cores in production and plans to add another 100,000 in the next six months, says Spyros Trigazis, adding that about 90 percent of CERN's compute resources are now delivered on OpenStack.

Trigazis (http://openlab.cern/about/people/spyridon-trigazis), who works on the compute management and provisioning team, offered a snapshot of all things cloud at CERN in a presentation at the recent CentOS Dojo (https://wiki.centos.org/Events/Dojo/Brussels2017) in Brussels.

RDO's Rich Bowen (https://twitter.com/rbowen) shot the video, which runs through CERN's three-and-a-half years of OpenStack in production as well as what's next for the humans in the CERN loop — the OpenStack team, procurement and software management and LinuxSoft, Ceph and DBoD teams.

Trigazis also outlined the container infrastructure, which uses OpenStack Magnum (http://superuser.openstack.org/articles/a-primer-on-magnum-openstack-containers-as-a-service/) to treat container orchestration engines (COEs) as first-class resources. Since Q4 2016, CERN has been in production with Magnum (http://superuser.openstack.org/articles/openstack-magnum-on-the-cern-production-cloud/) providing support for Docker Swarm, Kubernetes and Mesos a well as storage drivers for (CERN-specific) EOS and CernVM File System (CVMFs). Trigazis says that many users are interested in containers and usage has been ramping up around GitLab continuous integration, Jupyter/Swan and FTS. CERN is currently using the Newton release (http://superuser.openstack.org/articles/openstack-newton-new-landmark-open-source/), with "cherry-picks," he adds.

> Lots of #opensource (https://twitter.com/hashtag/opensource?src=hash) @CERN (https://twitter.com/CERN)! Great talk from Spyros Trigazis @CentOS (https://twitter.com/CentOS) Dojo on their @OpenStack (https://twitter.com/OpenStack) deployment featuring @RDOcommunity (https://twitter.com/RDOcommunity) @puppetize (https://twitter.com/puppetize) pic.twitter.com/tcZwhAPvBk (https://t.co/tcZwhAPvBk)
>
> — Unix (@UNIXSA) February 3, 2017 (https://twitter.com/UNIXSA/status/827502676137041920)

<script async="async" charset="utf-8" src="http://platform.twitter.com/widgets.js"></script>

Upcoming services include baremetal with Ironic (https://wiki.openstack.org/wiki/Ironic); the API server and conductor are already deployed and the first node is to come this month. Another is workflow service Mistral (https://wiki.openstack.org/wiki/Mistral) used to simplify operations, create users and clean up resources. It's already deployed and right now the team is testing prototype workflows. FileShare service Manila (https://wiki.openstack.org/wiki/Manila), which has been in pilot mode since Q4 of 2016, will be used to share configuration and certificates.

You can catch the entire 19-minute presentation on YouTube (https://www.youtube.com/watch?v=fz3XIvkf8S4&feature=youtu.be) or more videos from CentOS Dojo on the RDO blog (https://www.rdoproject.org/blog/).

For updates from the CERN cloud team, check out the OpenStack in Production blog (http://openstack-in-production.blogspot.com).

Cover Photo (https://www.flickr.com/photos/arselectronica/6032157177/) // CC BY NC (https://creativecommons.org/licenses/by-nc/2.0/)

The post CERN'S expanding cloud universe (http://superuser.openstack.org/articles/cern-expanding-cloud-universe/) appeared first on OpenStack Superuser (http://superuser.openstack.org).

by Nicole Martinelli at February 09, 2017 01:09 PM (http://superuser.openstack.org/articles/cern-expanding-cloud-universe/)

February 08, 2017

Daniel P. Berrangé (https://www.berrange.com)

Commenting out XML snippets in libvirt guest config by stashing it as metadata (https://www.berrange.com/posts/2017/02/08/commenting-out-xml-snippets-in-libvirt-guest-config-by-stashing-it-as-metadata/)

Libvirt uses XML as the format for configuring objects it manages, including virtual machines. Sometimes when debugging / developing it is desirable to comment out sections of the virtual machine configuration to test some idea. For example, one might want to temporarily remove a secondary disk. It is not always desirable to just delete the configuration entirely, as it may need to be re-added immediately after. XML has support for comments `<!-- .... some text -->` which one might try to use to achieve this. Using comments in XML fed into libvirt, however, will result in an unwelcome suprise – the commented out text is thrown into /dev/null by libvirt.

This is an unfortunate consequence of the way libvirt handles XML documents. It does not consider the XML document to be the master representation of an object's configuration – a series of C structs are the actual internal representation. XML is simply a data interchange format for serializing structs into a text format that can be interchanged with the management application, or persisted on disk. So when receiving an XML document libvirt will parse it, extracting the pieces of information it cares about which are they stored in memory in some structs, while the XML document is discarded (along with the comments it contained). Given this way of working, to preserve comments would require libvirt to add 100's of extra fields to its internal structs and extract comments from every part of the XML document that might conceivably contain them. This is totally impractical to do in realityg. The alternative would be to consider the parsed

XML DOM as the canonical internal representation of the config. This is what the libvirt-gconfig library in fact does, but it means you can no longer just do simple field accesses to access information – getter/setter methods would have to be used, which quickly becomes tedious in C. It would also involve re-refactoring almost the entire libvirt codebase so such a change in approach would realistically never be done.

Given that it is not possible to use XML comments in libvirt, what other options might be available ?

Many years ago libvirt added the ability to store arbitrary user defined metadata in domain XML documents. The caveat is that they have to be located in a specific place in the XML document as a child of the <metadata> tag, in a private XML namespace. This metadata facility to be used as a hack to temporarily stash some XML out of the way. Consider a guest which contains a disk to be “commented out”:

```
<domain type="kvm">
  ...
  <devices>
    ...
    <disk type='file' device='disk'>
    <driver name='qemu' type='raw'/>
    <source file='/home/berrange/VirtualMachines/demo.qcow2'/>
      <target dev='vda' bus='virtio'/>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0'/>
    </disk>
    ...
  </devices>
</domain>
```

To stash the disk config as a piece of metadata requires changing the XML to

```
<domain type="kvm">
  ...
  <metadata>
    <s:disk xmlns:s="http://stashed.org/1" type='file' device='disk'>
      <driver name='qemu' type='raw'/>
      <source file='/home/berrange/VirtualMachines/demo.qcow2'/>
      <target dev='vda' bus='virtio'/>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0'/>
    </s:disk>
  </metadata>
  ...
  <devices>
    ...
  </devices>
</domain>
```

What we have done here is

– Added a <metadata> element at the top level
– Moved the <disk> element to be a child of <metadata> instead of a child of <devices>
– Added an XML namespace to <disk> by giving it an ‘s:’ prefix and associating a URI with this prefix

Libvirt only allows a single top level metadata element per namespace, so if there are multiple tihngs to be stashed, just give them each a custom namespace, or introduce an arbitrary wrapper. Aside from mandating the use of a unique namespace, libvirt treats the metadata as entirely opaque and will not try to intepret or parse it in any way. Any valid XML construct can be stashed in the metadata, even invalid XML constructs, provided they are hidden inside a CDATA block. For example, if you’re using virsh edit to make some changes interactively and want to get out before finishing them, just stash the changed in a CDATA section, avoiding the need to worry about correctly closing the elements.

```
<domain type="kvm">
  ...
  <metadata>
    <s:stash xmlns:s="http://stashed.org/1">
    <![CDATA[
      <disk type='file' device='disk'>
        <driver name='qemu' type='raw'/>
        <source file='/home/berrange/VirtualMachines/demo.qcow2'/>
        <target dev='vda' bus='virtio'/>
        <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0'/>
      </disk>
      <disk>
        <driver name='qemu' type='raw'/>
        ...i'll finish writing this later...
    ]]>
    </s:stash>
  </metadata>
  ...
  <devices>
    ...
  </devices>
</domain>
```

Admittedly this is a somewhat cumbersome solution. In most cases it is probably simpler to just save the snippet of XML in a plain text file outside libvirt. This metadata trick, however, might just come in handy some times.

As an aside the real, intended, usage of the <metdata> facility is to allow applications which interact with libvirt to store custom data they may wish to associated with the guest. As an example, the recently announced libvirt websockets console proxy (https://www.berrange.com/posts/2017/01/26/announce-new-libvirt-console-proxy-project/) uses it to record which consoles are to be exported. I know of few other real world applications using this metadata feature, however, so it is worth remembering it exists :-) System administrators are free to use it for local book keeping purposes too.

by Daniel Berrange at February 08, 2017 07:14 PM (https://www.berrange.com/posts/2017/02/08/commenting-out-xml-snippets-in-libvirt-guest-config-by-stashing-it-as-metadata/)

OpenStack Superuser (http://superuser.openstack.org)

User Committee Elections are live (http://superuser.openstack.org/articles/user-committee-elections/)

OpenStack has been a vast success and continues to grow. Additional ecosystem partners are enhancing support for OpenStack and it has become more and more vital that the communities developing services around OpenStack lead and influence the products movement.

The OpenStack User Committee (https://www.openstack.org/foundation/user-committee/) helps increase operator involvement, collects feedback from the community, works with user groups around the globe, and parses through user survey data, to name a few. Users are critical, and the User Committee aims to represent the user. With all the growth we are seeing with OpenStack, we are looking to expand the User Committee and have kicked off an election.

We are looking to elect two (2) User Committee members for this election. These User Committee seats will be valid for a one-year term. For this election, the Active User Contributors (AUC) community will review the candidates and vote.

So what makes an awesome candidate for the User Committee?

Well, to start, the nominee has to be an individual member of the OpenStack Foundation who is an Active User Contributor (AUC) (http://superuser.openstack.org/articles/how-to-become-an-openstack-active-user-contributor/). Additionally, below are a few things that will make you stand out:

- If you are an OpenStack end-user and/or operator
- An OpenStack contributor from the User Committee working groups
- Actively engaged in the OpenStack community
- Organizer of an OpenStack local User Group meetup

Beyond the kinds of community activities you are already engaged in, the User Committee role adds some additional work. The User Committee usually interacts on e-mail to discuss any pending topics. Prior to each Summit, we spend a few hours going through the User Survey results and analyzing the data.

You can nominate yourself or someone else by sending an email to the user-committee@lists.openstack.org (mailto:user-committee@lists.openstack.org) mailing-list, with the subject: “UC candidacy” by **Friday, February 10**, 05:59 UTC (https://www.timeanddate.com/worldclock/timezone/utc).

The email should include a description of the candidate and what the candidate hopes to accomplish.

We look forward to receiving your submissions!

The post User Committee Elections are live (http://superuser.openstack.org/articles/user-committee-elections/) appeared first on OpenStack Superuser (http://superuser.openstack.org).

by Superuser at February 08, 2017 12:02 PM (http://superuser.openstack.org/articles/user-committee-elections/)

Bernard Cafarelli (http://blog.cafarelli.fr)

Tracking Service Function Chaining with Skydive (http://blog.cafarelli.fr/2017/02/tracking-service-function-chaining-with-skydive/)

Skydive (https://github.com/skydive-project/skydive) is “an open source real-time network topology and protocols analyzer”. It is a tool (with CLI and web interface) to help analyze and debug your network (OpenStack, OpenShift, containers, …). Dropped packets somewhere? MTU issues? Routing problems? These are some issues where running skydive whill help.

So as an update on my previous demo post (http://blog.cafarelli.fr/2016/11/service-function-chaining-demo-with-devstack/) (this time based on the Newton release), let’s see how we can trace SFC with this analyzer!

devstack installation

Not a lot of changes here, check out devstack on the *stable/newton* branch, grab the local.conf file I prepared (configure to use skydive 0.9 release) and run “./stack.sh”!

For the curious, the SFC/Skydive specific parts are:

```
# SFC
enable_plugin networking-sfc https://git.openstack.org/openstack/networking-sfc stable/newton

# Skydive
enable_plugin skydive https://github.com/skydive-project/skydive.git refs/tags/v0.9.0
enable_service skydive-agent skydive-analyzer
```

Skydive web interface and demo instances

Before running the script to configure the SFC demo instances, open the skydive web interface (it listens on port 8082, check your instance firewall if you cannot connect):

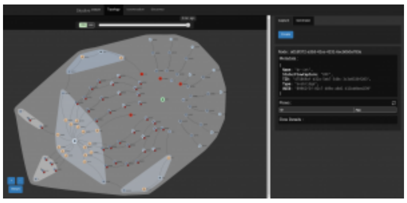
```
http://${your_devstack_ip}:8082
```

The login was configured with devstack, so if you did not change, use *admin*/*pass123456*.
Then add the demo instances as in the previous demo:

```
$ git clone https: //github.com/voyageur/openstack-scripts.git -b sfc_newton_demo
$ ./openstack-scripts/simple_sfc_vms.sh
```

And watch as your cloud goes from “empty” to “more crowded”:

(http://blog.cafarelli.fr/wp-content/uploads/2017/02/skydive_1_initial.png)



(http://blog.cafarelli.fr/wp-content/uploads/2017/02/skydive_2_with_vms.png)

Skydive CLI, start traffic capture

Now let's enable traffic capture on the integration bridge (br-int), and all tap interfaces (more details on the skydive CLI available in the documentation (https://skydive-project.github.io/skydive/getting-started/client/)):

```
$ export SKYDIVE_USERNAME=admin
$ export SKYDIVE_PASSWORD=pass123456
$ /opt/stack/go/bin/skydive --conf /tmp/skydive.yaml client capture create --gremlin "G.V().Has('Name', 'br-int', 'Type', 'ovsbridge')"
$ /opt/stack/go/bin/skydive --conf /tmp/skydive.yaml client capture create --gremlin "G.V().Has('Name', Regex('^tap.*'))"
```

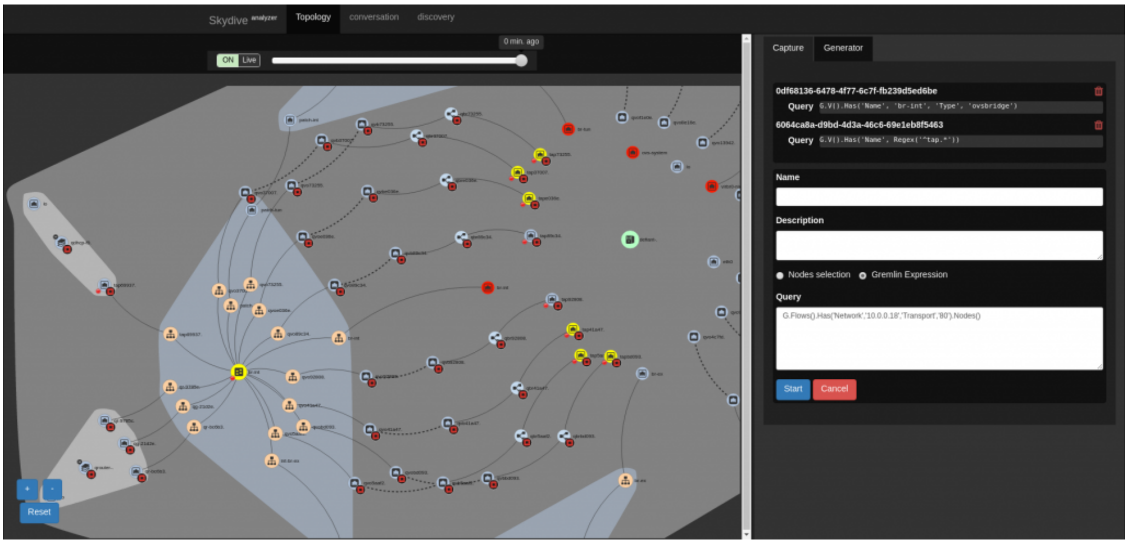
Note this can be done in the web interface too, but I wanted to show both interfaces.

Track a HTTP request diverted by SFC

Make a HTTP request from the source VM to the destination VM (see previous post (http://blog.cafarelli.fr/2016/11/service-function-chaining-demo-with-devstack/) for details). We will highlight the nodes where this request has been captured: in the GUI, click on the capture create button, select "Gremlin expression", and use the query:

```
G.Flows().Has('Network','10.0.0.18','Transport','80').Nodes()
```

This expression reads as "on all captured flows matching IP address 10.0.0.18 and port 80, show nodes". With the CLI you would get a nice JSON output of these nodes, here in the GUI these nodes will turn yellow:



0df68136-6478-4f77-6c7f-fb239d5ed6be
Query `G.V().Has('Name', 'br-int', 'Type', 'ovsbridge')`

6064ca8a-d9bd-4d3a-46c6-69e1eb8f5463
Query `G.V().Has('Name', Regex('^tap.*'))`

Name

Description

○ Nodes selection ● Gremlin Expression

Query
G.Flows().Has('Network','10.0.0.18','Transport','80').Nodes()

(http://blog.cafarelli.fr/wp-content/uploads/2017/02/skydive_3_highlight.png)

If you look at our tap interface nodes, you will see that two are not highlighted. If you check their IDs, you will find that they belong to the same service VM, the one in group 1 that did not get the traffic.

If you want to single out a request, in the skydive GUI, select one node where capture is active (for example br-int). In the flows table, select the request, scroll down to get its layer 3 tracking ID "L3TrackingID" and use it as Gremlin expression:

```
G.Flows().Has('L3TrackingID','5a7e4bd292e0ba60385a9cafb22cf37d744a6b46').Nodes()
```

Going further

Now it's your time to experiment! Modify the port chain, send a new HTTP request, get its L3TrackingID, and see its new path. I find the latest ID quickly with this CLI command (we will see how the skydive experts will react to this):

```
$ /opt/stack/go/bin/skydive --conf /tmp/skydive.yaml client topology query --gremlin "G.Flows().Has('Network','10.0.0.18','Transport','80').Limit(1)"
| jq ".[0].L3TrackingID"
```

You can also check each flow in turn, following the paths from a VM to another one, go further with SFC, or learn more about skydive:

- Project site: https://github.com/skydive-project/skydive (https://github.com/skydive-project/skydive)
- Documentation: https://skydive-project.github.io/skydive/ (https://skydive-project.github.io/skydive/)
- Another blog post on devstack deployment: https://blogs.rdoproject.org/7921/skydive-plugin-for-devstack (https://blogs.rdoproject.org/7921/skydive-plugin-for-devstack)
- The YouTube channel (https://www.youtube.com/channel/UCS2x2VrJKLA_qJWzEYOQRHA) with some demo videos

by Bernard Cafarelli at February 08, 2017 11:43 AM (http://blog.cafarelli.fr/2017/02/tracking-service-function-chaining-with-skydive/)

NFVPE @ Red Hat (https://blog.nfvpe.site)

Automated OSP deployments with Tripleo Quickstart (http://teknoarticles.blogspot.com/2017/02/automated-osp-deployments-with-tripleo.html)

In this article I'm going to show a method for automating OSP (RedHat OpenStack platform) deployments. These automated deployments can be very useful for CI, or simply to experiment and test with the system. Components involved ansible-cira: set of playbooks to deploy Jenkins, jenkins-job-builder and an optional ELK stack. This will install a ready to use system with all the preconfigured jobs (including OSP10 deployments and image building). ansible-cira jenkins-jobs: A set of job templates and macros, using jenkins-job-builder syntax, that get converted into Jenkins jobs for building the OSP base images and for deploying the system. ansible-cira job-configs: A...

by Yolanda Robla Mota at February 08, 2017 10:42 AM (http://teknoarticles.blogspot.com/2017/02/automated-osp-deployments-with-tripleo.html)

February 07, 2017

Cloudwatt (https://dev.cloudwatt.com/en/blog/index.html)

Instance backup script (https://dev.cloudwatt.com/en/blog/instance-backup-script.html)

This script is designed to allow you to schedule backups of your Nova instances.

You can set a policy to retain your backups.

Since the script is written in python, it can be run from any machine on which python is installed.

You do not need to install OpenStack clients to run this script.

