



Coming to AWS re:Invent? Meet CAST AI in booth #2346 or book a time for a 1-on-1 chat [here](#)

KUBERNETES

COMPARISON

INSIDER

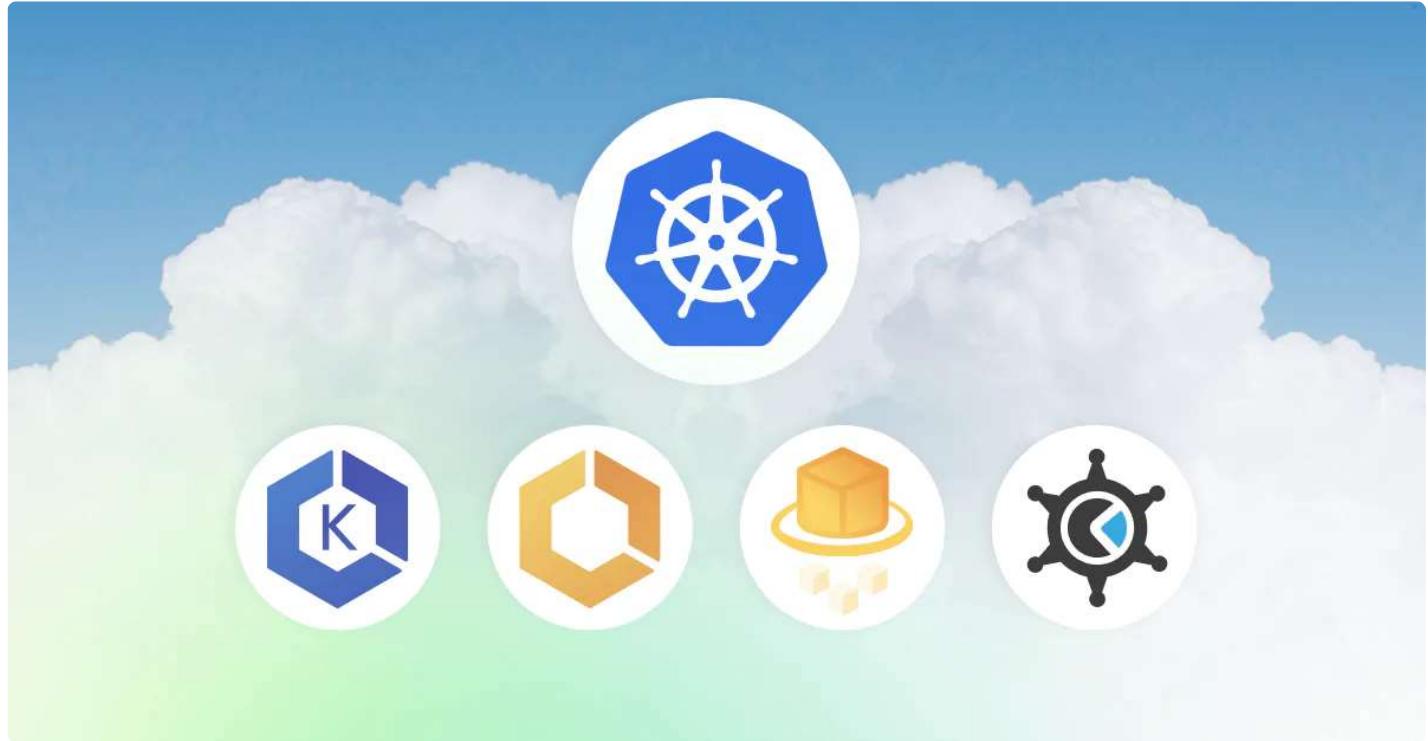
AWS EKS vs. ECS vs. Fargate vs. Kops: Where to Manage Your Kubernetes in 2022?



Žilvinas Urbonas

January 13, 2022 · 20 min read

Bonus Material: [1-page container service comparison table for AWS \(printable\)](#).



[ON THIS PAGE](#)

We use cookies to provide you with the best browsing experience on our website.

Ok

[Privacy policy](#)



We all love containers for their scalability. But that might easily become your overhead if you end up managing a large cluster.

This is where container orchestration comes in. When operating at scale, you need a platform that automates all the tasks related to the management, deployment, and scaling of container clusters.

There's a reason why almost **90% of containers** are orchestrated today.¹

If you're using Kubernetes on AWS, there are several options you can choose from:

- Elastic Container Service (ECS),
- Elastic Kubernetes Service (EKS),
- AWS Fargate,
- Kops for AWS.

Read on to find out which one is **the best match** for your workloads.

And if you know what's what in the world of AWS Kubernetes, you could still probably use a few best practices to reduce your cloud bill.

