

Cloud Operations Using FreeBSD Compute

FreeBSD is an excellent choice when it comes to operating system diversification in cloud IaaS.

Jason Tubnor

The modern cloud has been around for about two decades. With Amazon coming up with the idea to provide retail compute services (EC2), AWS was formed to rent out this compute, and other services such as object storage and database. Other players, like Microsoft with Azure, Google with GCP, and Oracle with OCI, joined in to provide a diverse range of Tier 1 cloud infrastructure for their customers to consume.

Initially, Linux was all that was available, mainly due to technical constraints and para-virtualisation (pre-KVM), not allowing other operating system kernels to run.

However, things started to change. In 2008, two years after AWS EC2 became publicly available, FreeBSD developer Colin Percival started porting the FreeBSD operating system to AWS. Two years of hard work paid off and in December 2010, Colin was able to boot FreeBSD on AWS EC2 infrastructure.

Fast forward through another 5 years of refinement and FreeBSD became a standard offering on EC2 in April 2015.

Work didn't stop there. FreeBSD was further refined and became an offering in other Tier 1 provider marketplaces as well as some Tier 2 cloud providers. Unfortunately, there was a little pullback on a lot of operating system offerings within Tier 2 providers, not just FreeBSD. But now some make it easy to BYO ISO if you are prepared to support it yourself. These providers typically present devices using the VirtIO framework, which is baked into the FreeBSD base system, making it easy to boot a vanilla, generic system within these environments.

FreeBSD is an excellent choice when it comes to operating system diversification in cloud Infrastructure as a Service (IaaS) environment. Like with any technology, certain parts should be diversified to ensure service continuity, in the event that there is a bug in one of the platforms. FreeBSD gives you this security, having native implementation of device drivers and the recent addition of the Open Container Initiative (OCI) making it easy to use your existing tooling to manage workloads on FreeBSD IaaS compute instances.

Other features included with the cloud FreeBSD images are IPv6 support and in some cases ZFS or only UFS support. The latter maybe where you only need a basic FreeBSD appliance, but ZFS is available if you have 'Root on ZFS' needs for your application and management.

Another benefit of FreeBSD being available in the various cloud marketplaces is that it is free to use and build upon. There is no ongoing subscription cost, you can spin up one instance or a thousand. It all cost zero dollars except for compute. Some marketplaces do force you through the same workflow as subscription-based operating systems but there will be no cost when you hit the subscription button.

But wait, there's more. Are you looking at running on ARM64 architecture? FreeBSD has you covered there. Since 2024, providers like AWS have had ARM64 available for their Graviton ARM CPU, allowing users to choose a platform that is of a lower cost and power consumption. As more

organisations move to an environmentally sustainable model, all parts of their operations need to reduce their carbon footprint, another place that FreeBSD and ARM compute can assist with.

To bring up a FreeBSD compute instance, it can be done either via the web console or via the AWS CLI API tool.

FreeBSD desktop users have easy access to the CLI tools that can control tenancies on the various cloud vendors. Simply, install it from the FreeBSD package repository:

```
# pkg install py312-awscli
```

Once installed, authenticate using your AWS Management Console credentials:

```
$ aws login
```

At this point, you'll be authenticated for a period of time until you are automatically logged out due to inactivity.

It is expected that you have already defined your SSH public key, security group and subnets. This is required to minimise exposure of your instance once it is created and booted. Additional information is needed to complete the EC2 creation: the instance type, how many you need, and which region you want it located in.

To create your instance issue:

```
$ aws ec2 run-instances --image-id ami-[freebsd AMI ID] --count 1 --instance-type t4g.micro --region ap-southeast-2 --key-name key-[my SSH public key ID] --security-group-ids sg-[security group ID] --subnet-id subnet-[subnet ID]
```

Depending on your security group and subnet, if it was set to be publicly available, you'll be able to SSH to it directly from your workstation:

```
$ ssh -I .ssh/id_awsec2 ec2@mynewhost.example.com
```

Cloud-init is used to set up your host on boot with a minimum to get things going. Access to root is as simple as invoking 'su'. Note: securing the instance is out of scope for this article.

FreeBSD compute can interact with other PaaS offerings by the cloud vendors. Multiple volumes of block storage can be attached to compute for ZFS storage needs. The FreeBSD NFS client can be used to attach flexible file systems for access to consumption-based file storage. The FUSE file system can attach S3 object storage to instances. Or just use the network for your application on the FreeBSD instance to connect to a managed database. FreeBSD provides a base of only the components you need to create a system that has a low attack surface and minimal moving parts, which also reduces costs of each host you commission – which can add up when you need hosts by the thousand for large-scale project.

This article has only scratched the surface of the functionality and flexibility of choosing FreeBSD as a compute operating system but hopefully has given you some ideas when you need to deploy the “Next Big Thing” into the cloud.

Jason Tubnor has over 28 years of IT industry experience in a vast range of disciplines and is currently the ICT Senior Security Lead at Latrobe Community Health Service (Victoria, Australia). Discovering Linux and Open Source in the mid 1990s, then being introduced to OpenBSD in 2000, Jason has used these tools to solve various problems in organizations that cover different industries. Jason is also a co-host on the [**BSDNow Podcast**](#).

Copyright © 2026 Held by Owner/Author. Publication Rights Licensed to The FreeBSD Journal. This work is licensed under a [**Creative Commons Attribution International 4.0 License**](#).

3980 Broadway Street
STE #103-107
Boulder CO 80304
1-720-207-5142
info@freebsd.foundation.org