Retrieving the script from the Git repository:

```
$ git clone https://github.com/myorg92/os-nova-backup.git
$ cd os-nova-backup/
```

Running the script requires that some environment variables be loaded:

`OS_USERNAME` : The user name of your OpenStack account On Cloudwatt = email address

`OS_PASSWORD` : The password of your OpenStack account

`OS_TENANT_ID` : Identifiant of the Openstack tenant

`OS_AUTH_URL` : The URL of the identification service On Cloudwatt = https://identity.fr2.cloudwatt.com/v2.0

`OS_COMPUTE_URL` : The URL of the Compute service On Cloudwatt = https://compute.fr1.cloudwatt.com/v2/ « tenant-id » for fr1 https://compute.fr2.cloudwatt.com/v2/« tenant-id » for fr2

```
utilisation: python backup.py <server_id> <server_name> <weekly|daily> <rotation>
```

Positional arguments:

`< server_id >` ID of the instance to backup

`< server_name >` Name of tje backup image.

`< weekly|daily >` Type of backup : “daily” or “weekly”.

`<rotation>` Parameter Int representing the number of backups to keep

You can schedule backups via crontab or Jenkins for example. Here are 2 examples of cron task:

Weekly backup with a retention of 4 weeks

```
0 2 * * 6 python backup.py 0a49912c-1661-4d92-b469-53dfea7ce3da Myinstance weekly 4
```

Daily backup with 1 week retention

```
0 2 1-5,7 * *  python backup.py 0a49912c-1661-4d92-b469-53dfea7ce3da Myinstance daily 6
```

Backups will be named as “instance name” - “year / day / hour / minute” - “type of backup” Example: myinstance-20171281425-weekly

This is actually a snapshot that will be stored as an image in your tenant. This image can be used to launch new instances or to restore an existing instance.

This version of the script does not allow yet the restoration (rebuild). It is therefore necessary to use the CLI:

```
$ nova rebuild 0a49912c-1661-4d92-b469-53dfea7ce3da myinstance-20171281425-weekly
```

by Kamel Yamani at February 07, 2017 11:00 PM (https://dev.cloudwatt.com/en/blog/instance-backup-script.html)

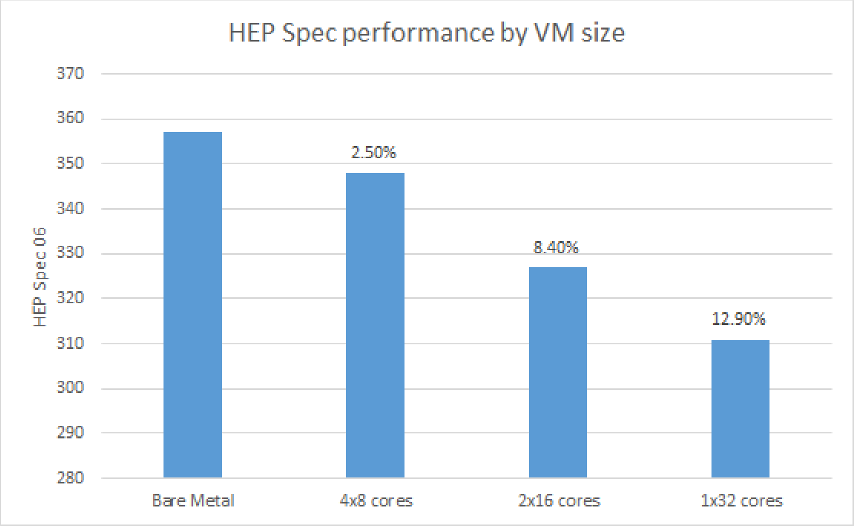
OpenStack in Production (http://openstack-in-production.blogspot.com/)

Tuning hypervisors for High Throughput Computing (http://openstack-in-production.blogspot.com/2015/08/kvm-and-hyper-v-comparison-for-high.html)

Over the past set of blogs, we've looked at a number of different options for tuning High Energy Physics workloads in a KVM environment such as the CERN OpenStack cloud.

This is a summary of the findings using the HEPSpec 06 (http://iopscience.iop.org/1742-6596/219/5/052009/pdf/1742-6596_219_5_052009.pdf) benchmark on KVM and a comparison with Hyper-V for the same workload.

For KVM on this workload, we saw a degradation in performance on large VMs.



(http://4.bp.blogspot.com/-oETOS9_n9pE/VcIvOlPIpFI/AAAAAAAALkI/tSwDbVB3ebo/s1600/vmoverhead.png)

Results for other applications may vary so each option should be verified for the target environment. The percentages from our optimisations are not necessarily additive but give an indication of the performance improvements to be expected. After tuning, we saw around 5% overhead from the following improvements.

Option	Improvement	Comments
CPU topology (http://openstack-in-production.blogspot.fr/2015/08/openstack-cpu-topology-for-high.html)	~0	The primary focus for this function was not for performance so result is as expected
Host Model (http://openstack-in-production.blogspot.fr/2015/08/cpu-model-selection-for-high-throughput.html)	4.1-5.2%	Some impacts on operations such as live migration
Turn EPT off (http://openstack-in-production.blogspot.fr/2015/08/ept-and-ksm-for-high-throughput.html)	6%	Open bug report for CentOS 7 guest on CentOS 7 hypervisor
Turn KSM off (http://openstack-in-production.blogspot.fr/2015/08/ept-and-ksm-for-high-throughput.html)	0.9%	May lead to an increase in memory usage
NUMA in guest (http://openstack-in-production.blogspot.fr/2015/08/numa-and-cpu-pinning-in-high-throughput.html)	~9%	Needs Kilo or later to generate this with OpenStack
CPU Pinning (http://openstack-in-production.blogspot.fr/2015/08/numa-and-cpu-pinning-in-high-throughput.html) (https://www.blogger.com/)	~3%	Needs Kilo or later (cumulative on top of NUMA)

Different applications will see a different range of improvements (or even that some of these options degrade performance). Experiences from other workload tuning would be welcome.

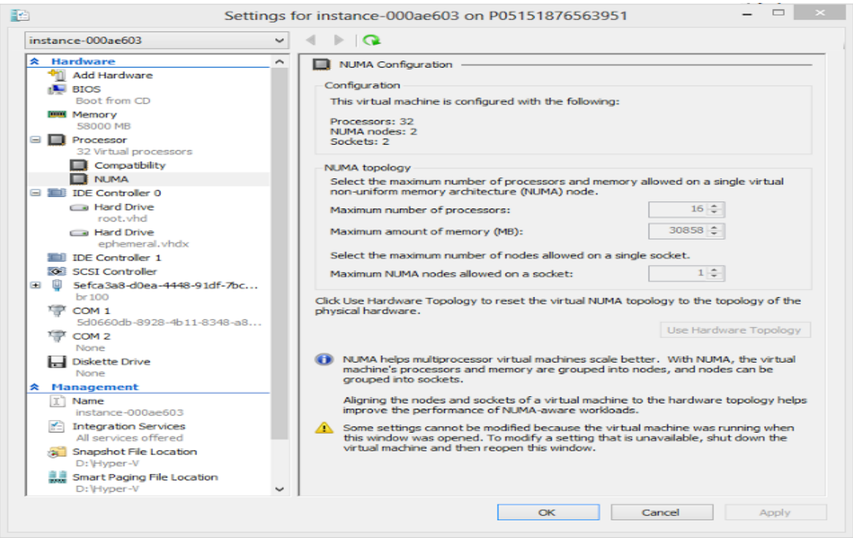
One of the things that led us to focus on KVM tuning was the comparison with Hyper-V. At CERN, we made an early decision to run a multi-hypervisor cloud building on the work by cloudbase.it (http://cloudbase.it/) and Puppet on Windows to share the deployment scripts for both CentOS and Windows hypervisors. This allows us to direct appropriate workloads to the best hypervisor for the job.

One of the tests when we saw a significant overhead on the default KVM configuration was to compare the performance overheads for a Linux configuration on Hyper-V. Interestingly, Hyper-V achieved better performance without tuning compared to the configurations with KVM. Equivalent tests on Hyper-V showed

- 4 VMs 8 cores: 0.8% overhead compared to bare metal
- 1 VM 32 cores: 3.3% overhead compared to bare metal

These performance results allowed us to focus on the potential areas for optimisation, that we needed to tune the hypervisor rather than a fundamental problem with virtualisation (with the results above for NUMA and CPU pinning)

The Hyper-V configuration pins each core to the underlying NUMA socket which is similar to how the Kilo NUMA tuning sets KVM up.



Configuration

This virtual machine is configured with the following:

Processors: 32
NUMA nodes: 2
Sockets: 2

NUMA topology

Select the maximum number of processors and memory allowed on a single virtual non-uniform memory architecture (NUMA) node.

Maximum number of processors: 16

Maximum amount of memory (MB): 30858

Select the maximum number of nodes allowed on a single socket.

Maximum NUMA nodes allowed on a socket: 1

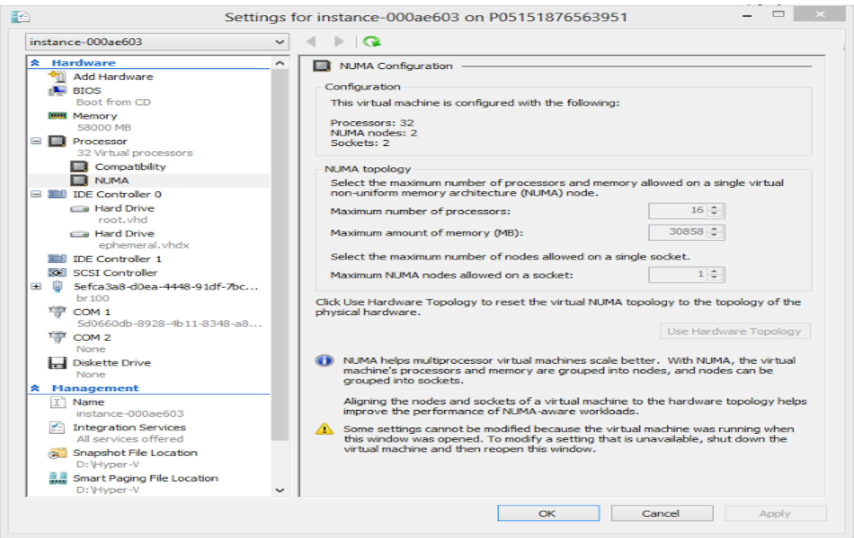
Click Use Hardware Topology to reset the virtual NUMA topology to the topology of the physical hardware.

NUMA helps multiprocessor virtual machines scale better. With NUMA, the virtual machine's processors and memory are grouped into nodes, and nodes can be grouped into sockets.

Aligning the nodes and sockets of a virtual machine to the hardware topology helps improve the performance of NUMA-aware workloads.

Some settings cannot be modified because the virtual machine was running when this window was opened. To modify a setting that is unavailable, shut down the virtual machine and then reopen this window.

(http://3.bp.blogspot.com/-j2lVBm-CYsc/VcItM_Y8EeI/AAAAAAAALj0/OjO1J11UwT0/s1600/hv001.png) and



Settings for instance-000ae603 on P05151876563951

instance-000ae603

Hardware
Add Hardware
BIOS
Boot from CD
Memory
58000 MB
Processor
32 Virtual processors
Compatibility
NUMA
IDE Controller 0
Hard Drive
root.vhd
Hard Drive
ephemeral.vhdx
IDE Controller 1
SCSI Controller
5efca3a8-d0ea-4448-91df-7bc...
br100
COM 1
5d0660db-8928-4b11-8348-a8...
COM 2
None
Diskette Drive
None
Management
Name
instance-000ae603
Integration Services
All services offered
Snapshot File Location
D:\Hyper-V
Smart Paging File Location
D:\Hyper-V

NUMA Configuration

Configuration

This virtual machine is configured with the following:

Processors: 32
NUMA nodes: 2
Sockets: 2

NUMA topology

Select the maximum number of processors and memory allowed on a single virtual non-uniform memory architecture (NUMA) node.

Maximum number of processors: 16

Maximum amount of memory (MB): 30858

Select the maximum number of nodes allowed on a single socket.

Maximum NUMA nodes allowed on a socket: 1

Click Use Hardware Topology to reset the virtual NUMA topology to the topology of the physical hardware.

NUMA helps multiprocessor virtual machines scale better. With NUMA, the virtual machine's processors and memory are grouped into nodes, and nodes can be grouped into sockets.

Aligning the nodes and sockets of a virtual machine to the hardware topology helps improve the performance of NUMA-aware workloads.

Some settings cannot be modified because the virtual machine was running when this window was opened. To modify a setting that is unavailable, shut down the virtual machine and then reopen this window.

(http://2.bp.blogspot.com/-HjxewvXb4ik/VcItTnpFPtI/AAAAAAAALj8/6leq_AWXJX0/s1600/hv001.png)

This gives the Linux guest configuration as seen from the guest running on a Hyper-V hypervisor

```
# numactl --hardware<o:p></o:p>
available: 2 nodes (0-1)<o:p></o:p>
node 0 cpus: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15<o:p></o:p>
node 0 size: 28999 MB<o:p></o:p>
node 0 free: 27902 MB<o:p></o:p>
node 1 cpus: 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31<o:p></o:p>
node 1 size: 29000 MB<o:p></o:p>
node 1 free: 28027 MB<o:p></o:p>
node distances:<o:p></o:p>
node   0   1<o:p></o:p>
  0:  10  20
  1:  20  10<o:p></o:p>
```

Thanks to the QEMU discuss mailing list and to the other team members who helped understand the issue (Sean Crosby (University of Melbourne) and Arne Wiebalck, Sebastian Bukowiec and Ulrich Schwickerath (CERN))

References

- Recent 2017 documentation is now at http://docs.openstack.org/admin-guide/compute-adv-config.html

by Tim Bell (noreply@blogger.com) at February 07, 2017 07:45 PM (http://openstack-in-production.blogspot.com/2015/08/kvm-and-hyper-v-comparison-for-high.html)

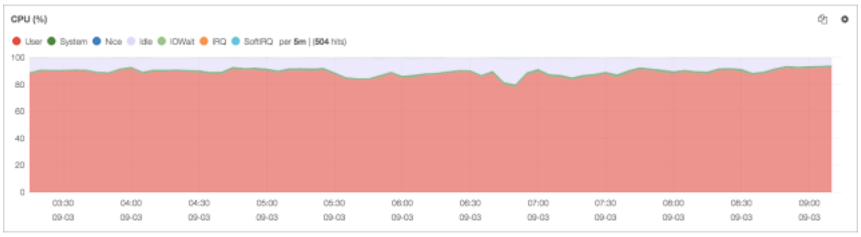
EPT, Huge Pages and Benchmarking (http://openstack-in-production.blogspot.com/2015/09/ept-huge-pages-and-benchmarking.html)

Having reported that EPT has a negative influence (http://openstack-in-production.blogspot.fr/2015/08/ept-and-ksm-for-high-throughput.html) on the High Energy Physics standard benchmark HepSpec06 (http://iopscience.iop.org/1742-6596/219/5/052009/pdf/1742-6596_219_5_052009.pdf), we have started the deployment of those settings across the CERN OpenStack cloud,

- Setting the flag in /etc/modprobe.d/kvm_intel.conf to off
- Waiting for the work on each guest to finish after stopping new VMs on the hypervisor
- Changing the flag and reloading the module
- Enabling new work for the hypervisor

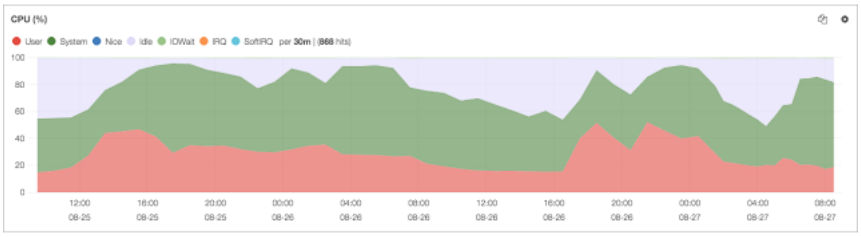
According to the HS06 tests, this should lead to a reasonable performance improvement based on the results of the benchmark and tuning. However, certain users reported significantly worse performance than previously. In particular, some workloads showed significant differences in the following before and after characteristics.

Before the workload was primarily CPU bound, spending most of its time in user space. CERN applications have to process significant amounts of data so it is not always possible to ensure 100% utilisation but the aim is to provide the workload with user space CPU.



(http://3.bp.blogspot.com/-sj8QOJji9JM/VeiN65TR6nI/AAAAAAAALok/FBbteU68mC4/s1600/with%2Bept%2Bon.png)

When EPT was turned off. some selected hypervisors showed a very difference performance profile. A major increase in non-user load and a reduction in the throughput for the experiment workloads. However, this effect was not observed on the servers with AMD processors.



(http://1.bp.blogspot.com/-wHjp7E08Gzg/VeiOlxPrfQI/AAAAAAAALos/Sissxfx2-qY/s1600/with%2Bept%2Boff.png)

With tools such as perf, we were able to trace the time down to handling the TLB misses. Perf gives

```
78.75% [kernel] [k] _raw_spin_lock<o:p></o:p>
6.76% [kernel] [k] set_spte<o:p></o:p>
1.97% [kernel] [k] memcmp<o:p></o:p>
0.58% [kernel] [k] vmx_vcpu_run<o:p></o:p>
0.46% [kernel] [k] ksm_docan<o:p></o:p>
0.44% [kernel] [k] vcpu_enter_guest<o:p></o:p>
```

The process behind the _raw_spin_lock is qemu-kvm.
<o:p></o:p>

Using systemtap kernel backtraces, we see mostly page faults and spte_* commands (shadow page table updates)
<o:p></o:p>
Both of these should not be necessary if you have hardware support for address translation: aka EPT.

There may be specific application workloads where the EPT setting was non optimal. In the worst case, the performance was several times slower. EPT/NPT increases the cost of doing page table walks when the page is not cached in the TLB. This document shows how processors can speed up page walks - http://www.cs.rochester.edu/~sandhya/csc256/seminars/vm_yuxin_yanwei.pdf (http://www.cs.rochester.edu/~sandhya/csc256/seminars/vm_yuxin_yanwei.pdf) and AMD includes a page walk cache in their processor which speeds up the walking of pages as described in this paper http://vglab.cse.iitd.ac.in/~sbansal/csl862-virt/readings/p26-bhargava.pdf (http://vglab.cse.iitd.ac.in/~sbansal/csl862-virt/readings/p26-bhargava.pdf)

In other words, EPT slows down HS06 results when there are small pages involved because the HS06 benchmarks miss the TLB a lot. NPT doesn't slow it down because AMD has a page walk cache to help speed up finding the pages when it's not in the TLB. EPT comes good again when we have large pages because it rarely results in a TLB miss. So, HS06 is probably representative of most of the job types, but the is a small share of jobs which are different and triggered the above-mentioned problem.

However, we have 6% overhead compared to previous runs due to EPT on for the benchmark as mentioned in the previous blog (http://openstack-in-production.blogspot.fr/2015/08/ept-and-ksm-for-high-throughput.html). Mitigating the EPT overheads following the comments on the previous blog (http://openstack-in-production.blogspot.fr/2015/08/ept-and-ksm-for-high-throughput.html), we looked into using dedicated Huge Pages. Our hypervisors run CentOS 7 and thus support both transparent huge pages and huge pages. Transparent huge pages performs a useful job under normal circumstances but are opportunistic in nature. They are also limited to 2MB and cannot use the 1GB maximum size.

We tried setting the default huge page to 1G using the Grub cmdline configuration.

```
$ cat /sys/kernel/mm/transparent_hugepage/enabled
[always] madvise never
$ cat /boot/grub2/grub.cfg | grep hugepage
linux16 /vmlinuz-3.10.0-229.11.1.el7.x86_64 root=UUID=7d5e2f2e-463a-4842-8e11-d6fac3568cf4 ro
rd.md.uuid=3ff29900:0eab9bfa:ea2a674d:f8b33550 rd.md.uuid=5789f86e:02137e41:05147621:b634ff66 console=tty0 nodmraid
crashkernel=auto crashkernel=auto rd.md.uuid=f6b88a6b:263fd352:c0c2d7e6:2fe442ac vconsole.font=latarcyrheb-sun16
vconsole.keymap=us LANG=en_US.UTF-8 default_hugepagesz=1G hugepagesz=1G hugepages=55 transparent_hugepage=never
$ cat /sys/module/kvm_intel/parameters/ept
Y
```

It may also be advisable to disable tuned for the moment until the bug #1189868 (https://bugzilla.redhat.com/show_bug.cgi?id=1189868)is resolved.