Note: The world of cloud technology changes rapidly, and we update this article regularly to keep up. Last update: 14.01.2022.

What is Elastic Container Service (ECS)?

AWS EKS vs. ECS vs. AWS Fargate: Container Orchestration ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



ECS with EC2 instances

In this model, containers are deployed to EC2 instances (VMs) created for the cluster. ECS manages them together with tasks that are part of the task definition.

Pros:

- You have full control over the type of EC2 instance used here. For example, you can use a GPU-optimized instance type if you need to run training for a machine learning model that comes with unique GPU requirements.
- You can take advantage of spot instances that reduce cloud costs by up to 90%.

Cons:

- You're the one responsible for security patches and network security of the instances, as well as their scalability in the cluster (but thankfully, you can use Kubernetes autoscaling for that).

Cost: You're charged for the EC2 instances run within your cluster and VPC networking.

ECS with AWS Fargate

In this variant, you don't need to worry about EC2 instances or servers anymore. Just choose the CPU and memory combo you need, and your containers will be deployed there.

Pros:

- No servers to manage.
- AWS is in charge of container availability and scalability. Still, better select the right CPU and memory – otherwise, you risk that your applica... [ON THIS PAGE](#)

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



over the networking layer (and you might need that in some scenarios).

Cost: You're charged based on the CPU and memory you select. The amount of CPU cores and GB determines the cost of running your cluster.

The screenshot shows a comparison chart for AWS Elastic Container Service (ECS) and AWS Fargate. It includes two main sections: 'EC2 instances' and 'With AWS Fargate'. Each section has a 'Pros' list and a 'Cons' list. A yellow hexagonal icon is positioned above the 'With AWS Fargate' section.

Category	EC2 instances	With AWS Fargate
Pros	<ul style="list-style-type: none">Full control over instance typeYou can use spot instances to drive costs down by up to 90%	<ul style="list-style-type: none">No servers to manageAWS takes care of availability and scalabilityYou can use Fargate Spot for spot instances
Cons	<ul style="list-style-type: none">You're the one responsible for security patches, network security, and instance scalability	<ul style="list-style-type: none">Supports only one networking mode - AWS VPC

Read More On <https://cast.ai/blog>

What is Elastic Kubernetes Service (EKS)?

EKS is a service that provides and manages a Kubernetes control plane on its own. You have no access to the master nodes on EKS since they're under a special AWS account.

To run a Kubernetes workload, EKS establishes the control plane and Kubernetes API in your managed AWS infrastructure and you're good to go.

At this point, you can deploy workloads using native K8s tools like kubectl, Kubernetes Dashboard, Helm, and Terraform.

[ON THIS PAGE](#)

We use cookies to provide you with the best browsing experience on our website.

[Ok](#) [Privacy policy](#)



Advantages

- No need to install, operate, and maintain your own Kubernetes control plane
- Easily runs Kubernetes tooling and plugins
- Automates load distribution and parallel processing really well
- Easy integration of Kubernetes assets
- Smooth migration
- Supports EC2 spot instances

Challenges

- Deployment requires expert configuration
- Pick EKS if you're ready to handle the scalability level of Kubernetes

Read More On <https://cast.ai/blog>

Advantages of AWS EKS

- You **don't have to install, operate, and maintain** your Kubernetes control plane.
- EKS allows you to **easily run tooling and plugins** developed by the Kubernetes open-source community.
- EKS **automates load distribution and parallel processing** better than any DevOps engineer could.
- Your Kubernetes assets integrate seamlessly with AWS services if you use EKS.
- EKS uses VPC networking.
- Any application running on EKS is compatible with one running in your existing Kubernetes environment. You can migrate to EKS without applying any changes to the code.
- Supports **EC2 spot instances** using managed node groups that follow spot best practices and allow some pretty great cost savings.

Download a 1-page comparison to weigh EKS, ECS, Fargate, and Kops side by side

[ON THIS PAGE](#)

We use cookies to provide you with the best browsing experience on our website.

[Ok](#) [Privacy policy](#)



AWS solves this problem with Fargate by taking over the management of that underlying server.

Instead of doing all the tasks yourself – from booting a server and installing the agent to making sure that it's up-to-date – you can simply create a cluster and add your workload to it. AWS will add pre-configured servers to the “pool” automatically to support your requirements.

