

OpenBSD and Kubernetes

LinuxDays 2025

Jiří Navrátil

<https://www.navratil.cz>

Me in 2025

- Authoring a book on OpenBSD, virtualization, containerization, and Kubernetes
- Self-hosting domains, emails, and web sites using OpenBSD on physical hardware
- Personal laptop¹ running OpenBSD
- Company laptop running Linux
- Unix lectures at the University of Ostrava utilizing OpenBSD on physical hardware
- System Architect at CGI, specializing in Kubernetes

¹ https://openbsd.navratil.info/openbsd_on_laptop/tuxedo_aura_14_gen3/

Me and this topic at LinuxDays

- My first talk here. Thank you for the invitation!
- OpenBSD?
It's BSD, closer to UNIX®, but together with Linux belongs to the Unix-like family
- Alpine Linux?
It's Linux and notably the one from few, which doesn't use GNU
- Kubernetes?
Linux is the primary and most robust platform for running Kubernetes

Me and OpenBSD

- 2002 - first OpenBSD deployment
- 2025 - most machines running OpenBSD
- secure², reliable, and stable
- free³ and open-source
- simple configuration with examples
- comprehensive documentation (FAQ, manual pages, source code, papers)
- daemons and services configured and controlled via rcctl(8)
- many supported platforms⁴ (i386, amd64, armv7, arm64, sparc64, ...)
- support from community and developers

² <https://www.openbsd.org/innovations.html>

³ <https://www.openbsd.org/policy.html>

⁴ <https://www.openbsd.org/plat.html>

Already benefiting from OpenBSD?

- if you use ssh, then it might be **OpenSSH** from OpenBSD

```
ssh -V  
OpenSSH_10.0, LibreSSL 4.1.0
```

- **hackathon** type of events come from OpenBSD (or Sun Microsystems)
- many firewalls use **pf** from OpenBSD
- ~~OpenSSL~~ → LibreSSL
- ~~sudo~~ → doas
- ~~screen~~ → tmux
- ~~rsync~~ → opensync⁵
- ~~git~~ → Got: <https://gameoftrees.org/>

⁵ Search or ask for „opensync on macOS“

Why Kubernetes?

- Widely adopted for **container orchestration**
- Easily manage and scale applications
- Key to building resilient, scalable **cloud-native** applications
- **Automated** deployment and management
- Flexibility to deploy across **various cloud environments**
- High demand for Kubernetes skills in the **job market**

„It's so good, it feels almost too good to be true.“

Where Kubernetes?

- **EKS** Amazon Elastic Kubernetes Service
- **GKE** Google Kubernetes Engine
- **AKS** Microsoft Azure Kubernetes Service
- **OKE** Oracle Container Engine for Kubernetes
- **IKS** IBM Cloud Kubernetes Service
- **OpenShift** by Red Hat, an IBM Company
- **LKE** Linode Kubernetes Engine (now Akamai Connected Cloud)
- **DOKS** DigitalOcean Kubernetes
- Scaleway Kubernetes Kapsule
- **ACK** Alibaba Cloud (Aliyun) Managed Clusters
- **BCE** Baidu Container Engine
- Huawei Cloud Container Engine
- **TKE** Tencent Kubernetes Engine
- **MKS** OVHcloud Managed Kubernetes Service

Which company do you trust, and what price do you consider affordable?

Why Kubernetes on OpenBSD

- I trust OpenBSD
- I prefer owned hardware for security and cost-effectiveness
- My servers are hosted at AVONET, a company owned by a friend of mine
- I'm using this Kubernetes setup for learning, writing and teaching

How Kubernetes on OpenBSD

- I **control** Kubernetes clusters directly from OpenBSD via kubectl

```
$ pkg_info kubectl | head -9
Information for inst:kubectl-1.29.5
```

Comment:

command line tool for controlling Kubernetes clusters

Description:

The Kubernetes command-line tool, kubectl, allows you to run commands against Kubernetes clusters. You can use kubectl to deploy applications, inspect and manage cluster resources, and view logs.

- I need a Linux VM⁶ to **deploy** Kubernetes cluster on OpenBSD⁷
- I will use the small, simple, and secure Alpine Linux⁸

⁶ K8s needs Linux kernel features to enable containerization, resource management, and isolation.

⁷ The same apply for macOS.

⁸ It uses musl, BusyBox, and OpenRC.

