vpsAdminOS

Since OpenVZ is dead and new distributions aren't supporting it, we had to find a way to upgrade our kernel, which meant choosing a different virtualization technology. Linux kernel now has some support for containers, so we've decided to stick with that. Next, we needed some distribution that we could use on nodes to serve as hypervisors, as a replacement of Scientific Linux 6 with OpenVZ kernel. We've chosen NixOS, which allows you to declare the system and its configuration and then reproducibly build it. And since we have a bit specific requirements, we've created our own distribution on top of NixOS.

vpsAdminOS is based on NixOS and not-os. It's a live distribution serving as a hypervisor for container virtualisation. Its as capable as OpenVZ Legacy was in its time. We have our own userspace tools to manage containers called osectl, which internally uses LXC. vpsAdminOS naturally integrates with vpsAdmin, our administration interface with web interface, which you're all using to manage your VPS. However, vpsAdminOS is meant to be fully usable even on its own, as a replacement to OpenVZ Legacy deployments. If you have some OpenVZ servers and would like a newer system, you can consider vpsAdminOS. We also have scripts to help with migration of OpenVZ containers onto vpsAdminOS.

Migration from OpenVZ to vpsAdminOS

The upgrade of our infrastructure with all VPSes to vpsAdminOS is divided into several phases:

1. Developing vpsAdminOS into something usable
2. Integration with vpsAdmin
3. Opening of a staging environment with vpsAdminOS
   1. Testing, fixing bugs, implementing missing features, preparing for production
4. New production nodes are using vpsAdminOS (we're here)
5. Gradual migration of all VPS from OpenVZ nodes to vpsAdminOS, one node after another
6. End of story

What does it mean for members

We're trying to make the migration to vpsAdminOS as seamless as possible, so that one day your VPS will stop on the OpenVZ node and will start on vpsAdminOS node a while later, without you having to do anything. However, it depends on what programs you're running and what configuration changes you have made. That's why we recommend for everyone to try VPS on vpsAdminOS in the staging environment, so that we can find and solve problems before we start migration production VPS.

Changes in VPS behaviour

General
Changes regarding VPS, but independent on the distribution used:

- `/proc/stat` reports only user (including system) and idle
- Swap is not shown in `/proc/meminfo`

**Debian/Ubuntu/Alpine**

- Network is configured using `ip` from `iproute2`, you no longer need `ifconfig` from `net-tools`
- If there is a directory called `/etc/network/interfaces.d`, it is sourced before `/etc/network/interfaces.tail`

**Behaviour changes in vpsAdmin**

- NAS and snapshots are not accessed using vpsAdmin mounts, but NFS exports
- IP address management is split into routed and interface addresses
- Reinstalling VPS on vpsAdminOS no longer deletes subdatasets and does not reset its configuration to the initial state, e.g. VPS features remain as they were.
- VPS features: bridge, iptables and NFS aren't configurable, they're always on.
- It is possible to change the network interface name within the VPS in VPS details page.

**Staging environment**

In order for all members to test VPS on vpsAdminOS, we've created so called staging environment. It's similar to playground, where everyone can create a VPS. When creating a VPS, just select location Staging and deselect Keep platform. The VPS will be created on a vpsAdminOS node.

It's terms of use are similar to playground VPS, only it can be a bit rougher, like unplanned outages and reboots when we need to fix something. Everyone can use up to 8 CPUs, 4 GB RAM, 120 GB disk space, 4 public IPv4 addresses and 32 IPv6 /64 addresses. You can split these resources among 4 VPS.

You can either create a new VPS or clone an existing production VPS. All mounts are removed when cloning, because NAS isn't accessible as of yet, see user namespaces.

**Features**

Features can be turned on/off individually. When any change is made, the VPS restarts.
- Docker (experimental) - Enables support for Docker.
- FUSE - “Filesystem in Userspace” Enables the kernel module to allow non-privileged users create their own file systems.
- KVM - “Kernel-based Virtual Machine” Enables KVM for hardware support of virtualization.
- LXC nesting - “Linux Containers” Enables nested LXC containers.
- PPP - “Point-to-Point Protocol” Enables communications protocol used to establish a direct connection between point-to-point links.
- TUN/TAP - “TUN routing/TAP bridging” Enables the creation of virtual interfaces that are then bridged.

We recommend only setting the features that your really need.

**More about vpsAdminOS**

- [https://vpsadminos.org](https://vpsadminos.org)
- [https://github.com/vpsfreecz/vpsadminos](https://github.com/vpsfreecz/vpsadminos)
- IRC chat.freenode.net #vpsadminos

**Reporting bugs and ideas**

Choose at your own discretion:

- IRC: #vpsfree and #vpsadminos on chat.freenode.net
- podpora@vpsfree.cz
- vpsAdminOS issues: [https://github.com/vpsfreecz/vpsadminos/issues](https://github.com/vpsfreecz/vpsadminos/issues)
- vpsAdmin issues: [https://github.com/vpsfreecz/vpsadmin/issues](https://github.com/vpsfreecz/vpsadmin/issues)