Today, 32% of AWS container environments run on Fargate.²

AWS Fargate



 Advantages	 Challenges
<p>AWS takes over the server management:</p> <ul style="list-style-type: none">▪ Creating a cluster▪ Adding your workload to it▪ AWS will add pre-configured servers to the pool automatically to support your requirements	<ul style="list-style-type: none">▪ Not a good fit for highly regulated environments▪ Supports only one networking mode - AWS VPC▪ You can set few controls over how it works

Read More On <https://cast.ai/blog>

Here are a few things you should know about Fargate before jumping on the Fargate-managed bandwagon:

[ON THIS PAGE](#)

We use cookies to provide you with the best browsing experience on our website.

[Ok](#) [Privacy policy](#)



Fargate allocates resources automatically, which you can't control over time. This is good for **workloads**. This can easily lead to uncontrolled cost rise if you fail in close monitoring (for example, in R&D environments). One way to deal with that is through self-hosting and creating limited-capacity clusters.

What is Kops?

Kops is short for Kubernetes Operations (also spelled kops or kOps), a set of tools AWS offers for installing, operating, and deleting Kubernetes clusters.

You can also use it to roll out an upgrade of an older version of Kubernetes to a new one and manage cluster add-ons. After creating a cluster, you can use the usual [kubectl CLI](#) to manage resources.

To quote the [documentation](#):

“

kops will not only help you create, destroy, upgrade and maintain production-grade, highly available, Kubernetes cluster, but it will also provision the necessary cloud infrastructure.

Key features of Kops:

- Kops simplifies Kubernetes cluster setup and management, especially in comparison to the amount of effort needed to manually set up master and worker nodes.
- It automates the provisioning of Highly Available Kubernetes clusters.
- The tool manages Route53, AutoScaling Groups, ELBs for many different tasks from master and node bootstrapping to the API server and rolling updates to your cluster.

WEBSITE | BLOG | PODCAST | NEWSLETTER | ADVERTISING | ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



Kops vs. EKS – Key similarities and differences

Quickly see all the four services side by side in this 1-page comparison table:

[Download the 1-page AWS container service comparison table](#)

Setting up a Kubernetes cluster

EKS

Setting up a cluster in EKS is relatively complicated and comes with some prerequisites. You need to set up and use the AWS CLI and aws-iam-authenticator, and set up IAM permissions and users, which adds to your workload.

EKS doesn't create worker nodes automatically, so you're also in charge of managing that process. You also need to make extra effort to set up EKS with CloudFormation or Terraform.

Kops

Kops does the job much faster. It's a CLI tool you need to install on your local machine together with kubectl. To get a cluster running, use the Kops create cluster command. The great thing about Kops is that it manages most of the AWS resources you need for your Kubernetes cluster.

Contrary to EKS, Kops creates master nodes as EC2 instances – you can always access them directly to make modifications around the networking layer, size, etc. You can monitor these nodes directly too.

Managing a Kubernetes cluster

[ON THIS PAGE](#)

We use cookies to provide you with the best browsing experience on our website.

[Ok](#) [Privacy policy](#)



- **Scaling up the cluster is simple** and based on adding more worker nodes. The control plane is fully managed, so there's no need to worry about adding or upgrading master node sizes when your cluster grows.
- **Carrying out the most common maintenance tasks** is easy in EKS – you can add worker nodes, replace them, terminate instances, and upgrade your setup with minimal disruption to your cluster.

Kops

- Kops lets you **create a cluster really fast**, but managing it is more complicated. There's much groundwork to cover for upgrading and replacing master nodes for new Kubernetes versions. Creating a new cluster in a new VPC with Kops might be easier than that.
- When it comes to **maintenance tasks** like adding, replacing, and upgrading worker nodes, Kops performs quite similarly to EKS. AutoScaling Groups come in handy here.

Security

EKS

- With EKS, you get to enjoy the Amazon EKS Shared Responsibility Model, meaning that you're not left alone in securing the control plane.
- The platform may be relatively new, but you can count on AWS' multiple years of security expertise.
- Note that EKS clusters have limited administrator access via IAM by default. Managing permissions with IAM is a piece of cake if you use other AWS services that rely on it as well.
- Setting up encrypted root volumes and private networking is easy.
- The AWS account you use for EKS doesn't have root access to the account. ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



and other services, you don't get that extra security support for the control plane.

- A cluster created with Kops gets private networking, encrypted root volumes, and security group controls. You can also set up IAM authentication like in EKS.
- Kops also creates security groups and combines them with private networking to make your cluster more secure.

AWS EKS vs. ECS – Similarities

1. They both have a layer of abstraction

EKS and ECS come with a layer of abstraction for containers called deployments (EKS) or tasks (ECS).

Their functionalities are quite similar. Both ECS and EKS have an abstraction called cluster – a combination of all the working components.