Layers

- OpenBSD 7.7 amd64 on my laptop
- vmd(8) for hosting virtual machine (host)
- Alpine Linux as the virtual machine (guest)
- k0s as lite version of Kubernetes

Host Requirements

- installed recent OpenBSD version⁹
- HW support for virtualisation

```
$ dmesg | egrep '(VMX/EPT|SVM/RVI)'  
vmm0 at mainbus0: VMX/EPT
```

- enough RAM and free storage

```
$ free
```

	total	used	free
Mem:	31.7G	30.5G	1.2G
Swap:	8.0G	0B	8.0G

```
$ df -h /home
```

Filesystem	Size	Used	Avail	Capacity	Mounted on
/dev/sd1h	3.6T	3.1T	274G	93%	/home

- root (doas) privileges
- IPv4¹⁰ network

⁹ I use OpenBSD 7.7 amd64 here

¹⁰ IPv6 is not demonstrated for simplification

Local DNS resolver in OpenBSD

I'm running locally unwind(8).

```
$ cat /etc/resolv.conf
nameserver 127.0.0.1
nameserver ::1
lookup file bind
```

```
$ host www.navratil.cz 127.0.0.1
Using domain server:
Name: 127.0.0.1
Address: 127.0.0.1#53
Aliases:
```

```
www.navratil.cz is an alias for navratil.cz.
navratil.cz has address 217.115.249.166
navratil.cz has IPv6 address 2a00:6500:0:1500::13
navratil.cz mail is handled by 10 yavanna.navratil.cz.
```

This is optional. You may also use some DNS resolver outside your machine.

Enable packet forwarding and setup NAT

- Enable IPv4 packet forwarding.

```
echo 'net.inet.ip.forwarding=1' | doas tee -a /etc/sysctl.conf
doas sysctl net.inet.ip.forwarding=1
```

- Extend pf.conf

- Replace localhost with IPv4 of external DNS resolver, if needed.
- The Shared Address Space address range 100.64.0.0/10 is from RFC6598

```
match out on egress from 100.64.0.0/10 to any nat-to (egress)
pass in proto { udp tcp } from 100.64.0.0/10 to any port domain \
    rdr-to localhost port domain
```

- Test and apply firewall changes.

```
doas pfctl -n -f /etc/pf.conf
doas pfctl -f /etc/pf.conf
```

Virtual Machine Daemon in OpenBSD

- The vmd(8) is part of system from OpenBSD 5.9

```
$ which vmd  
/usr/sbin/vmd
```

- Enable and start vmd(8).

```
$ doas rcctl enable vmd  
$ doas rcctl start vmd  
vmd(ok)
```

Alpine Linux - Prepare GPG Keys

```
$ cd /tmp
$ wget https://www.alpinelinux.org/keys/ncopa.asc
$ gpg --import ncopa.asc
:
$ gpg --edit-key 0x293ACD0907D9495A
:
gpg> trust
:
Your decision? 5
Do you really want to set this key to ultimate trust? (y/N) y

pub  rsa4096/0x293ACD0907D9495A
     created: 2014-12-10  expires: never           usage: SC
     trust: ultimate     validity: unknown
sub  rsa4096/0x9E2CE29C63FE7A06
     created: 2014-12-10  expires: never           usage: E
[ unknown] (1). Natanael Copa <ncopa@alpinelinux.org>
:
gpg> quit
```

Download and Verify Alpine Linux

- Download **virtual** flavor Alpine Linux ISO, checksum and gpg files
- URLs are at Alpine Linux home page¹¹ under Downloads

```
$ cd /tmp
$ wget https://dl-cdn.alpinelinux.org/alpine/v3.22/releases/x86_64/alpine-virt-3.22.0-x86_64.iso
$ wget https://dl-cdn.alpinelinux.org/alpine/v3.22/releases/x86_64/alpine-virt-3.22.0-x86_64.iso.sha256
$ wget https://dl-cdn.alpinelinux.org/alpine/v3.22/releases/x86_64/alpine-virt-3.22.0-x86_64.iso.asc
$ du -h alpine-virt-3.22.0-x86_64.iso
64.0M    alpine-virt-3.22.0-x86_64.iso
```

- Integrity verification

```
$ sha256 -c alpine-virt-3.22.0-x86_64.iso.sha256
(SHA256) alpine-virt-3.22.0-x86_64.iso: OK
```

- Authenticity and Integrity verification

```
$ gpg --verify alpine-virt-3.22.0-x86_64.iso.asc alpine-virt-3.22.0-x86_64.iso
gpg: Signature made Fri May 30 17:09:31 2025 CEST
gpg:                using RSA key 0482D84022F52DF1C4E7CD43293ACD0907D9495A
gpg: Good signature from "Natanael Copa <ncopa@alpinelinux.org>" [ultimate]
Primary key fingerprint: 0482 D840 22F5 2DF1 C4E7  CD43 293A CD09 07D9 495A
```