We also configured the XML manually to include the necessary huge pages. This will be available as a flavor or image option when we upgrade to Kilo in a few weeks.

```
  <memoryBacking>
        <hugepages>
```

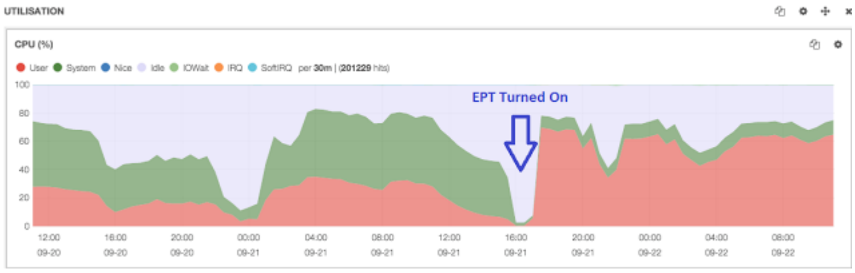
```
          <page size="1" unit="G" nodeset="0-1"/>
        </hugepages>
  </memoryBacking>
```

The hypervisor was configured with huge pages enabled. However, we saw a problem with the distribution of huge pages across the NUMA nodes.

```
$ cat /sys/devices/system/node/node*/meminfo | fgrep Huge
Node 0 AnonHugePages: 311296 kB
Node 0 HugePages_Total: 29
Node 0 HugePages_Free: 0
Node 0 HugePages_Surp: 0
Node 1 AnonHugePages: 4096 kB
Node 1 HugePages_Total: 31
Node 1 HugePages_Free: 2

Node 1 HugePages_Surp: 0
```

This shows that the pages were not evenly distributed across the NUMA nodes., which would lead to subsequent performance issues. The suspicion is that the Linux boot up sequence led to some pages being used and this made it difficult to find contiguous blocks of 1GB for the huge pages. This led us to deploy 2MB pages rather than 1GB for the moment, while may not be the optimum setting allows better optimisations than the 4K settings and still gives some potential for KSM to benefit. These changes had a positive effect as the monitoring below shows when the reduction in system time.



(http://2.bp.blogspot.com/-wIlCMUGqfVU/VgVfO-1wQMI/AAAAAAAALqQ/iHlzxdBi70g/s1600/turnonept.png)

At the OpenStack summit in Tokyo, we'll be having a session on Hypervisor Tuning so people are welcome to bring their experiences along and share the various options. Details of the session will appear at https://etherpad.openstack.org/p/TYO-ops-meetup (https://etherpad.openstack.org/p/TYO-ops-meetup).

Contributions from Ulrich Schwickerath and Arne Wiebalck (CERN) and Sean Crosby (University of Melbourne) have been included in this article along with the help of the LHC experiments to validate the configuration.

References

- OpenStack documentation now at http://docs.openstack.org/admin-guide/compute-adv-config.html (http://docs.openstack.org/admin-guide/compute-adv-config.html)
- Previous analysis for EPT at http://openstack-in-production.blogspot.fr/2015/08/ept-and-ksm-for-high-throughput.html (http://openstack-in-production.blogspot.fr/2015/08/ept-and-ksm-for-high-throughput.html)
- Red Hat blog on Huge Pages at http://redhatstackblog.redhat.com/2015/09/15/driving-in-the-fast-lane-huge-page-support-in-openstack-compute/ (http://redhatstackblog.redhat.com/2015/09/15/driving-in-the-fast-lane-huge-page-support-in-openstack-compute/)
- Mirantis blog on Huge Pages at https://www.mirantis.com/blog/mirantis-openstack-7-0-nfvi-deployment-guide-huge-pages/ (https://www.mirantis.com/blog/mirantis-openstack-7-0-nfvi-deployment-guide-huge-pages/)
- VMWare paper on EPT at https://www.vmware.com/pdf/Perf_ESX_Intel-EPT-eval.pdf (https://www.vmware.com/pdf/Perf_ESX_Intel-EPT-eval.pdf)
- Academic studies of the overheads and algorithms of EPT and NPT (AMD's technology) at http://www.cs.rochester.edu/~sandhya/csc256/seminars/vm_yuxin_yanwei.pd (http://www.cs.rochester.edu/~sandhya/csc256/seminars/vm_yuxin_yanwei.pd)f and http://vglab.cse.iitd.ac.in/~sbansal/csl862-virt/readings/p26-bhargava.pdf (http://vglab.cse.iitd.ac.in/~sbansal/csl862-virt/readings/p26-bhargava.pdf)

by Tim Bell (noreply@blogger.com) at February 07, 2017 07:44 PM (http://openstack-in-production.blogspot.com/2015/09/ept-huge-pages-and-benchmarking.html)

OpenStack CPU topology for High Throughput Computing (http://openstack-in-production.blogspot.com/2015/08/openstack-cpu-topology-for-high.html)

We are starting to look at the latest features of OpenStack Juno and Kilo as part of the CERN OpenStack cloud to optimise a number of different compute intensive applications.

We'll break down the tips and techniques into a series of small blogs. A corresponding set of changes to the upstream documentation will also be made to ensure the options are documented fully.

In the modern CPU world, a server consists of multiple levels of processing units.

- Sockets where each of the processor chips are inserted
- Cores where each processors contain multiple processing units which can run multiple processes in parallel
- Threads (if settings such as SMT (https://en.wikipedia.org/wiki/Simultaneous_multithreading) are enabled) may allow multiple processing threads to be active at the expense of sharing a core

The typical hardware used at CERN is a 2 socket system. This provides optimum price performance for our typical high throughput applications which simulate and process events from the Large Hadron Collider. The aim is not to process a single event as quickly as possible but rather to process the maximum number of events within a given time (within the total computing budget available). As the price of processors vary according to the performance, the selected systems are often not the fastest possible but the ones which give the best performance/CHF.

A typical example of this approach is in our use of SMT (https://en.wikipedia.org/wiki/Simultaneous_multithreading) which leads to a 20% increase in total throughput although each individual thread runs correspondingly slower. Thus, the typical configuration is

```
# lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                32
On-line CPU(s) list:   0-31
Thread(s) per core:    2
Core(s) per socket:    8
Socket(s):             2
NUMA node(s):          2
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 62
Model name:            Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60GHz
Stepping:              4
CPU MHz:               2999.953
BogoMIPS:              5192.93
Virtualization:        VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              20480K
NUMA node0 CPU(s):     0-7,16-23
NUMA node1 CPU(s):     8-15,24-31
```

By default in OpenStack, the virtual CPUs in a guest are allocated as standalone processors. This means that for a 32 vCPU VM, it will appear as

- 32 sockets
- 1 core per socket
- 1 thread per socket

As part of ongoing performance investigations, we wondered about the impact of this topology on CPU bound applications.

With OpenStack Juno, there is a mechanism to pass the desired topology. This can be done through flavors or image properties.

The names are slightly different between the two usages, with flavors using properties which start `hw:` and images with properties starting `hw_`.

The flavor configurations are set by the cloud administrators and the image properties can be set by the project members. The cloud administrator can also set maximum values (i.e. hw_max_cpu_cores) so that the project members cannot define values which are incompatible with the underlying resources.

```
$ openstack image set --property hw_cpu_cores=8 --property hw_cpu_threads=2 --property hw_cpu_sockets=2 0215d732-7da9-444e-a7b5-798d38c769b5
```

The VM which is booted then has this configuration reflected.

```
# lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                32
On-line CPU(s) list:   0-31
Thread(s) per core:    2
Core(s) per socket:    8
Socket(s):             2
NUMA node(s):          1
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 62
Stepping:              4
CPU MHz:               2593.748
BogoMIPS:              5187.49
Hypervisor vendor:     KVM
Virtualization type:   full
L1d cache:             32K
L1i cache:             32K
L2 cache:              4096K

NUMA node0 CPU(s):     0-31
```

While this gives the possibility to construct interesting topologies, the performance benefits are not clear. The standard High Energy Physics benchmark (http://w3.hepix.org/processors/) show no significant change. Given that there is no direct mapping between the cores in the VM and the underlying physical ones, this may be because the cores are not pinned to the corresponding sockets/cores/threads and thus Linux may be optimising for a virtual configuration rather than the real one.

This work was in collaboration with Sean Crosby (University of Melbourne) and Arne Wiebalck (CERN).

The following documentation reports have been raised

- Flavors Extra Specs - https://bugs.launchpad.net/openstack-manuals/+bug/1479270
- Image Properties - https://bugs.launchpad.net/openstack-manuals/+bug/1480519

References

- Recent 2017 documentation is now at http://docs.openstack.org/admin-guide/compute-adv-config.html

by Tim Bell (noreply@blogger.com) at February 07, 2017 07:43 PM (http://openstack-in-production.blogspot.com/2015/08/openstack-cpu-topology-for-high.html)

Ed Leafe (https://blog.leafe.com)

API Longevity (https://blog.leafe.com/api-longevity/)

How long should an API, once released, be honored? This is a topic that comes up again and again in the OpenStack world, and there are strong opinions on both sides. On one hand are the **absolutists**, who insist that once a public API is released, it **must** be supported forever. There is never *any* justification for either changing or dropping that API. On the other hand, there are **pragmatists**, who think that APIs, like all software, should evolve over time, since the original code may be buggy, or the needs of its users have changed.

I'm not at either extreme. I think the best analogy is that I believe an API is like getting married: you put a lot of thought into it before you take the plunge. You promise to stick with it forever, even when it might be easier to give up and change things. When there are rough spots (and there will be), you work to smooth them out rather than bailing out.

But there comes a time when you have to face the reality that staying in the marriage isn't really helping anyone, and that divorce is the only sane option. You don't make that decision lightly. You understand that there will be some pain involved. But you also understand that a little short-term pain is necessary for long-term happiness.

And like a divorce, an API change requires extensive notification and documentation, so that everyone understands the change that is happening. Consumers of an API should never be taken by surprise, and should have as much advance notice as possible. When done with this in mind, an API divorce does not need to be a completely unpleasant experience for anyone.

by ed at February 07, 2017 06:19 PM (https://blog.leafe.com/api-longevity/)

RDO (http://rdoproject.org/blog/)

Videos from the CentOS Dojo, Brussels, 2017 (http://rdoproject.org/blog/2017/02/centos-dojo-brussels-2017-videos/)

Last Friday in Brussels, CentOS enthusiasts gathered for the annual CentOS Dojo, right before FOSDEM (http://fosdem.org/).

While there was no official videographer for the event, I set up my video camera in the talks that I attended, and so have video of five of the sessions.

First, I attended the session covering RDO CI in the CentOS build management system. I was a little late to this talk, so it is missing the first few minutes.

<iframe allowfullscreen="" frameborder="0" height="315" src="https://www.youtube.com/embed/ZLdiPUHvKSE?list=PL27cQhFqK1Qya8JJxM9mtmZTwudi7dOmh" width="560"></iframe>
Next, I attended an introduction to Foreman, by Ewoud Kohl van Wijngaarden

<iframe allowfullscreen="" frameborder="0" height="315" src="https://www.youtube.com/embed/Zq_hT1bgfjs?list=PL27cQhFqK1Qya8JJxM9mtmZTwudi7dOmh" width="560"></iframe>
Spiros Trigazis spoke about CERN's OpenStack cloud. Unfortunately, the audio is not great in this one.

<iframe allowfullscreen="" frameborder="0" height="315" src="https://www.youtube.com/embed/fz3Xlvkf8S4?list=PL27cQhFqK1Qya8JJxM9mtmZTwudi7dOmh" width="560"></iframe>
Nicolas Planel, Sylvain Afchain and Sylvain Baubeau spoke about the Skydive network analyzer tool.

<iframe allowfullscreen="" frameborder="0" height="315" src="https://www.youtube.com/embed/4mk_ROjaVRA?list=PL27cQhFqK1Qya8JJxM9mtmZTwudi7dOmh" width="560"></iframe>
Finally, there was a demo of Cockpit - the Linux management console by Stef Walter. The lighting is a little weird in here, but you can see the screen even when you can't see Stef.

<iframe allowfullscreen="" frameborder="0" height="315" src="https://www.youtube.com/embed/9HbQ8sjolPY?list=PL27cQhFqK1Qya8JJxM9mtmZTwudi7dOmh" width="560"></iframe>

by Rich Bowen at February 07, 2017 03:00 PM (http://rdoproject.org/blog/2017/02/centos-dojo-brussels-2017-videos/)

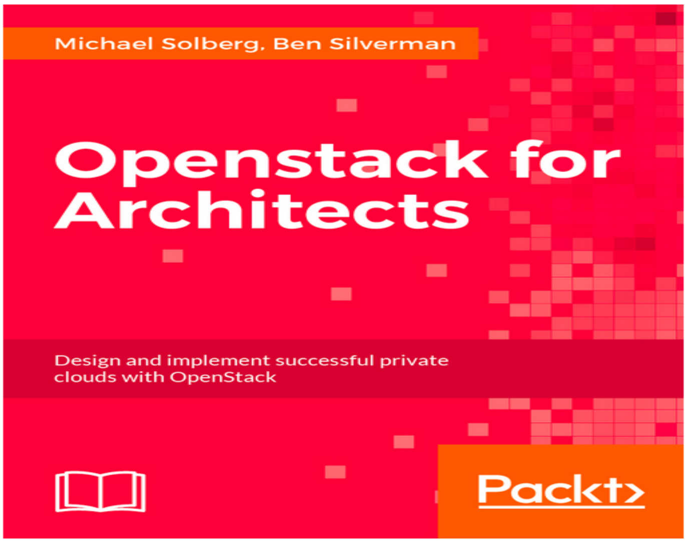
OpenStack Superuser (http://superuser.openstack.org)

How to design and implement successful private clouds with OpenStack (http://superuser.openstack.org/articles/openstack-architecture-book/)

A new book aims to help anyone who has a private cloud on the drawing board make it a reality.

Michael Solberg and Ben Silverman wrote "OpenStack for Architects," a guide to walk you through the major decision points to make effective blueprints for an OpenStack private cloud. Solberg (https://twitter.com/mpsolberg), chief architect at Red Hat, and Silverman (https://twitter.com/bensilverm) principal cloud architect for OnX Enterprise Solutions, penned the 214-page book (https://www.packtpub.com/virtualization-and-cloud/openstack-architects) available in multiple formats from Packt Publishing. (It will also be available on Amazon in March.)

Superuser talked to Solberg and Silverman about the biggest changes in private clouds, what's next and where you can find them at upcoming community events.


Michael Solberg, Ben Silverman
Openstack for Architects
Design and implement successful private clouds with OpenStack
Packt>

(https://www.packtpub.com/virtualization-and-cloud/openstack-architects)

Who will this book help most?

MS: We wrote the book for the folks who will be planning and leading the implementation of OpenStack clouds – the cloud architects. It answers a lot of the big picture questions that people have when they start designing these deployments – things like "How is this different than traditional virtualization?", "How do I choose hardware or third-party software plugins?" and "How do I integrate the cloud into my existing infrastructure?" It covers some of the nuts and bolts as well – there are plenty of code examples for unit tests and integration patterns – but it's really focused at the planning stages of cloud deployment.

What are some of the most common mistakes people make as beginners?

BS: I think that the biggest mistake people make is being overwhelmed by all of the available functionality in OpenStack and not starting with something simple. I'm pretty sure it's human nature to want to pick all the bells and whistles when they are offered, but in the case of OpenStack it can be frustrating and overwhelming. Once beginners decide what they want their cloud to look like, they tend to get confused by all of the architectural options. While there's an expectation that users should have a certain architectural familiarity with cloud concepts when working with OpenStack, learning how all of the interoperability works is still a gap for beginners. We're hoping to bridge that gap with our new book.

What are some of the most interesting use cases now?

MS: The NFV and private cloud use cases are pretty well defined at this point. We've had a couple of really neat projects lately in the genomics space where we're looking at how to best bring compute to large pools of data – I think those are really interesting. It's also possible that I just think genomics is interesting, though.

How have you seen OpenStack architecture change since you've been involved?

MS: We talk about this a bit in the book. The biggest changes happening right now are really around containers. Both the impact of containers in the tenant space and on the control plane. The changes in architecture are so large that we'll probably have to write a second edition as they get solidified over the next year or two.

Are there any new cases you're working with (and still building) that you can talk about?

BS: The idea of doing mobile-edge computing using OpenStack (http://www.redhat.com/blog/verticalindustries/a-path-to-mobile-edge-computing-with-red-hat-openstack/)as the orchestrator for infrastructure at the mobile edge is really hot right now. It is being led by the new ETSI Mobile-Edge Computing Industry Specification Group (http://www.etsi.org/technologies-clusters/technologies/mobile-edge-computing) and has the backing of about 80 companies.

Not only would this type of OpenStack deployment have to support NFV workloads over the new mobile 5G networks, but support specialized workloads that have to perform at high bandwidth and low latency geographically close to customers. We could even see open compute work into this use case as service providers try and get the most out of edge locations. It has been very cool over the last few years to have seen traditional service providers taking NFV back to the regional or national data center, but it's even cooler to see that they are now using OpenStack to put infrastructure back at the edge to extend infrastructure back out to customers.

Ben, curious about your tweet (https://twitter.com/bensilverm/status/824312281278320640) – "You must overcome the mindset that digital transformation is a tech thing, rather than an enterprise-wide commitment"…Why do people fall into this mentality? What's the best cure for it?

BS: The common misconception for a lot of enterprises is that technology transformation simply takes a transformation of the technology. Unfortunately, that's not the case when it comes to cloud technologies. Moving from legacy bare metal or virtualization platforms to a true cloud architecture won't provide much benefit unless business processes and developer culture changes to take advantage of it.

The old adage "build it and they'll come" doesn't apply to OpenStack clouds. Getting executive sponsorship and building a grassroots effort in the internal developer community goes a long way towards positive cultural change. I always advise my clients to get small wins for cloud and agile development first and use those groups as cheerleaders for their new OpenStack cloud.

I tell them, "bring others in slowly and collect small wins. If you don't go slow and get gradual adoption you'll end up with accelerated rejection and even with executive sponsorship, you could find yourself with a great platform and no tenants." I have seen this happen again and again simply because of uncontrolled adoption and the wrong workloads or the wrong team was piloted into OpenStack and had bad experiences.

Why is a book helpful now — there's IRC, mailing lists, documentation, video tutorials etc.?

MS: That was actually one of the biggest questions we had when we sat down to write the book! We've tried to create content which answers the kinds of questions that aren't easily answered through these kinds of short-form documentation. Most of the topics in the book are questions that other architects have asked us and wanted to have a verbal discussion around – either as a part of our day jobs or in the meetups.

BS: There are a lot of ways people can get OpenStack information today. Unfortunately I've found that a lot of it is poorly organized, outdated or simply, incomplete. I find Google helpful for information about OpenStack topics, but, if you type "OpenStack architecture" you end up with all sorts of results. Some of the top results are the official OpenStack documentation pages which, thanks to the openstack-manuals (http://docs.openstack.org/) teams are getting a major facelift (go team docs!). However, right below the official documentation are outdated articles and videos that are in the Cactus and Diablo timeframes (http://superuser.openstack.org/articles/navigating-openstack-community-release-cycles-events/). Not useful at all.

What's on your OpenStack bookshelf?

MS: Dan Radez's (http://superuser.openstack.org/articles/openstack-essentials-a-book-to-get-you-grounded-in-the-fundamentals/) book was one of the first physical books I had bought in a long time. I read it before I started this book to make sure we didn't cover content he had already covered there. I just finished "Common OpenStack Deployments" (https://superuser.openstack.org/articles/common_openstack_deployments_book/) as well – I think that's a great guide to creating a Puppet composition module, which is something we touch briefly on in our book.