2. They allow a mix of AWS compute platforms

Whether you're running your containers with ECS and EKS, you can choose one or more AWS compute options:

- **EC2 Instances** – virtual machines that offer a wide range of options and capacities.
- **AWS Fargate** – for serverless applications.
- **AWS Outposts** – a fully managed service that offers the same AWS infrastructure, services, APIs, and tools for data centers or on-premises facilities to make hybrid setups consistent. Great for workloads that need low latency access to on-premises systems or local data processing.
- **AWS Local Zones** – a kind of AWS infrastructure deployment that locates compute, storage, database, and other services closer to a user. ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

[Ok](#) [Privacy policy](#)



latency, mobile edge applications.

How to choose the best VM type? Consider your selection across these four dimensions:

- performance,
- cost,
- availability,
- reliability.

Understanding the differences between all these options is essential because they come with different financial commitments and complexity in management.

How to choose the best VM instance for EKS/ECS?



Consider These 4 Factors:



Performance



Availability



Reliability



Cost

Read More On <https://cast.ai/blog>

3. You don't need to monitor or operate them

ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



4. They share the approach to security

To access services and resources securely, AWS provides the Identity and Access Management (IAM) solution. You can create users (and user groups) – and then assign permissions to them.

This control system is available in both ECS and EKS. For example, you can use IAM to limit who can access ECS tasks or Kubernetes workloads.

Compare EKS and ECS side-by-side in a 1-page comparison table

AWS EKS vs. ECS – Differences

Summary:

Pricing

- ECS – ECS is free of charge and you only pay for the compute costs. It's a good match for those who are starting to explore microservices and containers.
- EKS – the solution costs \$0.1 per hour per Kubernetes cluster (c. \$74 per month) + compute costs. Pick it if you're ready to handle Kubernetes scalability.

Deployment

- ECS – simple to deploy, no control plane, configuration, and deployment directly from the AWS management console. Requires less expertise and operational knowledge.
- EKS – more complex deployment as you need to configure and deploy pods via Kubernetes first. Requires expert skills.

ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



Networking

- ECS – limited number of ENIs per instance. Might not be enough to support all the containers you want running on a particular instance.
- EKS – greater flexibility, you can share an ENI between multiple pods and place more pods per instance.

Community support

- ECS – limited community assistance and corporate AWS support.
- EKS – plenty of community support and resources.

1. Pricing

In general, if you run ECS and EKS clusters on EC2 instances, you'll be paying for compute costs that depend on the instance type you pick and its running time.

ECS doesn't come with any additional charges, but EKS does.

EKS will charge you \$0.1 per hour per Kubernetes cluster. This amounts to c. \$74 per month, which doesn't seem like a lot. But the costs might add up quickly depending on your setup.

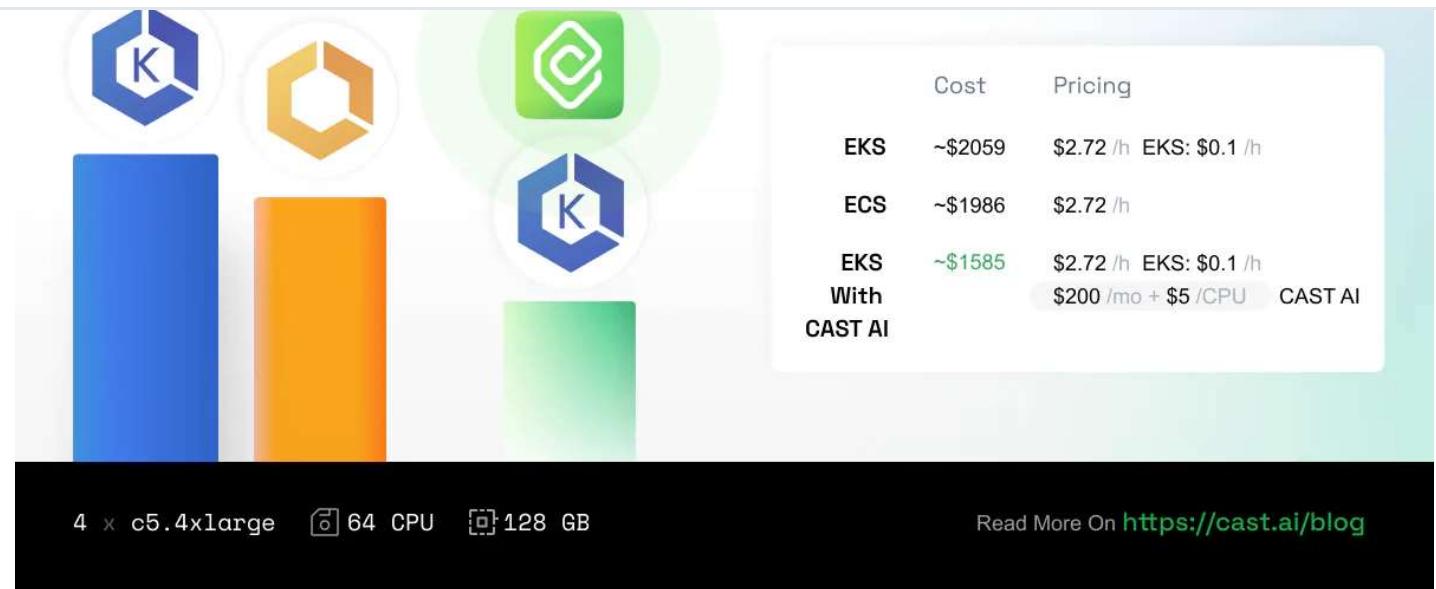
If you're just exploring microservices and containers, ECS is a better option. And if you're ready to handle the scalability level of Kubernetes, the \$74 extra on your bill isn't going to make much difference against your overall compute costs.