¹¹ <https://www.alpinelinux.org/>

Install Alpine Linux 1/3

- Create 4GB disk for the virtual machine

```
$ mkdir $HOME/vm/01
$ cd $HOME/vm/01
$ vmctl create -s 4G disk.qcow2
vmctl: qcow2 imagefile created
$ du -h disk.qcow2
192K    disk.qcow2
```

- Start the virtual machine with the ISO as booting device

```
(cd /tmp && ln alpine-virt-3.22.0-x86_64.iso alpine.iso)
doas vmctl start -i 1 -L -r /tmp/alpine.iso -m 2G -d disk.qcow2 k0s
```

- Check the status of running VMs

```
$ vmctl show
```

ID	PID	VCPUS	MAXMEM	CURMEM	TTY	OWNER	STATE	NAME
1	48200	1	2.0G	152M	ttypj	root	running	k0s

Install Alpine Linux 2/3

- Connect to the console of the virtual machine

```
$ doas vmctl console k0s  
Connected to /dev/ttypj (speed 115200)
```

```
Welcome to Alpine Linux 3.22  
Kernel 6.12.31-0-virt on x86_64 (/dev/ttyS0)
```

```
localhost login: root  
Welcome to Alpine!
```

```
The Alpine Wiki contains a large amount of how-to guides and general  
information about administrating Alpine systems.  
See <https://wiki.alpinelinux.org/>.
```

```
You can setup the system with the command: setup-alpine
```

```
You may change this message by editing /etc/motd.
```

```
localhost:~#
```

Install Alpine Linux 3/3

■ Setup the Alpine Linux¹²

```
localhost:~# setup-alpine
Enter system hostname (fully qualified form, e.g. 'foo.example.org') [localhost] k0s
Which one do you want to initialize? (or '?' or 'done') [eth0]
Ip address for eth0? (or 'dhcp', 'none', '?') [dhcp]
Changing password for root
New password:
Retype password:
Which timezone are you in? (or '?' or 'none') [UTC] Europe/Prague
HTTP/FTP proxy URL? (e.g. 'http://proxy:8080', or 'none') [none]
Which NTP client to run? ('busybox', 'openntpd', 'chrony' or 'none') [busybox]
Enter mirror number or URL: [1] f
Setup a user? (enter a lower-case loginname, or 'no') [no]
Which ssh server? ('openssh', 'dropbear' or 'none') [openssh]
Allow root ssh login? ('?' for help) [prohibit-password]
Enter ssh key or URL for root (or 'none') [none] ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIM615yS+/5KditOmGtI+1FzAFT0umcI2Qb37I+dERVVn jiri@navratil.cz
Which disk(s) would you like to use? (or '?' for help or 'none') [none] vda
How would you like to use it? ('sys', 'data', 'crypt', 'lvm' or '?' for help) [?] sys
WARNING: The following disk(s) will be erased:
  vda  (4.3 GB 0x0b5d )
WARNING: Erase the above disk(s) and continue? (y/n) [n] y
Installation is complete. Please reboot.
k0s:~# poweroff
```

■ Check on host, that the VM has been properly stopped.

```
$ vmctl show
  ID      PID  VCPUS  MAXMEM  CURMEM  TTY      OWNER    STATE NAME
```

¹² Output is intentionally shortened

Use Alpine Linux 1/2

- Start the virtual machine

```
$ cd $HOME/vm/01
$ doas vmctl start -i 1 -L -m 2G -d disk.qcow2 k0s
vmctl: started vm 1 successfully, tty /dev/ttypj
```

- Connect to the console¹³ and login as root

```
$ doas vmctl console k0s
Connected to /dev/ttypt (speed 115200)

Welcome to Alpine Linux 3.22
Kernel 6.12.33-0-virt on x86_64 (/dev/ttyS0)

k0s login: root
Password:
```

- Set the message of the day

```
k0s:~# echo Alpine Linux for k0s on OpenBSD > /etc/motd
```

¹³ You may extend `vmctl start` command with `-c` to immediately connect to the console.