BS: Looking at my bookshelf now I can see Shrivastwa and Sarat's "Learning OpenStack" (https://www.packtpub.com/virtualization-and-cloud/learning-openstack) (full disclosure, I was the technical reviewer), James Denton's second edition "Learning OpenStack Neutron" and an old copy of Doug Shelley and Amrith Kumar's "OpenStack Trove." (https://www.linkedin.com/in/amrith) Like Michael, I've got "Common OpenStack Deployments (http://superuser.openstack.org/articles/common_openstack_deployments_book/)" on order and I'm looking forward to reading it.

And what is the "missing manual" you'd like to see?

BS: I would love to see "Beginner's Guide to OpenStack Development and the OpenStack Python SDK(s)" I know enough Python to be dangerous and fix some low-hanging bugs, but a book that really dug into the Python libraries with examples and exercises would be pretty cool. It could even contain a getting started guide to help developers get familiar with the OpenStack development tools and procedures.

Are either of you attending the PTG and/or Boston Summit?

BS: I'll be at the PTG (https://www.openstack.org/ptg/), as a member of the openstack-manuals team, I'm looking forward to having some really productive sessions with our new project team lead (PTL) Alexandra Settle (https://www.linkedin.com/in/asettle). We've already started discussing some of our goals for Pike, so we're in good shape.

I'll also be at the Summit (https://www.openstack.org/summit) in Boston, I've submitted a talk entitled "Taking Capacity Planning to Flavor Town – Planning your flavors for maximum efficiency." My talk centers around the elimination of dead space in compute, storage and networking resources by deterministic flavor planning. Too many enterprises have weird-sized flavors all residing on the same infrastructure which leads to strange-sized orphan resources that are never consumed. On a small scale the impact is minimal but companies with thousands and tens of thousands of cores can recover hundreds of thousands of dollars in wasted CAPEX simply by planning properly.

MS: I'll be at the Boston Summit – more for catching up with my colleagues in the industry than anything else. You can find me at the Atlanta OpenStack Meetup (https://www.meetup.com/openstack-atlanta/)most months as well.

The authors are also keeping a blog (https://openstackforarchitects.com/) about the book where you can find updates on signings and giveaways.

Cover photo by: Brian Rinker
(http://superuser.openstack.org/feed/%26quot%3Bhttps%3A//www.flickr.com/photos/atin800/12211175353//%E2%80%9C)

The post How to design and implement successful private clouds with OpenStack (http://superuser.openstack.org/articles/openstack-architecture-book/) appeared first on OpenStack Superuser (http://superuser.openstack.org).

by Nicole Martinelli at February 07, 2017 12:47 PM (http://superuser.openstack.org/articles/openstack-architecture-book/)

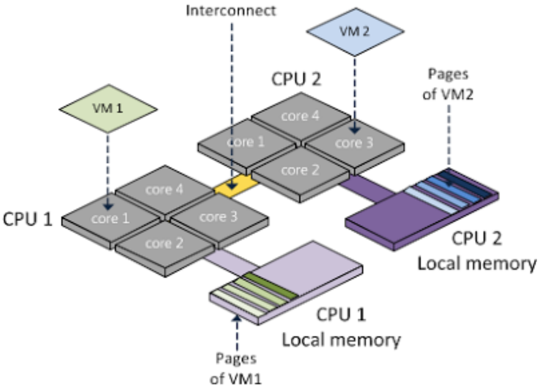
OpenStack in Production (http://openstack-in-production.blogspot.com/)

NUMA and CPU Pinning in High Throughput Computing (http://openstack-in-production.blogspot.com/2015/08/numa-and-cpu-pinning-in-high-throughput.html)

CERN's OpenStack cloud runs the Juno release on mainly CentOS 7 hypervisors.
Along with previous tuning options described in this blog which can be used on Juno, a number of further improvements have been delivered in Kilo.

Since this release will be installed at CERN during the autumn, we had to configure standalone KVM configurations to test the latest features, in particular around NUMA and CPU pinning.

NUMA (https://en.wikipedia.org/wiki/Non-uniform_memory_access) features have been appearing in more recent processors that means memory accesses are no longer uniform. Rather than a single large pool of memory accessed by the processors, the performance of the memory access varies according to whether the memory is local to the processor.



(http://4.bp.blogspot.com/-ryStuylAtk0/Vb3Wlg32pHI/AAAAAAAALi4/mbyDnaej6v8/s1600/NUMA.png) (Frank Denneman (http://frankdenneman.nl/2015/02/27/memory-deep-dive-numa-data-locality/) (https://www.blogger.com/))

A typical case above is where VM 1 is running on CPU 1 and needs a page of memory to be allocated. It is important that the memory allocated by the underlying hypervisor is the fastest access possible for the VM1 to access in future. Thus, the guest VM kernel needs to be aware of the underlying memory architecture of the hypervisor.

The NUMA configuration of a machine can be checked using lscpu. This shows two NUMA nodes on CERN's standard server configurations (two processors with 8 physical cores and SMT enabled)

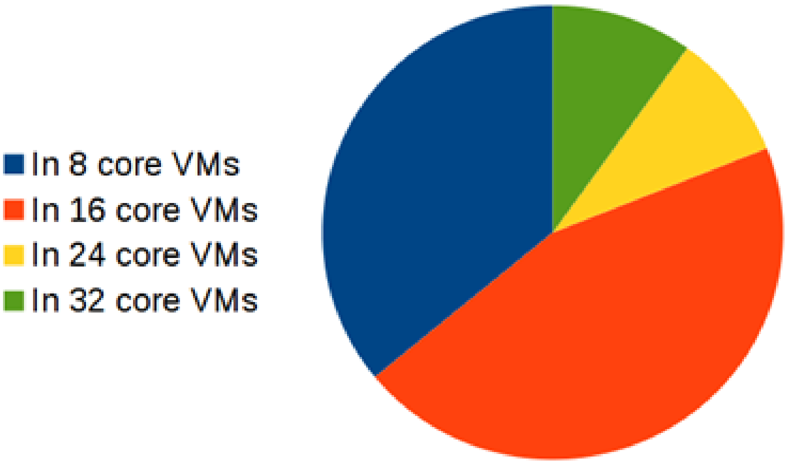
```
# lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                32
On-line CPU(s) list:   0-31
Thread(s) per core:    2
Core(s) per socket:    8
Socket(s):             2
NUMA node(s):          2
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 62
Model name:            Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60GHz
Stepping:              4
CPU MHz:               2257.632
BogoMIPS:              5206.18
Virtualization:        VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              20480K
NUMA node0 CPU(s):     0-7,16-23
NUMA node1 CPU(s):     8-15,24-31
```

Thus, cores 0-7 and 16-23 are attached to the first NUMA node with the others on the second. The two ranges come from SMT. VMs however see a single NUMA node.

```
NUMA node0 CPU(s): 0-31
```

First Approach - numad

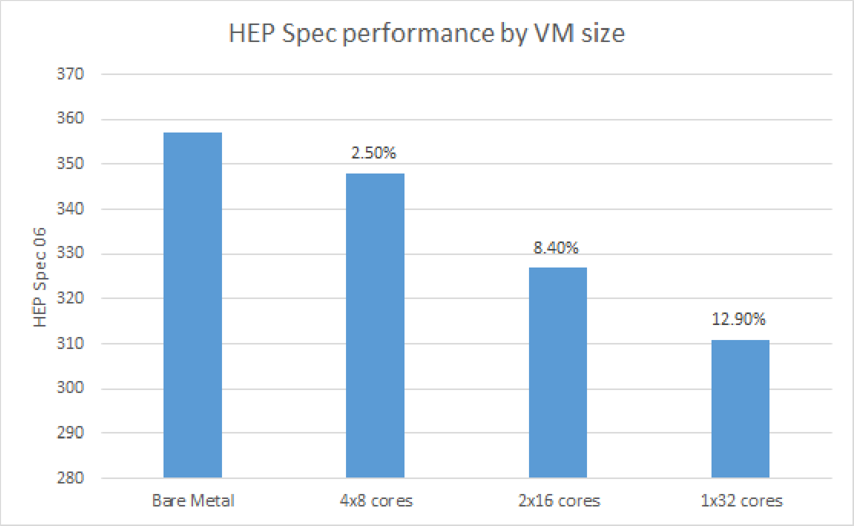
The VMs on the CERN cloud are distributed across different sizes. Since there is a mixture of VM sizes, NUMA has a correspondingly varied influence.



(http://2.bp.blogspot.com/-oLqiBX_ImXA/Vb3gTg2RTNI/AAAAAAAALjI/6Uxp6PNbXDA/s1600/vmdistribution.png)

Linux provides the numad daemon which provides some automated balancing of NUMA workloads to move memory near to the processor where the thread is running.

In the case of 8 core VMs, numad on the hypervisor provided a performance gain of 1.6%. However, the effects for larger VMs was much less significant. Looking at the performance for running 4x8 core VMs versus 1x32 core VM, there was significantly more overhead for the large VM case.



(http://2.bp.blogspot.com/-eXmMLu0x838/Vb3liHzchAI/AAAAAAAALjg/g4hMP_IL0Dc/s1600/vmoverhead.png)

Second approach - expose NUMA to guest VM

This can be done using appropriate KVM directives. With OpenStack Kilo, these will be possible via the flavors extra specs and image properties. In the meanwhile, we configured the hypervisor with the following XML for libvirt.

```
<cpu mode='host-passthrough'>
<numa>
<cell id='0' cpus='0-7' memory='16777216'/>
<cell id='1' cpus='16-23' memory='16777216'/>
<cell id='2' cpus='8-15' memory='16777216'/>
<cell id='3' cpus='24-31' memory='16777216'/>
</numa>
</cpu>
```

In an ideal world, there would be two cells defined (0-7,16-23 and 8-15,24-31) but KVM currently does not support non-contiguous ranges on CentOS 7 [1]. The guests see the configuration as follows

```
# lscpu
Architecture: x86_64
CPU op-mode(s): 32-bit, 64-bit
Byte Order: Little Endian
CPU(s): 32
On-line CPU(s) list: 0-31
Thread(s) per core: 2
Core(s) per socket: 8
Socket(s): 2
NUMA node(s): 4
Vendor ID: GenuineIntel
CPU family: 6
Model: 62
Model name: Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60GHz
Stepping: 4
CPU MHz: 2593.750
BogoMIPS: 5187.50
Hypervisor vendor: KVM
Virtualization type: full
L1d cache: 32K
L1i cache: 32K
L2 cache: 4096K
NUMA node0 CPU(s): 0-7
NUMA node1 CPU(s): 8-15
NUMA node2 CPU(s): 16-23
NUMA node3 CPU(s): 24-31
```

With this approach and turning off numad on the hypervisor, the performance of the large VM improved by 9%.

We also investigated the numatune options but these did not produce a significant improvement.

Third Approach - Pinning CPUs

From the hypervisor's perspective, the virtual machine appears as a single process which needs to be scheduled on the available CPUs. While the NUMA configuration above means that memory access from the processor will tend to be local, the hypervisor may then choose to place the next scheduled clock tick on a different processor. While this is useful in the case of hypervisor over-commit, for a CPU bound application, this leads to less memory locality.

With Kilo, it will be possible to pin a virtual core to a physical one. The same was done using the hypervisor XML as for NUMA.

```
<cputune>
<vcpupin vcpu="0" cpuset="0"/>
<vcpupin vcpu="1" cpuset="1"/>
<vcpupin vcpu="2" cpuset="2"/>
<vcpupin vcpu="3" cpuset="3"/>
<vcpupin vcpu="4" cpuset="4"/>
<vcpupin vcpu="5" cpuset="5"/>
...
```

This will mean that the virtual core #1 is always run on the physical core #1.
Repeating the large VM test provided a further 3% performance improvement.

The exact topology has been set in a simple fashion. Further investigation on getting exact mappings between thread siblings is needed to get the most of out of the tuning.

The impact on smaller VMs (8 and 16 core) is also needing to be studied. Optimising for one use case has a risk that other scenarios may be affected. Custom configurations for particular topologies of VMs increases the operations effort to run a cloud at scale. While the changes should be positive, or at minimum neutral, this needs to be verified.

Summary

Exposing the NUMA nodes and using CPU pinning has reduced the large VM overhead with KVM from 12.9% to 3.5%. When the features are available in OpenStack Kilo, these can be deployed by setting up the appropriate flavors with the additional pinning and NUMA descriptions for the different hardware types so that large VMs can be run at a much lower overhead.

This work was in collaboration with Sean Crosby (University of Melbourne) and Arne Wiebalck and Ulrich Schwickerath (CERN).

Previous blogs in this series are

- CPU topology - http://openstack-in-production.blogspot.fr/2015/08/openstack-cpu-topology-for-high.html (http://openstack-in-production.blogspot.fr/2015/08/openstack-cpu-topology-for-high.html)
- CPU model selection - http://openstack-in-production.blogspot.fr/2015/08/cpu-model-selection-for-high-throughput.html (http://openstack-in-production.blogspot.fr/2015/08/cpu-model-selection-for-high-throughput.html)
- KSM and EPT - http://openstack-in-production.blogspot.fr/2015/08/ept-and-ksm-for-high-throughput.html (http://openstack-in-production.blogspot.fr/2015/08/ept-and-ksm-for-high-throughput.html)

Updates

[1] RHEV does support this with the later QEMU rather than the default in CentOS 7 (http://cbs.centos.org/repos/virt7-kvm-common-testing/x86_64/os/Packages/, version 2.1.2)

References

- Detailed presentation on the optimisations - https://indico.cern.ch/event/384358/contributions/909247/ (https://indico.cern.ch/event/384358/contributions/909247/)
- Red Hat's tuning guide - https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Virtualization_Tuning_and_Optimization_Guide/sect-Virtualization_Tuning_Optimization_Guide-NUMA-NUMA_and_libvirt.html (https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Virtualization_Tuning_and_Optimization_Guide/sect-Virtualization_Tuning_Optimization_Guide-NUMA-NUMA_and_libvirt.html)
- Stephen Gordon's description of the Kilo features -http://redhatstackblog.redhat.com/2015/05/05/cpu-pinning-and-numa-topology-awareness-in-openstack-compute/ (http://redhatstackblog.redhat.com/2015/05/05/cpu-pinning-and-numa-topology-awareness-in-openstack-compute/)
- NUMA memory architecture - http://frankdenneman.nl/2015/02/27/memory-deep-dive-numa-data-locality/ (http://frankdenneman.nl/2015/02/27/memory-deep-dive-numa-data-locality/)
- OpenStack memory placement - https://wiki.openstack.org/wiki/VirtDriverGuestCPUMemoryPlacement (https://wiki.openstack.org/wiki/VirtDriverGuestCPUMemoryPlacement)
- Fedora work - http://bderzhavets.blogspot.fr/2015/07/cpu-pinning-and-numa-topology-on-rdo_31.html (http://bderzhavets.blogspot.fr/2015/07/cpu-pinning-and-numa-topology-on-rdo_31.html)

by Tim Bell (noreply@blogger.com) at February 07, 2017 09:43 AM (http://openstack-in-production.blogspot.com/2015/08/numa-and-cpu-pinning-in-high-throughput.html)

Christopher Smart (https://blog.christophersmart.com)

Fixing webcam flicker in Linux with udev (https://blog.christophersmart.com/2017/02/07/fixing-webcam-flicker-in-linux-with-udev/)

I recently got a new Dell XPS 13 (9360) laptop for work and it's running Fedora pretty much perfectly.

However, when I load up Cheese (or some other webcam program) the video from the webcam flickers. Given that I live in Australia, I had to change the powerline frequency from 60Hz to 50Hz to fix it.

```
 sudo dnf install v4l2-ctl
v4l2-ctl --set-ctrl power_line_frequency=1
```

I wanted this to be permanent each time I turned my machine on, so I created a udev rule to handle that.

```
 cat << EOF | sudo tee /etc/udev/rules.d/50-dell-webcam.rules
SUBSYSTEM=="video4linux", \
SUBSYSTEMS=="usb", \
ATTRS{idVendor}=="0c45", \
ATTRS{idProduct}=="670c", \
PROGRAM="/usr/bin/v4l2-ctl --set-ctrl \
power_line_frequency=1 --device /dev/%k", \
SYMLINK+="dell-webcam"
EOF
```

It’s easy to test. Just turn flicker back on, reload the rules and watch the flicker in Cheese automatically disappear 🙂

```
 v4l2-ctl --set-ctrl power_line_frequency=0
sudo udevadm control --reload-rules && sudo udevadm trigger
```

Of course I also tested with a reboot.

It’s easy to do with any webcam, just take a look on the USB bus for the vendor and product IDs. For example, here’s a Logitech C930e (which is probably the nicest webcam I’ve ever used, and also works perfectly under Fedora).

```
 Bus 001 Device 022: ID 046d:0843 Logitech, Inc. Webcam C930e
```

So you would replace the following in your udev rule:

- ATTRS{idVendor}==**“046d”**
- ATTRS{idProduct}==**“0843”**
- SYMLINK+=**“c930e”**

Note that SYMLINK is not necessary, it just creates an extra /dev entry, such as /dev/c930e, which is useful if you have multiple webcams.

by Chris at February 07, 2017 06:56 AM (https://blog.christophersmart.com/2017/02/07/fixing-webcam-flicker-in-linux-with-udev/)

February 06, 2017

Cloudwatt (https://dev.cloudwatt.com/en/blog/index.html)

5 Minutes Stacks, episode 52 : iceHRM (https://dev.cloudwatt.com/en/blog/5-minutes-stacks-episode-fifty-two-icehrm.html)

Episode 52 : iceHRM

iceHRM is a Human Resources Management tool allowing to manage a company ant its employees. It is possible to add their personnal information, to create plannings, payslips and to set up some projects. The interface is really intuitive.

iceHRM is developed in PHP and uses a MariaDB database to save all the data it needs.

This episode will help you to deploy iceHRM with a High Availability (HA) on a cluster of two instances behind a load-balancer, each instance being mutually replicated in real time.

Preparations

The Versions

- Ubuntu 16.04
- Apache 2.4.18
- MariaDB Galera Cluster 5.5.53
- GlusterFS 3.7.6
- iceHRM 18.0.OS

The prerequisites to deploy this stack

These should be routine by now:

- An Internet access
- A Linux shell
- A Cloudwatt account (https://www.cloudwatt.com/cockpit/#/create-contact) with a valid keypair (https://console.cloudwatt.com/project/access_and_security/?tab=access_security_tabs__keypairs_tab)
- The tools of the trade: OpenStack CLI (http://docs.openstack.org/cli-reference/content/install_clients.html)
- A local clone of the Cloudwatt applications (https://github.com/cloudwatt/applications) git repository (if you are creating your stack from a shell)

Size of the instance

By default, the stack deploys two instances of type “Standard 1” (n1.cw.standard-1). A variety of other instance types exist to suit your various needs, allowing you to pay only for the services you need. Instances are charged by the minute and capped at their monthly price (you can find more details on the Pricing page (https://www.cloudwatt.com/en/pricing.html) on the Cloudwatt website).

Stack parameters, of course, are yours to tweak at your fancy.

By the way…

If you do not like command lines, you can go directly to the “run it thru the console” section by clicking here (https://dev.cloudwatt.com/en/blog/rss.xml#console)

What will you find in the repository

Once you have cloned the github, you will find in the `bundle-xenial-icehrm/` repository:

- `bundle-xenial-icehrm.heat.yml` : HEAT orchestration template. It will be use to deploy the necessary infrastructure.
- `stack-start.sh` : Stack launching script. This is a small script that will save you some copy-paste.

- `stack-get-url.sh` : Flotting IP recovery script.

Start-up

Initialize the environment

Have your Cloudwatt credentials in hand and click HERE (https://console.cloudwatt.com/project/access_and_security/api_access/openrc/). If you are not logged in yet, you will go thru the authentication screen then the script download will start. Thanks to it, you will be able to initiate the shell accesses towards the Cloudwatt APIs.

Source the downloaded file in your shell. Your password will be requested.

```
$ source COMPUTE-[...]-openrc.sh
Please enter your OpenStack Password:

```

Once this done, the Openstack command line tools can interact with your Cloudwatt user account.