On the other hand, EKS enables you to use automated cost optimization with CAST AI that typically cuts around half of your cluster costs, here's how: [How to reduce your Amazon EKS costs by half in 15 minutes.](#)

ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



2. Deployment

You can set up both EKS and ECS from the AWS management console. But then things start looking different.

ECS is really simple to deploy. After all, it was designed to be a simple API for creating containerized workloads without any complex abstractions. You get no control plane, so once your cluster is set up, you can configure and deploy tasks directly from the AWS management console.

Deploying clusters on EKS is a bit more complex and requires expert configuration. You need to configure and deploy pods via Kubernetes first because EKS is just another layer for creating K8s clusters on AWS.

So compared to ECS, you need more expertise and operational knowledge to deploy and manage applications on EKS.

3. Multi cloud portability

ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



Kubernetes in EKS allows you to package your containers and move them to another platform quickly. ECS might lock you in.

So if you build an application in ECS, you're likely to encounter the vendor lock-in issue in the long run. And if you choose to design your application on Kubernetes, you can basically run it on any other Kubernetes cluster – from the cloud to on-premises.

4. Networking

In ECS, you use the *awsvpc* network that receives an elastic network interface (ENI) attached to the container instance hosting it.

You'll be looking at default limits to the number of network interfaces that can be attached to an EC2 instance (the primary network interface counts as one). Just to give you an idea, a c5.large instance may have up to 4 ENIs attached to it by default.



In ECS, the maximum number of ENIs you can assign varies by EC2 type. Even though AWS increased the limits, this might not be enough to support all the containers you want running on that particular instance.

Note that ECS supports launching container instances with larger ENI density using specific EC2 types. When you pick such a type and opt into the *awsvpcTrunking* account setting, you'll get some additional ENIs on newly launched container instances. So, you can place more tasks using the *awsvpc* network mode on every container instance.



With EKS, you can assign a dedicated network interface to a pod to improve security. All the containers inside that pod will share the internal network and public IP.

ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



5. Community support

The open-source Kubernetes rules over proprietary ECS here. The latter offers limited community assistance, so you can only count on the corporate support of AWS.

When running Kubernetes in EKS, you get:

- community support (Stack Overflow, Github issues),
- resources (from official training to online courses),
- community-maintained tools like kubectl extensions, Helm Charts, or Kubernetes Operators.

Should I choose EKS or ECS?

EKS	ECS
	
<ul style="list-style-type: none">■ When you need granular control over container placement■ When you need more networking modes■ When you want more control over your tooling	<ul style="list-style-type: none">■ When your DevOps resources are limited■ When you don't have time or resources to pick and choose add-ons■ When Kubernetes is too much

Read More On <https://cast.ai/blog>

When to choose EKS?

ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



- **When you need more networking modes** – ECS has only one networking mode available in Fargate. If your serverless app needs something else, EKS is a better choice.
- **When you want more control over your tooling** – ECS comes with a set of default tools. For example, you can use only Web Console, CLI, and SDKs for management. Logging and performance monitoring are carried out using CloudWatch, service discovery through Route 53, and deployments via ECS itself. If you don't like any of these tools, go for EKS.
- **When you want more control over your costs** – EKS allows you to control cloud expenses via dedicated automated platforms like CAST AI, which is beneficial if you're running a larger operation.

Already running K8s on ECS? Here's how to [migrate from ECS to EKS](#).

When to choose ECS?