Use Alpine Linux 2/2

- Look for guest IP, logout and disconnect from console with ~.

```
k0s:~# ifconfig eth0 | grep inet
      inet addr:100.64.1.3 Bcast:0.0.0.0 Mask:255.255.255.254
      inet6 addr: fe80::fcel:bbff:fed1:faff/64 Scope:Link
k0s:~# exit
Welcome to Alpine Linux 3.22
Kernel 6.12.33-0-virt on x86_64 (/dev/ttyS0)

k0s login:
[EOT]
```

- Login to Alpine Linux from host via ssh

```
$ ssh root@100.64.1.3
The authenticity of host '100.64.1.3 (100.64.1.3)' can't be established.
ED25519 key fingerprint is SHA256:UkJGrLEEZEoA3EN6sD5S1HXLbN7xwsCw1rBzsxmVz+o.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '100.64.1.3' (ED25519) to the list of known hosts.
Alpine Linux for k0s on OpenBSD
k0s:~#
```

Install k0s 1/3

- Install curl package

```
k0s:~# apk add curl
```

- Download and install k0s

```
k0s:~# curl -sSf https://get.k0s.sh | sh
Downloading k0s from URL:
https://github.com/k0sproject/k0s/releases/download/v1.33.1+k0s.1/k0s-v1.33.1+k0s.1-amd64
k0s is now executable in /usr/local/bin
You can use it to complete the installation of k0s on this node,
see https://docs.k0sproject.io/stable/install/ for more information.
```

- Install single node k0s as a OpenRC service

```
k0s:~# k0s install controller --single
```

- Start the k0s service

```
k0s:~# k0s start
```

Install k0s 2/3

- Check k0s status

```
k0s:~# k0s status
Version: v1.33.1+k0s.1
Process ID: 1724
Role: controller
Workloads: true
SingleNode: true
Kube-api probing successful: true
Kube-api probing last error:
```

- use the Kubernetes kubectl included in k0s

```
k0s:~# k0s kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
k0s	Ready	control-plane	3m10s	v1.33.1+k0s

```
k0s:~# k0s kubectl top nodes
```

NAME	CPU(cores)	CPU(%)	MEMORY(bytes)	MEMORY(%)
k0s	13m	1%	639Mi	34%

Install k0s 3/3

- k0s use **containerd** as the container runtime

```
k0s:~# ls -ltr /etc/k0s/
total 8
-rw-r--r--    1 root    root                362 Jun 21 09:15 containerd.toml
drwxr-xr-x    2 root    root            4096 Jun 21 09:15 containerd.d
```

- OpenBSD configured TAP interface with 100.64.1.2/31 for point-to-point link

```
$ ifconfig tap0 | grep inet
    inet 100.64.1.2 netmask 0xffffffffe
```

- resolv.conf(5) in VM points to that IP

```
k0s:~# cat /etc/resolv.conf
nameserver 100.64.1.2
```

- generate kubeconfig file

```
$ mkdir $HOME/.kube
$ k0s kubeconfig admin > $HOME/.kube/config
$ chmod 600 $HOME/.kube/config
```


Install kubectl to VM

- Install kubectl binary as per instructions on Kubernetes site¹⁴

```
k0s:~# export VER=$(curl -L -s https://dl.k8s.io/release/stable.txt)
k0s:~# cd /tmp
k0s:/tmp# curl -LO "https://dl.k8s.io/release/${VER}/bin/linux/amd64/kubectl"
k0s:/tmp# curl -LO "https://dl.k8s.io/release/${VER}/bin/linux/amd64/kubectl.sha256"
k0s:/tmp# echo "$(cat kubectl.sha256)  kubectl" | sha256sum -c
kubectl: OK
k0s:/tmp# install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
```

- Check the kubectl binary version

```
k0s:~# kubectl version --client
Client Version: v1.33.2
Kustomize Version: v5.6.0
```

- get cluster nodes directly with kubectl binary

```
k0s:~# kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
k0s	Ready	control-plane	54m	v1.33.1+k0s

¹⁴ <https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/>

Create regular user in VM

■ Create regular user account¹⁵

```
k0s:~# adduser -g users -G wheel -s /bin/sh -u 5006 navratil
```

■ Add your public ssh key

```
k0s:~# su - navratil
k0s:~$ mkdir ~/.ssh && chmod 700 ~/.ssh
k0s:~$ cat >> ~/.ssh/authorized_keys << "EOF"
> ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIM615yS+/5KditOmGtI+1FzAFT0umcI2Qb37I+dERVVn
> jiri@navratil.cz
> EOF
k0s:~$ exit
```

■ Install and configure doas¹⁶

```
k0s:~# apk add doas
k0s:~# echo "permit nopass setenv { -ENV PS1= SSH_AUTH_SOCK } :wheel" \
> /etc/doas.d/doas.conf
k0s:~# chmod 0640 /etc/doas.d/doas.conf
```

¹⁵ Use your preferred uid and name.