Adjust the parameters

With the `bundle-xenial-icehrm.heat.yml` file, you will find at the top a section named `parameters` . The parameters to adjust are `keypair_name` and `sqlpass` . Their `default` value must contain respectively a valid keypair with regards to your Cloudwatt user account and the password you want to use to the iceHRM database. This is within this same file that you can adjust the instance size by playing with the `flavor` parameter.

```
heat_template_version: 2013-05-23


description: All-in-one iceHRM stack


parameters:
  keypair_name:
    description: Keypair to inject in instance
    label: SSH Keypair
    type: string

  flavor_name:
    default: n1.cw.standard-1
    description: Flavor to use for the deployed instance
    type: string
    label: Instance Type (Flavor)
    constraints:
      - allowed_values:
          - n1.cw.standard-1
          - n1.cw.standard-2
          - n1.cw.standard-4
          - n1.cw.standard-8
          - n1.cw.standard-12
          - n1.cw.standard-16

  volume_attachment:
    description: Attacher un volume cinder de 20GO ?
    default: 0
    type: string
[...]
```

Start stack

In a shell, run the script `stack-start.sh` with his name in parameter:

```
 ./stack-start.sh iceHRM
+--------------------------------------+-----------------------+--------------------+----------------------+
| id                                   | stack_name            | stack_status       | creation_time        |
+--------------------------------------+-----------------------+--------------------+----------------------+
| 4785c76e-3681-4b02-8a91-a7a3cc4a6440 | iceHRM                | CREATE_IN_PROGRESS | 2016-12-21T13:53:56Z |
+--------------------------------------+-----------------------+--------------------+----------------------+
```

Within **5 minutes** the stack will be fully operational. (Use `watch` to see the status in real-time)

```
 $ watch heat resource-list iceHRM
+-----------------------+------------------------------------+-----------------+----------------------+
| resource_name         | resource_type                      | resource_status | updated_time         |
+-----------------------+------------------------------------+-----------------+----------------------+
| inst1                 | OS::Nova::Server                   | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| inst1_cinder          | OS::Heat::ResourceGroup            | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| inst1_port            | OS::Neutron::Port                  | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| inst2                 | OS::Nova::Server                   | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| inst2_cinder          | OS::Heat::ResourceGroup            | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| inst2_port            | OS::Neutron::Port                  | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| lbaas                 | OS::Neutron::LoadBalancer          | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| lbaas_pool            | OS::Neutron::Pool                  | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| lbaas_pool_vip        | OS::Neutron::FloatingIPAssociation | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| lbaas_vip_floating_ip | OS::Neutron::FloatingIP            | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| lbaas_vip_port        | OS::Neutron::Port                  | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| network               | OS::Neutron::Net                   | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| security_group        | OS::Neutron::SecurityGroup         | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
| subnet                | OS::Neutron::Subnet                | CREATE_COMPLETE | 2016-12-21T13:53:57Z |
+-----------------------+------------------------------------+-----------------+----------------------+
```

The `start-stack.sh` script takes care of running the API necessary requests to execute the normal heat template which:

- Starts two Ubuntu Xenial based instances preprovisionned with the iceHRM stack
- Expose it on the Internet via a floating IP

All of this is fine, but…

You do not have a way to create the stack from the console?

We do indeed! Using the console, you can deploy iceHRM:

1. Go the Cloudwatt Github in the applications/bundle-xenial-icehrm (https://github.com/cloudwatt/applications/tree/master/bundle-xenial-icehrm) repository
2. Click on the file named `bundle-xenial-icehrm.heat.yml`
3. Click on RAW, a web page will appear containing purely the template
4. Save the file to your PC. You can use the default name proposed by your browser (just remove the .txt)
5. Go to the « Stacks (https://console.cloudwatt.com/project/stacks/) » section of the console
6. Click on « Launch stack », then « Template file » and select the file you just saved to your PC, and finally click on « NEXT »
7. Name your stack in the « Stack name » field
8. Enter the name of your keypair in the « SSH Keypair » field
9. Write a passphrase that will be used for the database icehrmuser
10. Choose your instance size using the « Instance Type » dropdown and click on « LAUNCH »

The stack will be automatically generated (you can see its progress by clicking on its name). When all modules become green, the creation will be complete. You can then go to the "Instances" menu to find the floating IP, or simply refresh the current page and check the Overview tab for a handy link.

If you've reached this point, you're already done! Go enjoy iceHRM!

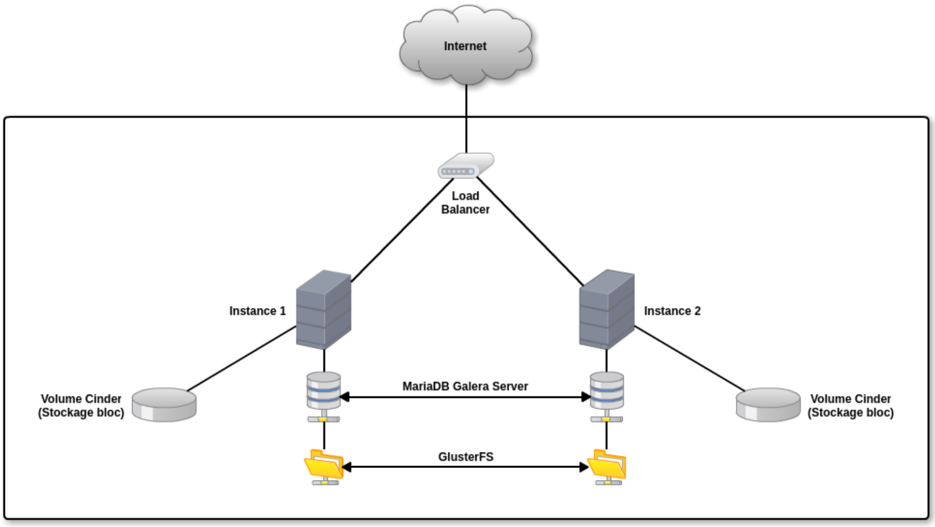
A one-click deployment sounds really nice…

… Good! Go to the Apps page (https://www.cloudwatt.com/en/apps/) on the Cloudwatt website, choose the apps, press **DEPLOY** and follow the simple steps… 2 minutes later, a green button appears… **ACCESS**: you have iceHRM.

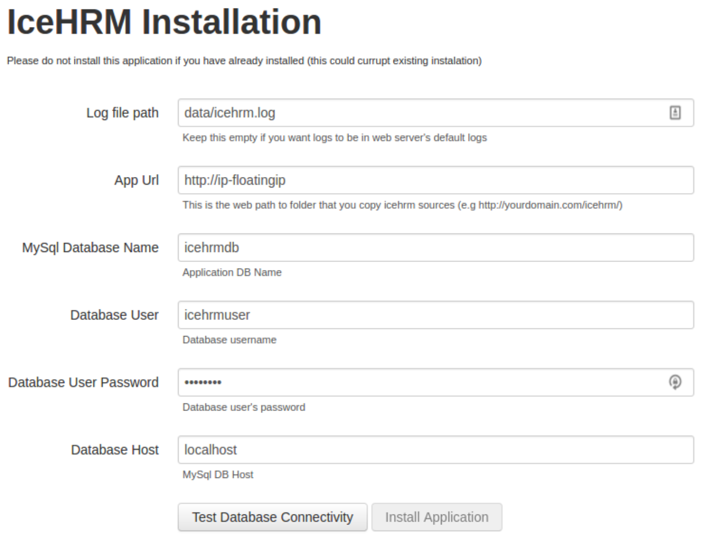
Enjoy

You are now in possession of iceHRM, you can enter via the URL `http://ip-floatingip`. Your full URL will be present in your stack overview in horizon Cloudwatt console.

The stack is composed like this:



At your first connexion you will ask to give the information about how to access to the database. Complete the fields as below, the password is which one you chose when you created the stack.



IceHRM Installation

Please do not install this application if you have already installed (this could currupt existing instalation)

Log file path: data/icehrm.log
Keep this empty if you want logs to be in web server's default logs

App Url: http://ip-floatingip
This is the web path to folder that you copy icehrm sources (e.g http://yourdomain.com/icehrm/)

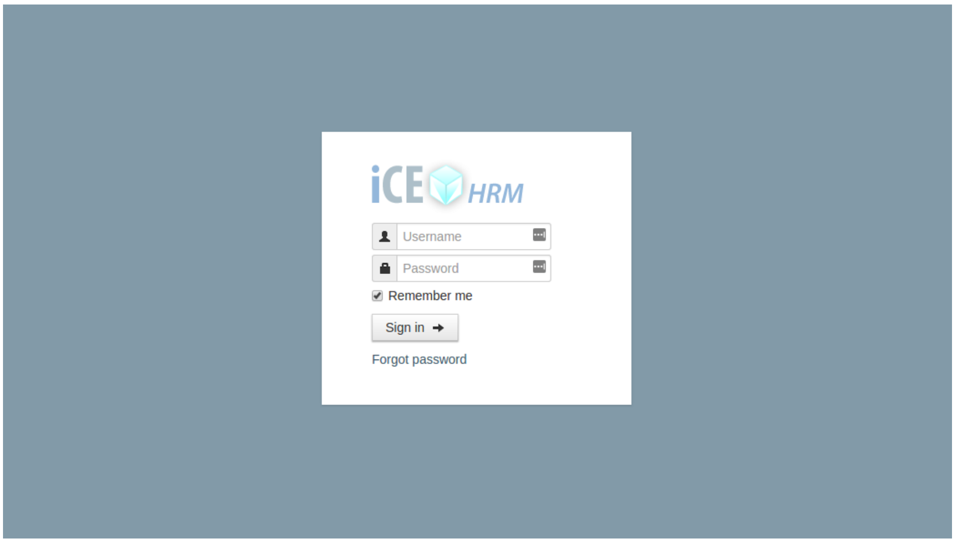
MySql Database Name: icehrmdb
Application DB Name

Database User: icehrmuser
Database username

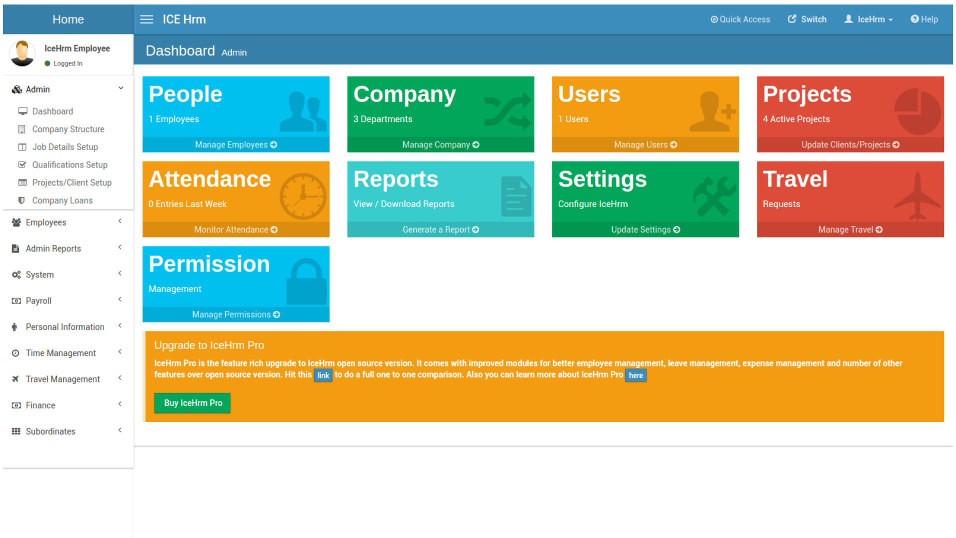
Database User Password: ••••••••
Database user's password

Database Host: localhost
MySql DB Host

The default username and password to connect to iceHRM are `admin` .



You can now discover the iceHRM’s interface:



Dashboard Admin

People
1 Employees
Manage Employees

Company
3 Departments
Manage Company

Users
1 Users
Manage Users

Projects
4 Active Projects
Update Clients/Projects

Attendance
0 Entries Last Week
Monitor Attendance

Reports
View / Download Reports
Generate a Report

Settings
Configure IceHrm
Update Settings

Travel
Requests
Manage Travel

Permission
Management
Manage Permissions

Upgrade to IceHrm Pro

IceHrm Pro is the feature rich upgrade to IceHrm open source version. It comes with improved modules for better employee management, leave management, expense management and number of other features over open source version. Hit this link to do a full one to one comparison. Also you can learn more about IceHrm Pro here

You can now setup your Human Resources Management tool, this one being hosted in France in a safe environment, you can completely trust on this product.

So watt?

The goal of this tutorial is to accelerate your start. At this point **you** are the master of the stack.

Some useful links:

- Home iceHRM (https://www.icehrm.com/)
- Help iceHRM (http://blog.icehrm.com/docs/home/)

Have fun. Hack in peace.

by Simon Decaestecker at February 06, 2017 11:00 PM (https://dev.cloudwatt.com/en/blog/5-minutes-stacks-episode-fifty-two-icehrm.html)

OpenStack Superuser (http://superuser.openstack.org)

Labs-based training and performance-based certifications are keys to solving IT training needs (http://superuser.openstack.org/articles/training-performance-based-certifications/)

Michael Apostol's assessment of IT hiring trends was spot on. In his recent interview on Superuser TV (http://superuser.openstack.org/articles/get-job-working-openstack/), the software engineering director at the OpenStack Innovation Center (https://osic.org/) highlighted a key shift in the skills needed in today's IT environments: whereas not long ago, IT departments sought specialists, today's corporations seek IT pros that are both specialists and generalists.

"If I look back to [previous hiring practices], what I was looking for… [was] very much a specialist…in languages, domains, apps, client…. I found that [approach] to be very successful, and I think that will continue to be very successful," Apostol (https://www.linkedin.com/in/apmichael/) commented on past trends. "Now what I'm really looking for [in IT candidates] is generalists, somebody who can come in and be a specialist and do very well, but who also has that ability—that emotional intelligence—to move on and be able to do other technologies."

Sounds like a job for an IT Superhero, you say? Perhaps. But Linux Academy (https://linuxacademy.com/)—a quickly-growing company out of Dallas-Ft. Worth—is helping individuals and corporations meet this dichotomous demand. "It's not enough anymore to have just technical knowledge and skills; you must effectively apply skills in your IT area and apply them in the context of a much larger, much more complex, continuously changing computing environment," commented Anthony James (https://www.linkedin.com/in/anthonydjames), founder and CEO of Linux Academy, which is known for its hands-on labs and in-depth training on Linux and cloud technologies.

This expectation is true even for new college graduates: "They come out and are awesome at writing software…," observed Apostol. "They're picking up Python, Linux, and those things well, but what they're lacking is a lot of the operational knowledge. They need to know how do I use OpenStack, what is it like to be an operator…."

Linux Academy has distinguished itself by combining the technical depth of their full-time subject matter experts with a hands-on learning environment and a focus on applied knowledge: "Our hands-on labs are key in training students to apply knowledge and skills," James stated.

At Linux Academy, every student has access to up to six personal servers to use in completing exercises and practicing skills. Additionally, Linux Academy features "LiveLabs!" which are pre-built lab environments that enable students to practice skills without having to spin up an environment before they can work on the task at hand. The hands-on labs and personal server space are what help bridge the expert-generalist gap: "The labs are where students see how pieces of technology intersect in real-world environments," Justin Overstreet (https://www.linkedin.com/in/justin-overstreet-07a6b146), Linux Academy's VP of business development, stated.

"In large OpenStack corporations, you may have one IT pro using Swift for object storage while another specializes in Ceph for block storage, with neither familiar with Nova and compute services," stated Amy Marrich (https://www.linkedin.com/in/amy-marrich-57917a1/), one of two OpenStack instructors at Linux Academy. "These experts could take courses at Linux Academy on Chef, Python, Ansible, or Puppet, and get hands-on training that expands potential opportunities. "And at smaller OpenStack companies," she continued, "having broader skills can aid in deployments and troubleshooting to help reduce overall costs."

Certifications like the Certified OpenStack Administrator (COA) are important as well, according to Apostol. "As a hiring manager, I love being able to measure, and [an applicant's] certification tells me something: they went out of their way, they spent a little money, they got the training and they proved that they could pass the test." The Certified OpenStack Administrator course is one of three OpenStack certification prep courses the Linux Academy offers. Across all content areas— Linux, AWS, OpenStack, DevOps, Azure, Big Data, and Containers—Linux Academy offers 15 certification prep courses.

Anthony James holds an impressive nine certifications, although his philosophy about IT training stands out even more: "Linux and cloud training should not just be about passing the certification exam. Our goal is to help you apply knowledge and skills in real-world environments. If you can demonstrate your knowledge and skills in our hands-on labs and exercises, then you should be able to pass certification exams."

IT training demands solutions that evolve with business needs. "We're piloting something today that works with bringing new college graduates to experts in upstream contributions, and it's been hard to figure out what to teach them," said Apostol. "We'd really like to see the trend of certifications and advanced certifications continue."

Linux Academy expects to announce a new partnership during the first fiscal quarter that brings a new solution to training and certification needs. When asked for preliminary details on the upcoming release, James only comment was, "labs-based learning and performance-based certification are the keys."

Stephen Smith is the vice president of content at Linux Academy.

Superuser is always interested in community content, email: editor@superuser.org (mailto:editor@superuser.org).

The post Labs-based training and performance-based certifications are keys to solving IT training needs (http://superuser.openstack.org/articles/training-performance-based-certifications/) appeared first on OpenStack Superuser (http://superuser.openstack.org).

by Stephen Smith at February 06, 2017 04:03 PM (http://superuser.openstack.org/articles/training-performance-based-certifications/)

Hugh Blemings (http://hugh.blemings.id.au)

Lwood-20170205 (http://hugh.blemings.id.au/2017/02/06/lwood-20170205/)

Introduction

Welcome to Last week on OpenStack Dev ("Lwood") for the week just past. For more background on Lwood, please refer here (http://hugh.blemings.id.au/openstack/lwood/).

Basic Stats for the week 30 January to 5 February for openstack-dev:

- ~474 Messages (down about 17% relative to the long term average)
- ~163 Unique threads (down about 9% relative to the long term average)

A little little quieter than last week but overall back into "normal" territory for list traffic volume. That said, not a lot of traffic that met the Lwood criteria this week – one of (from memory) only two or three weeks where this has occurred. Was it something I said ? :)

Notable Discussions – openstack-dev

Survey on Open Source Collaboration practices

A reminder that Allison Randal (http://lists.openstack.org/pipermail/openstack-dev/2017-January/110663.html) is running a brief survey on collaboration practices in Open Source development, particularly within corporations and OpenStack as part of academic research she's undertaking. This was flagged last week, but if you haven't already and you can spare a few minutes to answer the brief (ten question) survey it would be much appreciated. THe Survey is here (https://www.surveymonkey.com/r/os-participation).

End of Week Wrap-ups

A few more projects starting to produce "regular" weekly updates and/or meeting summaries. This week we have ones for Horizon (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111567.html) from Richard Jones, Ironic (http://lists.openstack.org/pipermail/openstack-dev/2017-January/110227.html) courtesy of Ruby Loo, TripleO CI (http://lists.openstack.org/pipermail/openstack-dev/2017-January/111197.html) from Attila Darazs and Triple O Containers (http://lists.openstack.org/pipermail/openstack-dev/2017-January/111262.html) from Flavio Percoco.

Notable Discussions – other OpenStack lists

No traffic particularly stood out on the other lists this last week.

People and Projects

PTL nominations & Changes

As Kendall Nelson points out in his email (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111410.html) elections are underway and will end on February 7 2017 at 23:45 UTC. Ironic, Keystone, Neutron, Quality Assurance and Stable Branch Maintenance are the five projects that have multiple candidates and so go to the vote.

If you do wish to review the Candidacy Statements, they are listed on the OpenStack Governance (https://governance.openstack.org/election/#pike-ptl-candidates) page. If you're eligible to vote you should have already received an email from the Condorcet system to allow you to do so.