Some teams might benefit from ECS more than EKS thanks to its simplicity. Here's when selecting ECS makes the most sense:

- **When you have limited DevOps resources** – ECS comes with a gentle learning curve, so if you're not prepared to re-architect your applications around Kubernetes concepts, ECS will be easier to adopt.
- **When you don't have time or resources to pick and choose add-ons** – Kubernetes and EKS, by extension, is a far more flexible tool. You can choose from many different add-ons available in the system. But each of them requires time, resources, and maintenance to make the most of it. ECS has only one option in each category – if it works for you, you're good to go.
- **When Kubernetes is too much** – If adopting K8s all at once is a little too much for you, ECS could be a good first step. It allows you to try your hand at containerization and move your workloads into a managed service without a huge upfront investment.

[ON THIS PAGE](#)

We use cookies to provide you with the best browsing experience on our website.

[Ok](#) [Privacy policy](#)



But if you'd like to have the freedom to integrate with the open-source Kubernetes community, putting in the energy and time into Kops is worth it. And there are plenty of cloud-native solutions to help you along the way.

To compare EKS, ECS, Fargate, and Kops side by side and have something to refer to in the future, download this 1-page comparison table:

[Download: 1-page table to compare AWS container services side-by-side](#)

References

[1] – [Datadog](#)

[2] – [Datadog](#)

FAQ

What is ECS and EKS in AWS?



What companies use EKS?



Is AWS ECS similar to Kubernetes?



What are the key differences between EKS, ECS, and Fargate?



[ON THIS PAGE](#)

We use cookies to provide you with the best browsing experience on our website.

Ok

[Privacy policy](#)



Can I use Fargate without ECS?



KUBERNETES

COMPARISON

INSIDER

Blog > AWS EKS vs. ECS vs. Fargate vs. Kops: Where to Manage Your Kubernetes in 2022?



23 COMMENTS

Oldest ▾

Gerald ⏲ 2021-06-17 5:50 AM

that answers my question, great article



Reply

teameks ⏲ 2021-07-20 11:10 AM

I prefer EKS, though it's nice to have a comparison



Reply

junior ⏲ 2021-07-21 11:16 PM

as a beginner I liked the FAQ in the end, helps to catch up with the article and fill the gaps. great post too, it really helps to find out the differences between eks ecs and fargate



Reply

ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok

Privacy policy



... ⏲ 2021-07-22 1:48 PM

it looks promising and all but honestly there is no way to know that this EKS analyzer won't mess up my cluster

Reply

Vidmantas CAST AI ⏲ 2021-07-23 10:40 AM

Reply to ...

Hi, thank you for your comment. CAST AI agent will use read-only mode to analyze your cluster and generate a report of possible savings. We encourage you to check service account permissions for the agent and ask away any questions you may have about it. Only when you decide that you want to try out cost optimization – our team of engineers will work with you to implement our tool and ensure that everything is running smoothly.

Reply

Antonio ⏲ 2021-08-11 7:27 AM

I guess moving from Fargate or EC2 to EKS is beneficial with a little learning curve. The future moves with kubernetes, so does the companies that shift to EKS

Reply

Mariusz ⏲ 2021-08-11 8:30 AM

If you are already using K8s, it would be sensible to go with everything that is tailored around it, so EKS it is.

Reply

ariel ⏲ 2021-08-12 9:52 AM

I love those faqs that you have at the end of some other artic ON THIS PAGE its a

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



care of k8s while working on other more important things

Reply

Donna ⏱ 2021-08-17 8:29 AM

We've been using ecs but the fact that we need to update the security patches and manage out scalability does eat a lot of our time out of the pocket and most of the time at bad moments, when we are near the crunching end of a feature or a quarter..

Reply

clint ⏱ 2021-08-17 8:51 AM

The fact that we don't have to monitor them as much as just deploying without any environment with kubernetes is astonishing and such a life saver..

Reply

Doni ⏱ 2021-08-18 3:11 AM

I can see that you guys only have EKS cluster attachment, pretty sad for me as I was looking for ECS cluster cost optimizations. I'll reach out to check on whether you guys are planning it for the future

Reply

Joakhim ⏱ 2021-08-18 3:24 AM

since we started out with Fargate, and since we scaled out a bit, I've been thinking lately about EKS.. but the fact of moving everything to a different environment and the labor itself to make everything work again as smoothly as possible is just a bit too much for us right now

Reply

ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



Reply

Kubernetes Services ⏲ 2021-09-16 6:01 AM

The tips you've provided are really helpful and I'll surely implement these in my article.

Reply

Dhrumil ⏲ 2021-10-08 3:33 AM

Very Interesting and precise comparison analysis. Helpful for engineers to carefully craft the solution architecture stack.

Reply

Adam Cole ⏲ 2022-01-10 7:15 AM

Thank you for an excellent article.