¹⁶ The OpenBSD replacement for sudo.

Control k0s as regular user in VM

- Prepare copy of kubeconfig for regular user

```
k0s:~# su - navratil
k0s:~$ mkdir $HOME/.kube
k0s:~$ doas cp /root/.kube/config $HOME/.kube
k0s:~$ doas chown navratil $HOME/.kube/config
k0s:~$ chmod 600 $HOME/.kube/config
```

- Use kubectl in VM

```
k0s:~$ kubectl top nodes
```

NAME	CPU(cores)	CPU(%)	MEMORY(bytes)	MEMORY(%)
k0s	13m	1%	823Mi	43%

Control k0s directly from OpenBSD

- Download kubeconfig to host

```
$ mkdir -p $HOME/.kube/k0s
$ cd $HOME/.kube/k0s
$ sftp root@100.64.1.3:.kube
sftp> get config
sftp> exit
```

- Point¹⁷ KUBECONFIG to k0s in VM

```
$ export KUBECONFIG=$HOME/.kube/k0s/config
```

- Use kubectl from host

```
$ kubectl get nodes
NAME      STATUS    ROLES    AGE      VERSION
k0s       Ready     control-plane  64m      v1.33.1+k0s

$ kubectl get pods --all-namespaces
NAMESPACE   NAME                                     READY   STATUS    RESTARTS   AGE
kube-system  coredns-7494494fcf-2zgmk               1/1     Running   0           65m
kube-system  kube-proxy-w87tq                        1/1     Running   0           65m
kube-system  kube-router-7k8p7                      1/1     Running   0           65m
kube-system  metrics-server-7db8586f5-7fmbf         1/1     Running   0           65m

$ kubectl top nodes
NAME      CPU(cores)   CPU%    MEMORY(bytes)  MEMORY%
k0s       15m          1%      1238Mi         65%
```

¹⁷ I don't use default kubeconfig. I always decide via environment, which cluster I want to access.

Upgrade kubectl on host

- The OpenBSD kubectl version from packages is older¹⁸ than the k0s version

```
$ kubectl version
Client Version: v1.29.5
Kustomize Version: v5.0.1
Server Version: v1.33.1+k0s
WARNING: version difference between client (1.29) and server (1.33)
         exceeds the supported minor version skew of +/-1
```

- Delete the version installed from packages

```
$ doas pkg_delete kubectl
kubectl-1.29.5: ok
```

- Compile and install v1.33.1

```
$ cd $HOME/src
$ git clone https://github.com/kubernetes/kubernetes.git
$ cd kubernetes
$ git checkout v1.33.1
$ gmake kubectl
$ doas install -o navratil -g wheel -m 0755 \
  _output/local/go/bin/kubectl /usr/local/bin/
```

¹⁸ <https://kubernetes.io/releases/version-skew-policy/>

Use kubectl inside tmux on host

- You may use `tmux(1)` terminal multiplexer, when working with more clusters
- Choose `kubeconfig` and create separate named session for that `kubeconfig`

```
$ export KUBECONFIG=$HOME/.kube/k0s/config
$ tmux new-session -s k0s
$ echo $KUBECONFIG
/home/navratil/.kube/k0s/config
$ kubectl version
Client Version: v1.33.1
Kustomize Version: v5.6.0
Server Version: v1.33.1+k0s
```

- Note the session name `[k0s]` in `tmux` status bar
- You may detach from this session with the `C-b d` key strokes

```
[detached (from session k0s)]
```

- You may reattach to existing `k0s` session with

```
tmux attach -t k0s
```

K8s vs k0s

- The name **K8s** is a play on the word **Kubernetes**
- Both Kubernetes and K8s have meaning **full Kubernetes cluster**
- How was K8s constructed? See the Python code bellow

```
$ python3
>>> name="Kubernetes"
>>> print(f"{name[0]}{len(name)-2}{name[-1]}")
K8s
```

- Or if you prefer AWK

```
$ echo "Kubernetes" | awk \
'{printf "%s%d%s\n", substr($0,1,1), length($0)-2, substr($0,length($0),1)}'
K8s
```

- The name **k0s** is a play on the word **Kubernetes** too
- But there is 0 instead of 8 to stress "zero" complexity

Summary

- OpenBSD is **secure** Unix-like operating system
- You may **control** multiple K8s clusters from OpenBSD
- You may **deploy** light Kubernetes version to OpenBSD



Thank you for your attention

<https://linuxdays.navratil.cz/>