Core nominations & changes

- [Craton] Proposing Harry Harrington (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111341.html) for Craton core – Sulochan Acharya
- [Keystone] removing Guang Yee (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111443.html) (gyee) from keystone-core – Steve Martinelli
- [Meteos-UI] Add initial member – Hiroyuki Eguchi (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111319.html)
- [Openstack-ansible] Propose Marc Gariepy (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111585.html) as core reviewer – Jesse Pretorius
- [TripleO] Proposing Dmitry Tantsur and Alex Schultz (http://lists.openstack.org/pipermail/openstack-dev/2017-January/111260.html) as instack-undercloud cores – Ben Nemec

Miscellanea

Further reading

Don't forget these excellent sources of OpenStack news – most recent ones linked in each case

- What's Up, Doc? (http://lists.openstack.org/pipermail/openstack-dev/2017-January/111085.html) by Lana Brindley
- What's Happening in OpenStack Ansible (WHOA) (https://major.io/2016/09/30/whats-happening-in-openstack-ansible-whoa-september-2016/) by Major Hayden
- API Working Group newsletter (http://lists.openstack.org/pipermail/openstack-dev/2017-February/111491.html) – Chris Dent and the API WG
- OpenStack Developer Mailing List Digest (http://lists.openstack.org/pipermail/openstack-dev/2017-January/111139.html) by Mike Perez & Kendall Nelson
- OpenStack news over on opensource.com (https://opensource.com/article/17/2/openstack-news-february-6) by Jason Baker
- OpenStack Foundation Events Page (https://www.openstack.org/community/events/) for a frequently updated list of events

Credits

Finally got speakers set up (bliss) and so very much enjoyed listening to The Lachy Doley Group's (http://www.lachydoley.com/) new album, Lovelight while preparing Lwood this week.

Last but by no means least, thanks, as always, to Rackspace (http://rackspace.com) :)

by hugh at February 06, 2017 10:38 AM (http://hugh.blemings.id.au/2017/02/06/lwood-20170205/)

Mirantis (https://www.mirantis.com)

Mirantis OpenStack 9.2 Boosts NFV Performance and Streamlines Operations (https://www.mirantis.com/blog/mirantis-openstack-9-2-boosts-nfv-streamlines-operations/)

The post Mirantis OpenStack 9.2 Boosts NFV Performance and Streamlines Operations (https://www.mirantis.com/blog/mirantis-openstack-9-2-boosts-nfv-streamlines-operations/) appeared first on Mirantis | Pure Play Open Cloud (https://www.mirantis.com).

Many of Mirantis' largest customers are telcos using OpenStack for NFV, and in the new Mirantis OpenStack 9.2 release (https://www.mirantis.com/software/openstack/download/), we build upon the NFV features introduced in version 9.0 to better support high-performance and large-scale VNF deployments. MOS 9.2 also adds important new features to streamline operations, including improved lifecycle management and an enhanced update mechanism that more gracefully handles customizations, patches and plugins.

Emphasis on DPDK

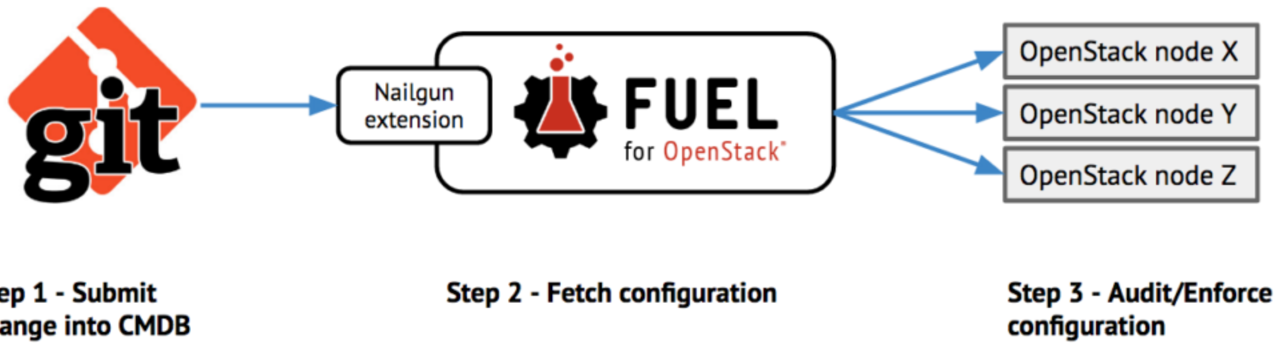
Most of the NFV improvements in Mirantis OpenStack 9.2 focus on optimizing performance and security of Open vSwitch with the Data Plane Development Kit (OVS DPDK). Cloud users running DPDK applications, such as for packet forwarding or processing, load balancing, and so on, on a VM that also utilizes OVS DPDK at the host OS can now achieve near wire speed performance thanks to efforts to harden OVS DPDK for the DPDK VNF use case. This includes NIC multiqueue support for improved network performance when scaling VNF workloads with additional vCPUs. VXLAN based segmentation and OVS based security groups enable high-performance tenant networking that is scalable and secure. IPv6 support has also been added for tenant networks that use the latest Internet Protocol version with DPDK.

Besides OVS DPDK related improvements, MOS 9.2 also updates the Linux kernel to version 4.4 as part of the included Ubuntu 14.04 LTS release so that cloud owners can deploy MOS with the latest servers to meet their capacity needs. Updating to Linux kernel 4.4 can also potentially improve OVS and KVM performance.

A Pipeline to Improved Operations

Another big change in the Mirantis OpenStack 9.2 release is a new system of lifecycle management, based on a Git repository. Using Git enables infrastructure operators to manage all the configuration files for cloud infrastructure from a single, centralized system with precise version control, thus providing auditability of cloud configuration changes.

Based on principles of Infrastructure as Code, which favors continuous small changes over occasional batch updates, this new approach greatly streamlines Day 2 (post-deployment) operations. Fuel orchestrator capabilities continually push freshly committed configuration and/or package files from the Git repository through a deployment pipeline to nodes that satisfy predefined update criteria. As a result, small, frequent updates are quickly deployed into a cloud. This can be done automatically through scheduled maintenance jobs. Rollbacks to previous configurations are supported too. In particular, updating configurations on a global or role-based level is much easier with Git-based lifecycle management.


git
Nailgun extension
FUEL for OpenStack
OpenStack node X
OpenStack node Y
OpenStack node Z
Step 1 - Submit change into CMDB
Step 2 - Fetch configuration
Step 3 - Audit/Enforce configuration

To improve scalability in Day1 and Day 2 operations, MOS 9.2 also supports the decomposition of the Fuel controller role by moving some critical services – namely Neutron, Keystone, MySQL and RabbitMQ – onto separate nodes. Spreading them onto separate nodes helps to prevent bottlenecks in huge deployments for better performance and reliability.

Besides new features related to Git-based lifecycle management, MOS 9.2 has also made updates faster and easier. Infrastructure operators with clouds running MOS 9.0 or 9.1 can update to 9.2 and retain previously installed customizations and patches, as well as plugins for Contrail, StackLight or Murano. Eliminating the need to re-install these items helps to minimize downtime during updates. Additionally, the 9.2 update will uniformly update any previously deployed Ceph nodes.

Software Updates

Last but not least, Mirantis OpenStack 9.2 includes several software updates worth mentioning:

- RabbitMQ 3.6.6
- Linux kernel 4.4
- OVS 2.6.1
- QEMU-KVM 2.5
- DPDK 16.07

Ready to update to MOS 9.2? Read the instructions here (https://docs.mirantis.com/openstack/fuel/fuel-9.2/release-notes/update-product.html) and reach out to our support team for assistance.

Previous installation of MOS 9.0 or 9.1 is required, and you can get them here (https://www.mirantis.com/software/openstack/download/).

The post Mirantis OpenStack 9.2 Boosts NFV Performance and Streamlines Operations (https://www.mirantis.com/blog/mirantis-openstack-9-2-boosts-nfv-streamlines-operations/) appeared first on Mirantis | Pure Play Open Cloud (https://www.mirantis.com).

by Michelle Yakura at February 06, 2017 10:02 AM (https://www.mirantis.com/blog/mirantis-openstack-9-2-boosts-nfv-streamlines-operations/)

Opensource.com (https://opensource.com/taxonomy/term/5126/feed/feed)

Big tent projects, an introduction to Heat, and more OpenStack news (https://opensource.com/article/17/2/openstack-news-february-6)

Explore what's happening this week in OpenStack, the open source cloud computing project.

by Jason Baker at February 06, 2017 06:00 AM (https://opensource.com/article/17/2/openstack-news-february-6)

February 03, 2017

OpenStack Superuser (http://superuser.openstack.org)

Create an Open vSwitch STT tunnel between KVM and Hyper-V-hosted VMs (http://superuser.openstack.org/articles/create-an-open-vswitch-stt-tunnel-between-kvm-and-hyper-v-hosted-vms/)

In this post we will explain how to configure manually a Open vSwitch STT tunnel between VMs running on Hyper-V and KVM hosts.

KVM OVS configuration

In this example, KVM1 provides a **STT** tunnel with local endpoint 14.14.14.1:

- *stt-1* connected to *Hyper-V* (14.14.14.2) through br-eth3

```
ubuntu@ubuntu:~$ sudo ovs-vsctl show
82585eef-349c-4573-8d77-91f9602bb535
    Bridge br-int
        fail_mode: secure
        Port br-int
            Interface br-int
                type: internal
        Port "vm1"
            Interface "vm1"
                type: internal
        Port patch-tun
            Interface patch-tun
                type: patch
                options: {peer=patch-int}
    Bridge "br-eth3"
        Port "eth3"
            Interface "eth3"
        Port "br-eth3"
            Interface "br-eth3"
                type: internal
    Bridge br-tun
        fail_mode: secure
        Port "stt-1"
            Interface "stt-1"
                type: stt
                options: {in_key=flow, local_ip="14.14.14.1", out_key=flow, remote_ip="14.14.14.2"}
        Port patch-int
            Interface patch-int
                type: patch
                options: {peer=patch-tun}
        Port br-tun
            Interface br-tun
                type: internal
ububtu@ubuntu:~$ ifconfig eth3
eth3      Link encap:Ethernet  HWaddr 00:0c:29:25:db:8c
          inet6 addr: fe80::20c:29ff:fe25:db8c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:40051 errors:0 dropped:0 overruns:0 frame:0
          TX packets:51087 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6907123 (6.9 MB)  TX bytes:81805610 (81.8 MB)
ubuntu@ubuntu:~$ ifconfig br-eth3
br-eth3   Link encap:Ethernet  HWaddr 00:0c:29:25:db:8c
          inet addr:14.14.14.1  Bcast:14.14.14.255  Mask:255.255.255.0
          inet6 addr: fe80::d413:1fff:fe62:cdd8/64 Scope:Link
          UP BROADCAST RUNNING  MTU:1500  Metric:1
          RX packets:1377 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1573 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:315330 (315.3 KB)  TX bytes:283030 (283.0 KB)
ubuntu@@ubuntu:~$ ifconfig vm1
vm1       Link encap:Ethernet  HWaddr 6a:d6:1b:77:2d:95
          inet addr:10.0.0.1  Bcast:10.0.0.255  Mask:255.255.255.0
          inet6 addr: fe80::68d6:1bff:fe77:2d95/64 Scope:Link
          UP BROADCAST RUNNING  MTU:1420  Metric:1
          RX packets:506 errors:0 dropped:0 overruns:0 frame:0
          TX packets:768 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:39788 (39.7 KB)  TX bytes:61932 (61.9 KB)
```

Please note the MTU value on vm1 is set to 1420.

Hyper-V OVS configuration

Let us assume that you have a Hyper-V Virtual Switch of type external bound to the interface ***port1*** called *vSwitch*.

The following commands will: create an IP-able device called br-port1, add the physical NIC to the bridge called br-port1, enable the device named br-port1, set the IP 14.14.14.2 to br-port1, add a bridge br-int in which we shall add the VMs later on, and will create another bridge with the tunneling information on the port ***stt-1.***

```
ovs-vsctl.exe add-br br-port1
ovs-vsctl.exe add-port br-port1 port1
Enable-NetAdapter br-port1
New-NetIpAddress -IpAddress 14.14.14.2 -PrefixLength 24 -InterfaceAlias br-port1
ovs-vsctl.exe add-br br-int
ovs-vsctl.exe add-port br-int patch-tun -- set interface patch-tun type=patch options:peer=patch-int
ovs-vsctl.exe add-br br-tun
ovs-vsctl.exe add-port br-tun patch-int -- set interface patch-int type=patch options:peer=patch-tun
ovs-vsctl.exe add-port br-tun stt-1 -- set interface stt-1 type=stt options:local_ip=14.14.14.2 options:remote_ip=14.14.14.1 options:in_key=flow options:out_key=flow
```

As you can see, all the commands are very familiar if you are used to OVS on Linux.

As introduced before, the main area where the Hyper-V implementation differs from its Linux counterpart is in how virtual machines are attached to a given OVS port. This is easily accomplished by using the *Set-VMNetworkAdapterOVSPort* PowerShell cmdlet provided with the installer (please refer to part1 for details on installing OVS).

Let us say that we have a Hyper-V virtual machine called “instance-00000003” and that we want to connect it to the Hyper-V OVS switch. All we have to do for each VM network adapter is to connect it to the Hyper-V Virtual Switch named ***vSwitch*** as you would normally do, assign it to a given OVS port and create the corresponding ports in OVS:

```
$vnic = Get-VMNetworkAdapter instance-00000003
Connect-VMNetworkAdapter -VMNetworkAdapter $vnic -SwitchName vSwitch
$vnic | Set-VMNetworkAdapterOVSPort -OVSPortName vm2
ovs-vsctl.exe add-port br-int vm2
```

Here is how the resulting OVS configuration looks like on Hyper-V:

```
PS C:\> ovs-vsctl.exe show
a81a54fc-0a3c-4152-9a0d-f3cbf4abc3ca
    Bridge br-int
        Port "vm2"
            Interface "vm2"
        Port patch-tun
            Interface patch-tun
                type: patch
                options: {peer=patch-int}
        Port br-int
            Interface br-int
                type: internal
    Bridge br-tun
        Port "stt-1"
            Interface "stt-1"
                type: stt
                options: {in_key=flow, local_ip="14.14.14.2", out_key=flow, remote_ip="14.14.14.1"}
        Port patch-int
            Interface patch-int
                type: patch
                options: {peer=patch-tun}
        Port br-tun
            Interface br-tun
                type: internal
    Bridge "br-port1"
        Port "port1"
            Interface "port1"
        Port "br-port1"
            Interface "br-port1"
                type: internal
```

Further control can be accomplished by applying flow rules.

OVS based networking is now fully functional between KVM and Hyper-V hosted virtual machines!

P.S.: Don’t forget to check out part 1 (OpenStack) (http://superuser.openstack.org/articles/tutorial-open-vswitch-hyper-v-openstack/), part 2 (VXLAN) (http://superuser.openstack.org/articles/manage-hyper-v-open-vswitch/) and part 3 (GRE) (http://superuser.openstack.org/articles/connect-kvm-hyper-v-hosted-vms-using-open-vswitch-gre-tunnel/) of this series if you missed them!

This post first appeared on the Cloudbase Solutions (https://cloudbase.it/open-vswitch-2-5-hyper-v-stt-part-3/) blog. Superuser is always interested in community content, email: editor@superuser.org (mailto:editor@superuser.org).

Cover Photo (https://flic.kr/p/7PLkHj) // CC BY NC (https://creativecommons.org/licenses/by-nc/2.0/)

The post Create an Open vSwitch STT tunnel between KVM and Hyper-V-hosted VMs (http://superuser.openstack.org/articles/create-an-open-vswitch-stt-tunnel-between-kvm-and-hyper-v-hosted-vms/) appeared first on OpenStack Superuser (http://superuser.openstack.org).

by Alin Serdean at February 03, 2017 04:19 PM (http://superuser.openstack.org/articles/create-an-open-vswitch-stt-tunnel-between-kvm-and-hyper-v-hosted-vms/)

James Page (https://javacruft.wordpress.com)

snap install openstackclients (https://javacruft.wordpress.com/2017/02/03/snap-install-openstackclients/)

Over the last month or so I’ve been working on producing snap packages for a variety of OpenStack components. Snaps (http://snapcraft.io) provide a new fully isolated, cross-distribution packaging paradigm which in the case of Python is much more aligned to how Python projects manage their dependencies.

Alongside work on Nova, Neutron, Glance and Keystone snaps (which I’ll blog about later), we’ve also published snaps for end-user tools such as the OpenStack clients, Tempest and Rally.

If you’re running on Ubuntu 16.04 its really simple to install and use the openstackclients snap:

```
sudo snap install --edge --classic openstackclients
```

right now, you’ll also need to enable snap command aliases for all of the clients the snap provides:

```
ls -1 /snap/bin/openstackclients.* | cut -f 2 -d . | xargs sudo snap alias openstackclients
```

after doing this, you’ll have all of the client tools aligned to the OpenStack Newton release available for use on your install:

```
aodh
barbican
ceilometer
cinder
cloudkitty
designate
freezer
glance
heat
ironic
magnum
manila
mistral
monasca
murano
neutron
nova
openstack
sahara
senlin
swift
tacker
trove
vitrage
watcher
```

The snap is currently aligned to the Newton OpenStack release; the intent is to publish snaps aligned to each OpenStack release using the series support that’s planned for snaps – so you’ll be able to pick clients appropriate for any supported OpenStack release or for the current development release.

You can check out the source for the snap on github (https://github.com/openstack/snap-openstackclients); writing a snap package for a Python project is pretty simple, as it makes use of the standard pip tooling to describe dependencies and install Python modules. Kudos to the snapcraft team who have done a great job on the Python plugin.

Let us know what you think by reporting bugs (https://bugs.launchpad.net/snap-openstackclients) or by dropping into #openstack-snaps on Freenode IRC!

Add a Comment (http://feeds.wordpress.com/1.0/gocomments/javacruft.wordpress.com/887/)

by JavaCruft at February 03, 2017 11:07 AM (https://javacruft.wordpress.com/2017/02/03/snap-install-openstackclients/)

StackHPC Team Blog (https://www.stackhpc.com/)

TripleO, NUMA and vCPU Pinning: Improving Guest Performance (https://www.stackhpc.com/tripleo-numa-vcpu-pinning.html)

The hardware powering modern cloud and High Performance Computing (HPC) systems is variable and complex. The assumption that access to memory and devices across a system is uniform is often incorrect. Without knowledge of the properties of the physical hardware of the host, the virtualised guests running atop can perform poorly.

This post covers how we can take advantage of knowledge of the system architecture in OpenStack Nova to improve guest VM performance, and how to configure OpenStack TripleO to support this.

Non-Uniform Memory Access (NUMA)

Server CPU clock speeds have for a long time ceased increasing (https://www.comsol.com/blogs/havent-cpu-clock-speeds-increased-last-years/). In order to continue to improve system performance, CPU and system vendors now scale outwards instead of upwards, offering servers with multiple CPU sockets and multiple cores per CPU. In multi-socket systems access to memory and devices is no longer uniform between the CPU nodes across all memory, as the inter-node communication paths are limited. This leads to variable memory bandwidth and latency, and is known as Non-Uniform Memory Access (NUMA).

When virtualisation is used on NUMA systems, typically guest VMs will have no knowledge of the memory architecture of the physical host. Consequently they will make poor use of the system's resources, making many expensive memory accesses across the interconnect bus. The same is true when accessing I/O devices such as Network Interface Cards (NICs).

To avoid these issues it is possible to expose all or a subset of the memory architecture of the physical system to the guest VM, allowing it to make more intelligent decisions around the use of memory and how tasks are scheduled to CPU cores.

We call this process the "physicalisation" of virtualisation. Revealing the underlying hardware sacrifices some generality and flexibility, but delivers performance gains through informed placement and scheduling. A compromise is struck; we find we can get most of the benefits of software defined infrastructure without paying a price in performance.

vCPU Pinning

In KVM, the virtual CPUs of a guest VM are emulated by host tasks in userspace of the hypervisor. As such they may be scheduled across any of the cores in the system. This behaviour can lead to sub-optimal cache performance as virtual CPUs are scheduled between CPU cores within a NUMA node or worse, between NUMA nodes. With virtual CPU pinning, we can improve this behaviour by restricting the physical CPU cores on which each virtual CPU can run.