Reply

Adriel ⏲ 2022-01-17 8:40 PM

Thanks for the awesome article

Reply

Matt Warren ⏲ 2022-03-14 1:41 PM

Great article. Helped clarify how these similar AWS services fit.

Reply

ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



Reply

vimal ⓘ 2022-06-02 4:18 AM

This is a very well written article!

Reply

Jolokia ⓘ 2022-07-14 12:50 PM

All depends on your distributed application's use case. I find Kubernetes eats up too much of development time with the complexity. Plus multiple pods per node has shown to be a big issue for our use case, we need big nodes that run one container frequently (which we right size with EC2's offerings per app).

Both have tradeoffs, Kubernetes is ultra popular for good reason, but ECS nails it out of the park for our use case.

Whatever works best for you is the best option 😊

Leave a reply

Join the discussion

B I U S , </>

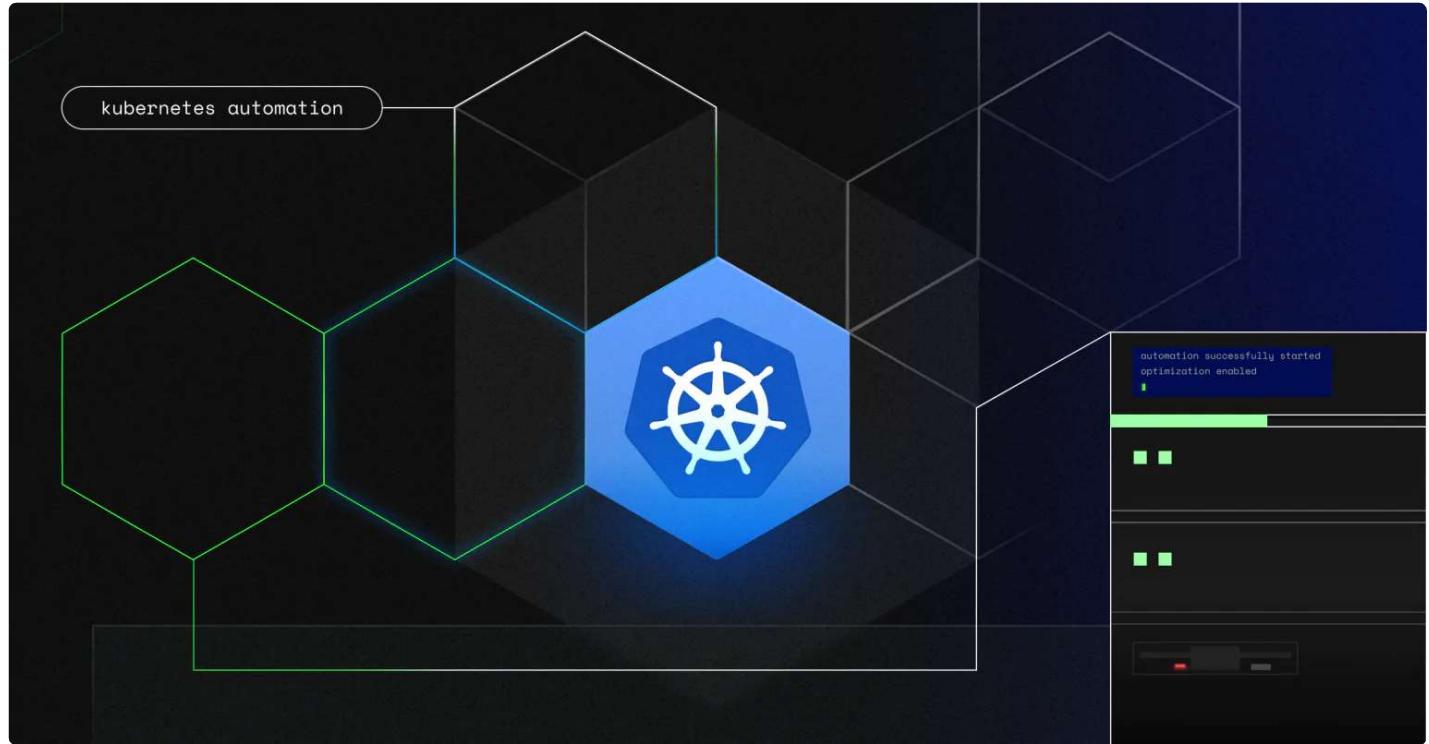
Name*

Email*



We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



COST OPTIMIZATION

Kubernetes Automation: 7 Reasons Why CAST AI Beats Savings Plans and Reserved Instances



Laurent Gil

November 23, 2022 · 5 min read

[ON THIS PAGE](#)

We use cookies to provide you with the best browsing experience on our website.

[Ok](#) [Privacy policy](#)



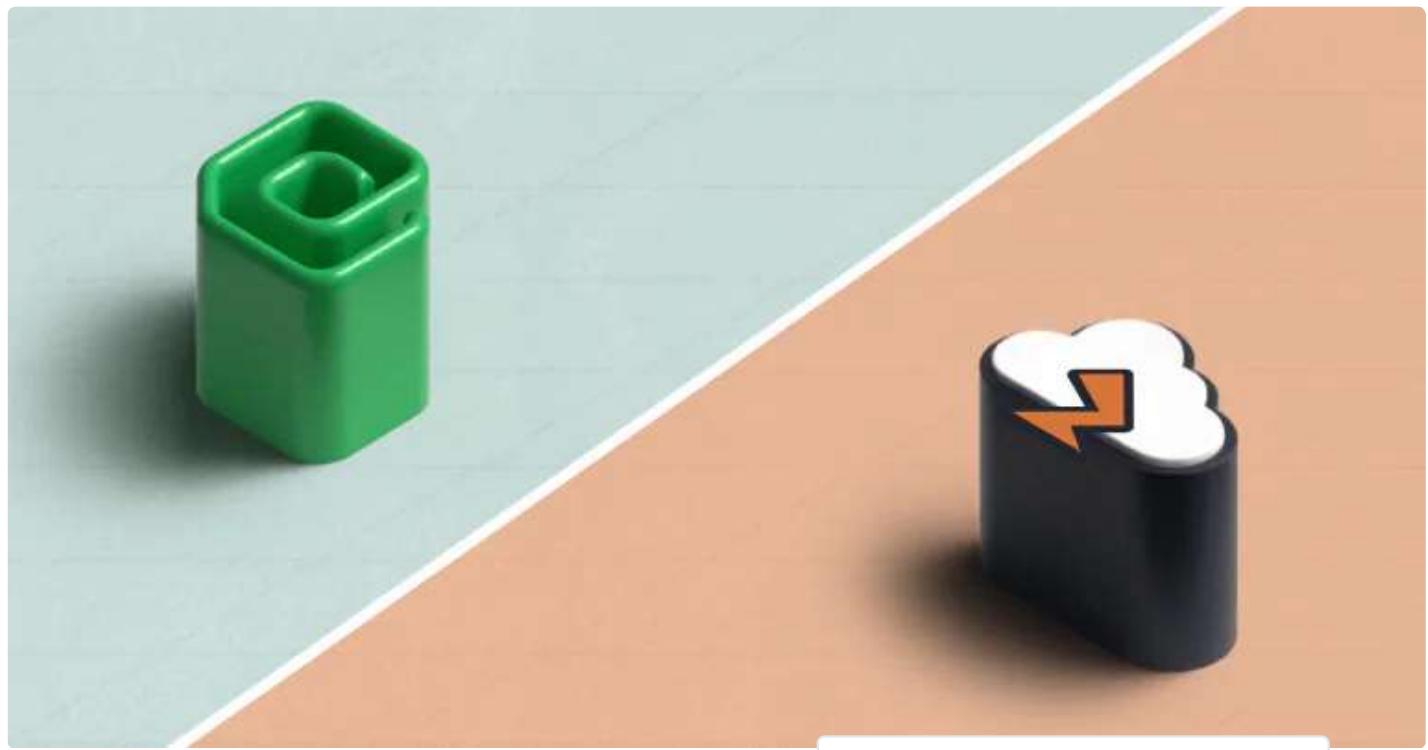
KUBERNETES

Kubernetes Cordon: How It Works And When To Use It



Laurent Gil

November 17, 2022 · 5 min read



ON THIS PAGE

We use cookies to provide you with the best browsing experience on our website.

Ok Privacy policy



Platform

Cloud Cost Optimization	EKS
Cloud Cost Savings Report	GKE
Cloud Cost Monitoring	KOPS
Cloud Security	AKS

Resources

Blog	Documentation
Events	Release Notes
Case Studies	Support

Company

About Us
Career 4
Slack
Press Releases
Partner Deal Registration

Pricing

Plans

[ON THIS PAGE](#)

We use cookies to provide you with the best browsing experience on our website.

[Ok](#) [Privacy policy](#)



U
S

[Privacy Policy](#) [Service Level Agreement](#) [Terms of Service](#) [EU Programs](#)
[Information Security Policy](#)

© Copyright 2022 CAST AI Group Inc. All rights reserved.

[ON THIS PAGE](#)

We use cookies to provide you with the best browsing experience on our website.

[Ok](#) [Privacy policy](#)