It can in some scenarios be advantageous to also restrict host processes to a subset of the available CPU cores to avoid adverse interactions between hypervisor processes and the application workloads.

NUMA and vCPU Pinning in Nova

Support for NUMA topology awareness and vCPU pinning in OpenStack Nova was first introduced in October 2015 with the Juno release. These features allow Nova to more intelligently schedule VM instances onto the available hardware. Essentially, we can request a NUMA topology via Nova flavor keys or Glance image properties when creating a Nova instance. The same is true for vCPU pinning. The OpenStack admin guide (https://docs.openstack.org/admin-guide/compute-cpu-topologies.html) provides some useful information on how to use these features. There is a good blog article (http://redhatstackblog.redhat.com/2015/05/05/cpu-pinning-and-numa-topology-awareness-in-openstack-compute/) by Red Hat on this topic which is worth reading. We are going to build on that here by describing how to deliver these capabilities in TripleO.

NUMA and vCPU Pinning in TripleO

The OpenStack TripleO (http://www.tripleo.org) project provides tools to deploy an OpenStack cloud, and Red Hat's popular OpenStack Platform (OSP) (https://access.redhat.com/documentation/en/red-hat-openstack-platform) is based on TripleO. The default configuration of TripleO is not optimal for NUMA placement and vCPU pinning, so we'll outline a few steps that can be taken to improve the situation.

Kernel Command Line Arguments

We can use the `isolcpus` kernel command line argument to restrict host processes to a subset of the total available CPU cores. The argument specifies a list of ranges of CPU IDs from which host processes should be isolated. In other words, the CPUs we will use for guest VMs. For example, to reserve CPUs 4 through 23 exclusively for guest VMs we could specify:

```
isolcpus=4-23
```

Ideally we want this argument to be applied on the first and subsequent boots rather than applying it dynamically during deployment, to avoid waiting for our compute nodes to reboot. Currently Ironic does not provide a mechanism to specify additional kernel arguments on a per-node or per-image basis, so we must bake them into the overcloud image instead.

If using the Grub bootloader, additional arguments can be provided to the kernel by modifying the `GRUB_CMDLINE_LINUX` variable in `/etc/default/grub` in the overcloud compute image, then rebuilding the Grub configuration. We use the `virt-customize` command to apply post-build configuration to the overcloud images:

```
$ export ISOLCPUS=4-23
$ function cpu_pinning_args {
      CPU_PINNING_ARGS="isolcpus=${ISOLCPUS}"
      echo --run-command \"echo GRUB_CMDLINE_LINUX=\"'\\\"'\"\\$\{GRUB_CMDLINE_LINUX\} ${CPU_PINNING_ARGS}\"'\\\"'\" \>\> /etc/default/grub\"
  }
$ (cpu_pinning_args) | xargs virt-customize -v -m 4096 --smp 4 -a overcloud-compute.qcow2
```

(We structure the execution this way because typically we are composing a string of operations into a single invocation of `virt-customize`). Alternatively this change could be a applied with a custom `diskimage-builder` element.

One Size Does Not Fit All: Multiple Overcloud Images

While the isolcpus argument may provide performance benefits for guest VMs on compute nodes, it would be seriously harmful to limit host processes in the same way on controller and storage nodes. With different sets of nodes requiring different arguments, we now need multiple overcloud images. Thankfully, TripleO provides an (undocumented) set of options to set the image for each of the overcloud roles. We'll use the name `overcloud-compute` for the compute image here.

When uploading overcloud images to Glance, use the `OS_IMAGE` environment variable to reference an image with a non-default name:

```
$ export OS_IMAGE=overcloud-compute.qcow2
$ openstack overcloud image upload
```

We can execute this command multiple times to register multiple images. To specify a different image for the overcloud compute roles, create a Heat environment file containing the following:

```
parameter_defaults:
  NovaImage: overcloud-compute
```

Ensure that the image name matches the one registered with Glance and that the environment file is referenced when deploying or updating the overcloud. Other node roles will continue to use the default image `overcloud-full`. Our specialised kernel configuration is now only applied where it is needed, and not where it is harmful.

KVM and Libvirt

The Nova compute service will not advertise the NUMA topology of its host if it determines that the versions of libvirt and KVM are inappropriate. As of the Mitaka release, the following version restrictions are applied:

- `libvirt: >= 1.2.8, != 1.2.9.7`
- `qemu-kvm: >= 2.1.0`

On CentOS 7.3, the `qemu-kvm` package is at version `1.5.3`. This can be updated to a more contemporary `2.4.1` by adding the kvm-common (http://mirror.centos.org/centos/7/virt/x86_64/kvm-common/) Yum repository and installing qemu-kvm-ev:

```
$ cat << EOF | sudo tee /etc/yum.repos.d/kvm-common.repo
[kvm-common]
name=KVM common
baseurl=http://mirror.centos.org/centos/7/virt/x86_64/kvm-common/
gpgcheck=0
EOF
$ sudo yum -y install qemu-kvm-ev
```

This should be applied to the compute nodes either after deployment or during the overcloud compute image build procedure. If applied after deployment, the `openstack-nova-compute` service should be restarted on the compute nodes to ensure it checks the library versions again:

```
$ sudo systemctl restart openstack-nova-compute
```

Kernel Same-page Merging (KSM)

Kernel Same-page Merging (KSM) allows for more efficient use of memory by guest VMs by allowing identical memory pages used by different processes to be backed by a single page of Copy On Write (COW) physical memory. When used on a NUMA system this can have adverse effects if pages are merged belonging to VMs on different nodes. In Linux kernels prior to 3.14 KSM was not NUMA-aware, which could lead to VM performance and stability issues. It is possible to disable KSM merging memory across NUMA nodes:

```
$ echo 0 > /sys/kernel/mm/ksm/merge_across_nodes
```

Some additional commmands (https://www.centos.org/forums/viewtopic.php?t=59749) are required if any shared pages are already in use. If memory is not constrained, greater performance may be achieved by disabling KSM altogether:

```
$ echo 0 > /sys/kernel/mm/ksm/run
```

Nova Scheduler Configuration

The Nova scheduler provides the `NUMATopologyFilter` filter to incorporate NUMA topology information into the placement process. TripleO does not appear to provide a mechanism to append additional filters to the default list (although it may be possible with sufficient 'puppet-fu'). To override the default scheduler filter list, use a Heat environment file like the following:

```
parameter_defaults:
  controllerExtraConfig:
    nova::scheduler::filter::scheduler_default_filters:
      - RetryFilter
      - AvailabilityZoneFilter
      - RamFilter
      - DiskFilter
      - ComputeFilter
      - ComputeCapabilitiesFilter
      - ImagePropertiesFilter
      - ServerGroupAntiAffinityFilter
      - ServerGroupAffinityFilter
      - NUMATopologyFilter
```

The `controllerExtraConfig` parameter (recently renamed to `ControllerExtraConfig`) allows us to specialise the overcloud configuration. Here `nova::scheduler::filter::scheduler_default_filters` references a variable in the Nova scheduler puppet manifest (https://github.com/openstack/puppet-nova/blob/stable/mitaka/manifests/scheduler/filter.pp). Be sure to include this environment file in your openstack overcloud deploy command as a -e argument.

Nova Compute Configuration

The Nova compute service can be configured to pin virtual CPUs to a subset of the physical CPUs. We can use the set of CPUs previously isolated via kernel arguments. It is also prudent to reserve an amount of memory for the host processes. In TripleO we can again use a Heat environment file to set these options:

```
parameter_defaults:
  NovaComputeExtraConfig:
    nova::compute::vcpu_pin_set: 4-23
    nova::compute::reserved_host_memory: 2048
```

Here we are using CPUs 4 through 23 for vCPU pinning and reserving 2GB of memory for host processes. As before, remember to include this environment file when managing the TripleO overcloud.

Performance For All

We hope this guide helps the community to improve the performance of their TripleO-based OpenStack deployments. Thanks to the University of Cambridge for the use of their development cloud while developing this configuration.

Further Reading

- OpenStack for Scientific Research (https://www.openstack.org/science/)
- StackHPC blog: OpenStack and Virtualised HPC (https://stackhpc.com/hpc-and-virtualisation.html)
- Red Hat Virtualization Tuning and Optimization Guide (https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Virtualization_Tuning_and_Optimization_Guide/sect-Virtualization_Tuning_Optimization_Guide-NUMA-NUMA_and_libvirt.html)
- OpenStack Nova NUMA placement specification (https://specs.openstack.org/openstack/nova-specs/specs/juno/implemented/virt-driver-numa-placement.html)
- OpenStack Nova vCPU pinning specification (https://specs.openstack.org/openstack/nova-specs/specs/juno/approved/virt-driver-cpu-pinning.html)
- StackHPC blog: Understanding VXLAN & OVS bandwidth (https://stackhpc.com/vxlan-ovs-bandwidth.html)

by Mark Goddard at February 03, 2017 10:30 AM (https://www.stackhpc.com/tripleo-numa-vcpu-pinning.html)

About

Planet OpenStack is a collection of thoughts from the developers and other key players of the OpenStack projects. If you are working on OpenStack technology you should add your OpenStack blog (http://wiki.openstack.org/AddingYourBlog).

Subscriptions

- (http://blog.aaronorosen.com/category/openstack/feed/) Aaron Rosen (http://blog.aaronorosen.com)
- (http://blog.adamspiers.org/tag/openstack/feed/atom/) Adam Spiers (http://blog.adamspiers.org)
- (http://adam.younglogic.com/category/software/openstack/feed/) Adam Young (http://adam.younglogic.com)
- (http://www.blogger.com/feeds/5556854748152045563/posts/default/-/OpenStack) Aditya Patawari (http://blog.adityapatawari.com/search/label/OpenStack)
- (http://www.17od.com/tag/openstack/feed/) Adrian Smith (http://www.17od.com)
- (https://cloudbase.it/category/openstack/feed/) Alessandro Pilotti (https://cloudbase.it)
- (https://aababilov.wordpress.com/category/openstack-2/feed/) Alessio Ababilov (https://aababilov.wordpress.com)
- (https://alvarolopez.github.io/feeds/openstack.atom.xml) Alvaro Lopez Garcia (http://alvarolopez.github.io/)
- (http://www.buildcloudstorage.com/feeds/posts/default) Amar Kapadia (http://www.buildcloudstorage.com/)
- (https://hypecycles.com/category/OpenStack/feed/) Amrith Kumar (https://hypecycles.com)
- (http://amalagon.github.io/blog/categories/openstack/atom.xml) Ana Malagon (http://amalagon.github.io/)
- (http://www.blogger.com/feeds/8789296710216751709/posts/default/-/OpenStack) Andreas Jaeger (http://jaegerandi.blogspot.com/search/label/OpenStack)
- (https://virtualandy.wordpress.com/tag/openstack/feed/atom/) Andy Hill (https://virtualandy.wordpress.com)
- (https://ahsalkeld.wordpress.com/category/openstack/feed/atom/) Angus Salkeld (https://ahsalkeld.wordpress.com)
- (http://anteaya.info/atom.xml) Anita Kuno (http://anteaya.info/)
- (http://justwriteclick.com/tag/openstack/feed/) Anne Gentle (https://justwriteclick.com)
- (http://ops.anthonygoddard.com/blog/categories/openstack/atom.xml) Anthony Goddard (http://ops.anthonygoddard.com/)
- (https://www.reversengineered.com/tag/openstack/feed/) Antony Messerli (https://www.reversengineered.com)
- (http://blog.appformix.com/rss.xml?tag=openstack) AppFormix (http://blog.appformix.com)
- (https://feeds.feedburner.com/Aptira?format=xml) Aptira (https://aptira.com)
- (http://abregman.com/category/openstack/feed/) Arie Bregman (http://abregman.com)
- (http://www.berezins.com/category/openstack/feed/) Arthur Berezin (http://www.berezins.com)
- (https://notartom.net/category/openstack/feed/) Artom Lifshitz (https://notartom.net)
- (https://arunsag.wordpress.com/tag/openstack/feed/) Arun S A G (https://arunsag.wordpress.com)
- (http://arxcruz.net/category/openstack/feed/) Arx Cruz (http://arxcruz.net)
- (https://assafmuller.com/category/openstack/feed/) Assaf Muller (https://assafmuller.com)
- (http://www.blogger.com/feeds/3693502231491518239/posts/default/-/openstack) Augustina Ragwitz (http://hackhackblog.blogspot.com/search/label/openstack)
- (http://blog.nemebean.com/taxonomy/term/3/feed) Ben Nemec (http://blog.nemebean.com/tags/openstack)
- (http://benjaminkerensa.com/category/openstack/feed) Benjamin Kerensa (http://benjaminkerensa.com)
- (http://blog.cafarelli.fr/category/openstack/feed/) Bernard Cafarelli (http://blog.cafarelli.fr)
- (https://blog.betacloud.io/tag/openstack-planet/feed/atom/) Betacloud (https://betacloud.io)
- (http://www.logicative.com/category/openstack/feed/) Beth Elwell (http://www.logicative.com)
- (http://www.blogger.com/feeds/8711250493465785199/posts/default/-/openstack) Boden Russell (http://bodenr.blogspot.com/search/label/openstack)
- (http://www.bretpiatt.com/blog/category/openstack/feed/) Bret Piatt (http://www.bretpiatt.com/blog)
- (http://blog.briancurtin.com/categories/openstack.xml) Brian Curtin (http://briancurtin.com/blog/)
- (http://www.blogger.com/feeds/1977547450130135970/posts/default/-/OpenStack) Cameron Seader (http://blog.seader.us/search/label/OpenStack)
- (http://captainkvm.com/tag/openstack/feed/) Captain KVM (http://captainkvm.com)
- (http://blog.episodicgenius.com/tags/openstack/index.xml) Carl Baldwin (http://blog.episodicgenius.com/tags/openstack/index.xml)
- (http://anstack.github.io/atom.xml) Carlos Camacho (http://anstack.github.io/)
- (http://ceph.com/community/blog/tag/openstack/feed/) Ceph (http://ceph.com)
- (http://chandankumar.dgplug.org/categories/openstack.xml) Chandan Kumar (http://chandankumar.dgplug.org/)
- (https://blog.chmouel.com/category/openstack/feed/) Chmouel Boudjnah (https://blog.chmouel.com)
- (https://anticdent.org/feeds/openstack.atom.xml) Chris Dent (https://anticdent.org/)
- (http://chris.yeoh.info/category/openstack/feed/) Chris Yeoh (http://chris.yeoh.info)
- (https://blog.christophersmart.com/category/foss/openstack/feed/) Christopher Smart (https://blog.christophersmart.com)
- (http://cpallar.es/tagged/openstack/rss) Cindy Pallares (http://cpallar.es/)
- (http://citrix-openstack.siteleaf.net/feed.xml) Citrix OpenStack Blog (http://citrix-openstack.siteleaf.net/)
- (http://fewbar.com/feed.xml) Clint Byrum (http://fewbar.com/)
- (https://www.symantec.com/connect/item-feeds/all/all/feed/all/all/3249921) Cloud Platform @ Symantec (https://www.symantec.com/connect/item-feeds/all/all/feed/all/all/3249921)
- (https://cloudbau.github.io/feed.openstack.xml) Cloudbau Blog (https://cloudbau.github.io)
- (http://getcloudify.org/tags/OpenStack/atom.xml) Cloudify Engineering
- (http://www.cloudscaling.com/blog/category/openstack/feed/) Cloudscaling Corporate Blog (http://www.cloudscaling.com)
- (http://feeds.feedburner.com/SimplicityScales?format=xml) Cloudscaling Engineering (http://engineering.cloudscaling.com)
- (https://dev.cloudwatt.com/en/blog/rss.xml) Cloudwatt (https://dev.cloudwatt.com/en/blog/index.html)
- (http://openstack.prov12n.com/category/openstack/feed/) Cody Bunch (http://openstack.prov12n.com)
- (https://mcwhirter.com.au/tags/OpenStack/index.atom) Craige McWhirter (http://mcwhirter.com.au//tags/OpenStack/)
- (https://dronopenstack.wordpress.com/category/openstack/feed/) Dafna Ron (https://dronopenstack.wordpress.com)
- (http://blog.thekingshotts.com/category/openstack/feed/) Dan Kingshott (http://blog.thekingshotts.com)
- (https://dprince.github.io/feeds/tag_openstack.atom.xml) Dan Prince (https://dprince.github.io/)
- (http://www.jaddog.org/category/openstack/feed/) Dan Radez (http://www.jaddog.org)
- (http://www.danplanet.com/blog/category/openstack/feed/) Dan Smith (http://www.danplanet.com/blog)
- (https://www.berrange.com/topics/openstack/feed/) Daniel P. Berrangé (https://www.berrange.com)
- (https://www.stackevolution.com/taxonomy/term/2/feed) Darryl Weaver (https://www.stackevolution.com/openstack)
- (http://www.blogger.com/feeds/93851419426953678/posts/default/-/openstack) David Medberry (http://dowdberry.blogspot.com/search/label/openstack)
- (https://dmsimard.com/category/openstack/rss.xml) David Moreau Simard (https://dmsimard.com/)

- (https://dstanek.com/tags/openstack/index.xml) David Stanek (https://dstanek.com/tags/openstack/index.xml)
- (http://hackstack.org/x/blog/category/openstack/feed/) Dean Troyer (http://hackstack.org/x/blog)
- (http://goodsquishy.com/tag/openstack/feed/atom/) Derek Higgins (http://goodsquishy.com)
- (http://feeds.doughellmann.com/doughellmann/openstack) Doug Hellmann (https://doughellmann.com/blog)
- (http://www.dougalmatthews.com/feeds/tag/openstack.atom.xml) Dougal Matthews (http://www.dougalmatthews.com/)
- (https://www.dreamhost.com/blog/tag/openstack/feed/) DreamHost (https://www.dreamhost.com/blog)
- (http://www.blogger.com/feeds/8825992/posts/default/-/openstack) Duncan McGreggor (http://oubiwann.blogspot.com/search/label/openstack)
- (https://blog.leafe.com/category/openstack/feed/) Ed Leafe (https://blog.leafe.com)
- (http://princessleia.com/journal/category/openstack/feed/) Elizabeth K. Joseph (http://princessleia.com/journal)
- (http://my1.fr/blog/category/virtualization/openstack/feed/atom/) Emilien Macchi (http://my1.fr/blog)
- (https://enriquetaso.com/category/OpenStack/feed/) Enriquez Laura Sofia (https://enriquetaso.com)
- (http://itsonlyme.name/openstack/storlets/rss.xml) Eran Rom (http://itsonlyme.name/blog)
- (http://blog.phymata.com/category/openstack.atom.xml) Everett Toews (http://blog.phymata.com/)
- (http://blog.fsquat.net/?tag=openstackplanet&feed=rss2) Fabien Boucher (http://blog.fsquat.net)
- (http://blog.flaper87.com/feeds/openstack.atom.xml) Flavio Percoco (https://blog.flaper87.com/)
- (https://fleio.com/blog/tag/openstack/feed/) Fleio Blog (https://fleio.com/blog)
- (http://www.florentflament.com/blog/feeds/openstack.atom.xml) Florent Flament (http://www.florentflament.com/blog/)
- (http://galsagie.github.io/feed.xml) Gal Sagie (http://galsagie.github.io)
- (http://galeracluster.com/category/planet-openstack/feed/) Galera Cluster by Codership (http://galeracluster.com/category/blog/)
- () Geetika Batra
- (http://giuliofidente.com/feeds/tag/openstack.atom.xml) Giulio Fidente (http://giuliofidente.com/)
- (http://gorka.eguileor.com/feed/?cat=10&feed=rss2) Gorka Eguileor (https://gorka.eguileor.com)
- (http://graham.hayes.ie/categories/cat_openstack.xml) Graham Hayes (http://graham.hayes.ie/)
- (https://openstackgd.wordpress.com/feed/) Grid Dynamics OpenStack Team (https://openstackgd.wordpress.com)
- (http://community.hpe.com/html/assets/hp-maint.png) HP Public Cloud (http://www.hpcloud.com/blog/categories/1246)
- (http://ehaselwanter.com/en/openstack.xml) Haselwanter Edmund (http://ehaselwanter.com)
- (http://blog.hendrikvolkmer.de/feed/OpenStack/atom.xml) Hendrik Volkmer (http://blog.hendrikvolkmer.de/)
- (http://hugh.blemings.id.au/category/openstack/feed/) Hugh Blemings (http://hugh.blemings.id.au)
- (https://developer.ibm.com/opentech/category/openstack/feed/) IBM OpenTech Team (https://developer.ibm.com/opentech)
- (https://blog.zhaw.ch/icclab/category/articles/openstack-2/feed/) ICCLab (https://blog.zhaw.ch/icclab)
- (http://drop.isi.edu/node/407/rss) ISI High Performance Cloud Computing Group (http://www.isi.edu/research_groups/HPCC/blog)
- (http://blog.imaginea.com/tag/openstack/) Imaginea Technologies
- (https://blog.imaginea.com/tag/openstack/feed/) Imaginea Technologies (https://blog.imaginea.com)
- (http://www.infralovers.com/en/articles/categories/openstack/atom.xml) Infralovers (http://www.infralovers.com/)
- (http://www.internap.com/feed/?cat=1001410) Internap (http://www.internap.com)
- (http://jjasghar.github.io/blog/categories/openstack/atom.xml) JJ Asghar (http://jjasghar.github.io/)
- (http://amo-probos.org/feed/rss2_0?tag=openstack) James E. Blair (http://amo-probos.org/)
- (https://javacruft.wordpress.com/category/openstack/feed/) James Page (https://javacruft.wordpress.com)
- (http://blog-slagle.rhcloud.com/?feed=rss2&tag=OpenStack) James Slagle (http://blog-slagle.rhcloud.com)
- (http://www.jamielennox.net/blog/categories/openstack/atom.xml) Jamie Lennox (http://www.jamielennox.net/)
- (http://www.jpena.net/?cat=7&feed=atom) Javier Peña (http://www.jpena.net)
- (http://www.jasondotstar.com/feed.openstack.xml) Jay Clark (http://www.jasondotstar.com)
- (http://www.joinfu.com/category/openstack/feed/) Jay Pipes (http://www.joinfu.com)
- (http://odyssey4me.github.io/openstack.xml) Jesse Pretorius (http://odyssey4me.github.io/)
- (https://software.intel.com/en-us/user/335367/feed) Jiangang Duan (Intel) (https://software.intel.com/en-us/recent/335367)
- (https://jfehlig.wordpress.com/category/openstack/feed/) Jim Fehlig (https://jfehlig.wordpress.com)
- (https://tropicaldevel.wordpress.com/category/openstack/feed/) John Bresnahan (https://tropicaldevel.wordpress.com)
- (http://programmerthoughts.com/openstack/openstackposts.xml) John Dickinson (http://programmerthoughts.com/)
- (http://john.eckersberg.com/feeds/tag/openstack.atom.xml) John Eckersberg (http://john.eckersberg.com/)
- (https://griffithscorner.wordpress.com/category/openstack/feed/) John Griffith (https://griffithscorner.wordpress.com)
- (http://blog.iwokeupcoveredinblood.com/blog/categories/openstack/atom.xml) Jon Proulx (http://blog.iwokeupcoveredinblood.com/)
- (http://josh.people.rcbops.com/tag/openstack/feed/) Joshua Hesketh (http://josh.people.rcbops.com)
- (http://www.jpichon.net/feeds/atom/tag/openstack/) Julie Pichon (http://www.jpichon.net/tag/openstack/)
- (https://julien.danjou.info/blog/tags/OpenStack.xml) Julien Danjou (https://julien.danjou.info/blog/)
- (http://www.juliosblog.com/tag/openstack/rss/) Julio Villarreal Pelegrino (http://www.juliosblog.com/)
- (https://kashyapc.com/tag/openstack/feed/) Kashyap Chamarthy (https://kashyapc.com)
- (https://cloud-guy.net/category/openstack/feed/atom/) Keith Tobin (https://cloud-guy.net)
- (http://ken.pepple.info/openstack.atom) Ken Pepple (http://ken.pepple.info)
- (https://cloudarchitectmusings.com/tag/openstack/feed/) Kenneth Hui (https://cloudarchitectmusings.com)
- (https://kimizhang.wordpress.com/category/openstack/feed/) Kimi Zhang (https://kimizhang.wordpress.com)
- (http://www.siliconloons.com/categories/openstack/index.xml) Kyle Mestery (https://www.siliconloons.com/categories/openstack/)
- (http://lbragstad.com/category/openstack/feed/) Lance Bragstad (http://lbragstad.com)
- (http://blog.oddbit.com/tag/openstack/rss/) Lars Kellogg-Stedman (http://blog.oddbit.com/)
- (http://www.thequietstack.com/?tag=openstack&feed=rss) Laura Alves (http://www.thequietstack.com)
- (http://uxd-stackabledesign.rhcloud.com/category/openstack/feed/) Liz Blanchard (http://uxd-stackabledesign.rhcloud.com)
- (http://feeds.feedburner.com/logilaborg_openstack) Logilab (https://www.logilab.org/view?rql=Any%20X%20ORDERBY%20AA%20DESC%20LIMIT%2010%20WHERE%20E%20eid%20115182%2C%20E%20tags%20X%2C%20X%20creation_date%20AA%2C%20
- (https://lorinhochstein.wordpress.com/category/openstack/feed/atom/) Lorin Hochstein (https://lorinhochstein.wordpress.com)
- (http://dachary.org/?feed=rss2&cat=32) Loïc Dachary (http://dachary.org)
- (http://www.blogger.com/feeds/5819640694385843490/posts/default/-/OpenStack) Maish Saidel-Keesing (http://technodrone.blogspot.com/search/label/OpenStack)
- (https://major.io/tag/openstack/feed/) Major Hayden (https://major.io)
- (http://www.blogger.com/feeds/7208049467061137140/posts/default/-/openstack) Manishanker Talusani (http://manishankert.blogspot.com/search/label/openstack)
- (http://blog.mbonell.com/category/openstack/feed/) Marcela Bonell (http://blog.mbonell.com)
- (https://crustyblaa.com/feeds/openstack.atom.xml) Mark McLoughlin (https://crustyblaa.com/)
- (http://www.markshuttleworth.com/archives/tag/openstack/feed) Mark Shuttleworth (http://www.markshuttleworth.com)
- (http://markvoelker.github.io/blog/index.xml) Mark T. Voelker (http://markvoelker.github.io/blog/)

- (http://hybridcloudburst.com/category/openstack/feed/atom/) Marten Hauville (https://hybridcloudburst.wordpress.com)
- (http://mkissam.tumblr.com/tagged/openstack/rss) Marton Kiss (http://mkissam.tumblr.com/)
- (https://domsch.com/blog/?feed=atom&tag=openstack) Matt Domsch (https://domsch.com/blog)
- (http://www.madorn.com/feeds/openstack.rss.xml) Matt Dorn (http://www.madorn.com/)
- (https://engineeredweb.com/tag/OpenStack%20Planet/atom.xml) Matt Farina (https://engineeredweb.com/tag/OpenStack%20Planet/)
- (http://www.mattfischer.com/blog/?feed=rss2&cat=50) Matt Fischer (http://www.mattfischer.com/blog)
- (http://www.mattgriffin.com/tag/openstack/feed/) Matt Griffin (http://www.mattgriffin.com)
- (https://leastresistance.wordpress.com/category/openstack/feed/) Matt Ray (https://leastresistance.wordpress.com)
- (https://spinningmatt.wordpress.com/category/openstack/feed/atom/) Matthew Farrellee (https://spinningmatt.wordpress.com)
- (http://blog.kortar.org/?cat=2&feed=rss2) Matthew Treinish (http://blog.kortar.org)
- (https://www.matthias-runge.de/tags/openstack.atom.xml) Matthias Runge (http://www.matthias-runge.de/)
- (https://blog.sileht.net/feeds/openstack.atom.xml) Mehdi Abaakouk (https://blog.sileht.net/)
- (http://www.blogger.com/feeds/1011343219801614525/posts/default/-/openstack) Michael Davies (http://lifelog.michaeldavies.org/search/label/openstack)
- (https://krotscheck.net/tag/openstack/feed) Michael Krotscheck (https://krotscheck.net)
- (http://www.stillhq.com/openstack/index.rss) Michael Still (http://www.stillhq.com)
- (http://acksyn.org/categories/openstack.xml) Michele Baldessari (http://acksyn.org/)
- (https://blog.midonet.org/category/openstack/feed/) MidoNet (https://blog.midonet.org)
- (http://blog.midokura.com/category/OpenStack/feed/) Midokura (http://blog.midokura.com)
- (http://www.ajo.es/tagged/openstack/rss) Miguel Ángel Ajo (http://www.ajo.es/)
- (http://www.blogger.com/feeds/4447965219658920183/posts/default/-/Openstack) Mika Ayenson (http://ayenson.blogspot.com/search/label/Openstack)
- (http://www.dorm.org/blog/category/openstack/feed/) Mike Dorman (http://www.dorm.org/blog)
- (https://mirandazhangq.wordpress.com/category/openstack/feed/atom/) Miranda ZHANG (https://mirandazhangq.wordpress.com)
- (https://www.mirantis.com/blog/feed/) Mirantis (https://www.mirantis.com)
- (http://inaugust.com/feed/rss2_0?tag=openstack) Monty Taylor (http://inaugust.com/)
- (https://blog.nfvpe.site/feed/) NFVPE @ Red Hat (https://blog.nfvpe.site)
- (https://blog-nkinder.rhcloud.com/?cat=2&feed=atom) Nathan Kinder (https://blog-nkinder.rhcloud.com)
- (https://thenetworkway.wordpress.com/tag/openstack/feed/) Nir Yechiel (https://thenetworkway.wordpress.com)
- (https://osic.org/blog/feed) OSIC - The OpenStack Innovation Center (https://osic.org/blog)
- (https://www.olindata.com/taxonomy/term/309/feed) OlinData (https://www.olindata.com/tags/openstack)
- (http://netapp.io/category/openstack/) OpenStack @ NetApp
- (https://www.openstack.org/blog/feed/) OpenStack Blog (https://www.openstack.org/blog)
- (https://openstackindia.wordpress.com/feed/) OpenStack India (https://openstackindia.wordpress.com)
- (http://openStackindiaonline.github.io/feeds/all.atom.xml) OpenStack India Online (http://openStackindiaonline.github.io/)
- (https://novarollup.wordpress.com/feed/) OpenStack Nova Developer Rollup (https://novarollup.wordpress.com)
- (https://blog.chmouel.com/) OpenStack Reactions (https://blog.chmouel.com)
- (http://secstack.org/category/openstack/feed/) OpenStack Security Blog (http://secstack.org)
- (http://superuser.openstack.org/feed/) OpenStack Superuser (http://superuser.openstack.org)
- (http://www.blogger.com/feeds/6032896665180559/posts/default) OpenStack in Production (http://openstack-in-production.blogspot.com/)
- (https://opensource.com/taxonomy/term/5126/feed/feed) Opensource.com (https://opensource.com/taxonomy/term/5126/feed/feed)
- (https://openstack-security.github.io/feed.xml) Openstack Security Project (https://openstack-security.github.io/)
- (http://www.blogger.com/feeds/7675777825226307020/posts/default/-/openstack) Openstack-br (http://openstackbr.blogspot.com/search/label/openstack)
- (http://mezzanine-lordkrandel.rhcloud.com/tag/openstack/feeds/rss/) Paolo Gatti (http://mezzanine-lordkrandel.rhcloud.com/)
- (https://www.percona.com/blog/category/openstack/feed/) Percona (https://www.percona.com/blog)
- (http://www.cisco.com/c/en/us/about/corporate-strategy-office/acquisitions/piston.html) Piston (http://pistoncloud.com)
- (http://www.pixelbeat.org/feed/openstack.rss2.xml) Pádraig Brady (http://www.pixelbeat.org/)
- (http://rdoproject.org/blog/feed.xml) RDO (http://rdoproject.org/blog/)
- (https://blogs.rdoproject.org/category/openstackdev/feed) RDO Blogs (http://blogs.rdoproject.org)
- (https://developer.rackspace.com/blog/categories/openstack/atom.xml) Rackspace Developer Blog (https://developer.rackspace.com)
- (https://trickycloud.wordpress.com/category/openstack/feed/atom/) Ramon Acedo (https://trickycloud.wordpress.com)
- (https://www.ravellosystems.com/blog/tag/openstack/feed) Ravello Systems (https://www.ravellosystems.com/blog)
- (http://redhatstackblog.redhat.com/feed/) Red Hat Stack (http://redhatstackblog.redhat.com)
- (http://drbacchus.com/tag/openstack/feed/) Rich Bowen (http://drbacchus.com)
- (http://mechanicalcat.net/richard/log/OpenStack/rss) Richard Jones (http://mechanicalcat.net/richard/log/OpenStack)
- (https://robhirschfeld.com/category/openstack/feed/) Rob Hirschfeld (https://robhirschfeld.com)
- (https://rbtcollins.wordpress.com/tag/openstack/feed/atom/) Robert Collins (https://rbtcollins.wordpress.com)
- (http://ronaldbradford.com/blog/category/cloud/openstack/feed/) Ronald Bradford (http://ronaldbradford.com/blog)
- (http://en.roozbehshafiee.com/category/openstack/feed) Roozbeh Shafiee (http://en.roozbehshafiee.com)
- (https://rossella-sblendido.net/category/openstack/feed/) Rossella Sblendido (https://rossella-sblendido.net)
- (https://blog.russellbryant.net/category/openstack/feed/) Russell Bryant (https://blog.russellbryant.net)
- (http://feeds.feedburner.com/RyanLanesBlog_openstack) Ryan Lane (https://blog.ryandlane.com)
- (https://www.suse.com/communities/blog/tag/openstack/feed/) SUSE Conversations (https://www.suse.com/communities/blog)
- (https://cloudblog.switch.ch/category/openstack/feed/atom/) SWITCH Cloud Blog (https://cloudblog.switch.ch)
- (http://blog.mathys.io/feeds/posts/default/-/OpenStack?alt=rss) Sandro Mathys (http://blog.mathys.io/search/label/OpenStack)
- (https://saschpe.wordpress.com/category/openstack/feed/) Sascha Peilicke (https://saschpe.wordpress.com)
- (https://dague.net/category/openstack/feed/) Sean Dague (https://dague.net)
- (http://sarob.com/category/openstack/feed/) Sean Roberts (https://sarob.com)
- (https://shaifaliagrawal.wordpress.com/category/openstack/feed/) Shaifali Agrawal (https://shaifaliagrawal.wordpress.com)
- (http://www.debug-all.com/?feed=rss2&tag=openstack) Shannon McFarland (http://www.debug-all.com)
- (http://solinea.com/tag/openstack/feed) Solinea (http://solinea.com)
- (http://blog.warma.dk/tag/planetopenstack/feed/) Soren Hansen (http://blog.linux2go.dk)
- (http://engineering.spilgames.com/category/planetopenstack/feed/) Spilgames Engineering (http://engineering.spilgames.com)
- (https://www.stackhpc.com/feeds/stackhpc.atom.xml) StackHPC Team Blog (https://www.stackhpc.com/)
- (http://www.stackmasters.eu/tag/openstack/feed/) Stackmasters team (http://www.stackmasters.eu)
- (https://maffulli.net/category/openstack/feed/) Stefano Maffulli (https://maffulli.net)
- (http://www.blogger.com/feeds/7938511026917047919/posts/default/-/openstack) Steve Baker (http://blog.stevebaker.org/search/label/openstack)
- (https://sdake.io/tag/openstack/feed/) Steve Dake (https://sdake.io)
- (http://www.blogger.com/feeds/7962934533489004582/posts/default/-/openstack) Steve Hardy (http://hardysteven.blogspot.com/search/label/openstack)
- (http://blog.coolsvap.net/category/openstack/feed/) Swapnil Kulkarni (http://blog.coolsvap.net)

(http://swiftstack.com/blog/categories/planetopenstack/atom.xml) SwiftStack Team (http://swiftstack.com/)
(https://sbauza.wordpress.com/tag/openstack/feed/) Sylvain Bauza (https://sbauza.wordpress.com)
(http://www.sebastien-han.fr/openstack.xml) Sébastien Han (https://sebastien-han.fr/)
(http://telekomcloud.github.io/atom.xml) TelekomCloud DevOps team (http://telekomcloud.github.io)
(http://terriyu.info/blog/feeds/tag/openstack.atom.xml) Terri Yu (http://terriyu.info/blog/)
(http://blog.otherwiseguy.com/tag/openstack/rss/) Terry Wilson (http://blog.otherwiseguy.com/)
(http://www.tesora.com/tag/openstack/feed/) Tesora Corp (http://www.tesora.com)
(https://blog.rackspace.com/tag/openstack/feed) The Official Rackspace Blog (https://blog.rackspace.com)
(https://storyboard-blog.sotk.co.uk/) The StoryBoard Team
(https://ttx.re/feeds/openstack.atom.xml) Thierry Carrez (https://ttx.re/)
(https://toabctl.wordpress.com/tag/openstack/feed/) Thomas Bechtold (https://toabctl.wordpress.com)
(https://ubuntuserver.wordpress.com/tag/openstack/feed/) Ubuntu Server & Cloud blog (https://ubuntuserver.wordpress.com)
(http://vmartinezdelacruz.com/category/OpenStack/feed/) Victoria Martínez de la Cruz (http://vmartinezdelacruz.com)
(http://www.vuntz.net/journal/feed/tag/openstack/atom) Vincent Untz (http://www.vuntz.net/journal/)
(https://blog.xenproject.org/tag/openstack/feed/) Xen Project Blog (https://blog.xenproject.org)
(http://www.blogger.com/feeds/2528454834564901150/posts/default/-/openstack) Yun Mao (http://cloudystuffhappens.blogspot.com/search/label/openstack)
(http://www.zerobanana.com/feeds/tags/OpenStack) Zane Bitter (http://www.zerobanana.com/tags/OpenStack)
(http://www.zmanda.com/blogs/?feed=rss2&cat=22) Zmanda (http://www.zmanda.com/blogs)
(https://www.hastexo.com/feeds/openstack.atom.xml) hastexo (https://www.hastexo.com/)
(http://sites.rcbops.com/lca2014_openstack/feed/) linux.conf.au 2014 OpenStack miniconf (http://sites.rcbops.com/openstack_miniconf)
(https://lizards.opensuse.org/tag/openstack/feed/) openSUSE Lizards (https://lizards.opensuse.org)

Last updated:
February 20, 2017 08:39 AM
All times are UTC.

Powered by:
(http://www.planetplanet.org/)

OpenStack

About the Foundation (/foundation)
Projects (http://openstack.org/projects/)
OpenStack Security (http://openstack.org/projects/openstack-security/)
Common Questions (http://openstack.org/projects/openstack-faq/)
Blog (http://openstack.org/blog/)

Community

User Groups (http://openstack.org/community/)
Events (http://openstack.org/community/events/)
Jobs (http://openstack.org/community/jobs/)
Companies (http://openstack.org/foundation/companies/)
Contribute (https://wiki.openstack.org/wiki/How_To_Contribute)

Documentation

OpenStack Manuals (http://docs.openstack.org)
Getting Started (http://openstack.org/software/start/)
API Documentation (http://developer.openstack.org)
Wiki (https://wiki.openstack.org)

Branding & Legal

Logos & Guidelines (http://openstack.org/brand/)
Trademark Policy (http://openstack.org/brand/openstack-trademark-policy/)
Privacy Policy (http://openstack.org/privacy/)
OpenStack CLA (https://wiki.openstack.org/wiki/How_To_Contribute#Contributors_License_Agreement)

Stay In Touch

(https://[illegible]StackFoundation)