

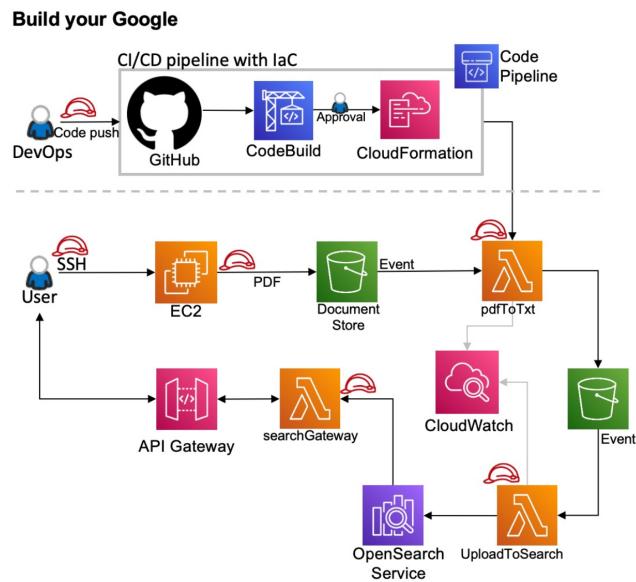
Patrick Jones, 8/4/2025..

PGPCCAUG4\_PATRICK\_JONES\_PROJECT2

## Deploying a search engine using AWS Manage Services

### Architecture

greatlearning



### Github

- setup private repository
- created classic token for aws pipeline authentication
- upload directories from local host

wendpmj / wendpmj-lextech-upload-to-search

**wendpmj-lextech-upload-to-search** · Private

main · 1 Branch · 0 Tags

Go to file · Add file · Code

**Patrick Jones and Patrick Jones** Fix control characters in PDFToTxt buildspec.yml · b917831 · yesterday · 66 Commits

PDFToTxt · Fix control characters in PDFToTxt buildspec.yml · yesterday

SearchFunction · Add simplified CloudFormation template for SearchFunction · yesterday

SearchGateway · Fix CloudFormation package template path in SearchGateway · yesterday

UploadToSearch · Add simplified CloudFormation template for UploadToSearch · yesterday

About · AWS Search Engine

Readme · Activity · 0 stars · 0 watching · 0 forks

Release

## Codebuild – for each function

- Automates the building and deployment of the four Lambda functions.
- CodeBuild executes the buildspec.yml for each specific Lambda
- Buildspec.yml runs aws cloudformation stack.

Developer Tools > CodeBuild > Build projects

Name	Source provider	Repository	Latest build status	Description	Last Modified
upload-to-search-build	AWS CodePipeline	-	Succeeded	Build project for UploadToSearch Lambda	1 day ago
search-gateway-build	AWS CodePipeline	-	Succeeded	Build project for SearchGateway Lambda	1 day ago
search-function-build	AWS CodePipeline	-	Succeeded	Build project for SearchFunction Lambda	1 day ago
pdftotxt-build	AWS CodePipeline	-	Succeeded	Build project for PDFToTxt Lambda	1 day ago

## Cloud Formation stacks

- Manages the API Gateway infrastructure for the project.
- Handles deployments related to the UploadToSearch function pipeline.
- Manages the deployment pipeline for the Search Gateway
- Manages the pipeline for deployments of the Search Function.

- Manages resources around the PDF to Text conversion function.

Stack name	Status	Created time	Description
<a href="#">UploadToSearch-pipeline-stack</a>	<span>CREATE_COMPLETE</span>	2025-08-03 10:38:08 UTC-0500	CI/CD Pipeline for UploadToSearch Lambda Function
<a href="#">SearchGateway-pipeline-stack</a>	<span>CREATE_COMPLETE</span>	2025-08-03 10:32:35 UTC-0500	CI/CD Pipeline for SearchGateway Lambda Function
<a href="#">SearchFunction-pipeline-stack</a>	<span>CREATE_COMPLETE</span>	2025-08-03 10:30:58 UTC-0500	CI/CD Pipeline for SearchFunction Lambda Function
<a href="#">PDFtoTxt-pipeline-stack</a>	<span>UPDATE_COMPLETE</span>	2025-08-03 10:29:20 UTC-0500	CI/CD Pipeline for PDFtoTxt Lambda Function

## Pipelines

- CodePipeline directs the process by triggering the CodeBuild project
- CodePipeline triggers when source code changes
- S3 bucket ([lextech-artifacts-bucket-219342442719](#))

```
//Store source code from GitHub
//Pass artifacts between Source → Build stages
```

Name	Latest execution status	Latest source revisions	Latest execution started	Most recent executions
<a href="#">pdftotxt-pipeline</a>	<span>Succeeded</span>	SourceAction - <a href="#">b9178317</a> : Fix control characters in PDFtoTxt buildspec.yml	1 day ago	<span>View details</span>
<a href="#">search-gateway-pipeline</a>	<span>Succeeded</span>	SourceAction - <a href="#">b9178317</a> : Fix control characters in PDFtoTxt buildspec.yml	1 day ago	<span>View details</span>
<a href="#">upload-to-search-pipeline</a>	<span>Succeeded</span>	SourceAction - <a href="#">b9178317</a> : Fix control characters in PDFtoTxt buildspec.yml	1 day ago	<span>View details</span>
<a href="#">search-function-pipeline</a>	<span>Succeeded</span>	SourceAction - <a href="#">b9178317</a> : Fix control characters in PDFtoTxt buildspec.yml	1 day ago	<span>View details</span>

## OpenSearch Domain: lextech-search

[https://vpc-lextech-search-sdr75chqyc4xsooutqraalhi3m-aos.us-east-1.on.aws/\\_dashboards](https://vpc-lextech-search-sdr75chqyc4xsooutqraalhi3m-aos.us-east-1.on.aws/_dashboards)

- Stores processed PDF content that was extracted and converted to text by the PDFtoTxt

## Lambda

The screenshot shows the AWS OpenSearch Service console with the domain 'lextech-search' selected. The 'General information' section displays the following details:

Name	Domain processing status	Version	OpenSearch Dashboards URL (dual stack)
lextech-search	Active	OpenSearch 2.19 (latest)	<a href="https://vpc-lextech-search-sdr75chqyc4xsooutqraalhi3m-aos.us-east-1.on.aws/_dashboards">https://vpc-lextech-search-sdr75chqyc4xsooutqraalhi3m-aos.us-east-1.on.aws/_dashboards</a>
Domain ARN	Configuration change status	Service software version	IPv4 URL
arn:aws:es:us-east-1:219342442719:domain/lextech-search	Completed	OpenSearch_2_19_R20250630-P4 (latest)	<a href="https://vpc-lextech-search-sdr75chqyc4xsooutqraalhi3m-aos.us-east-1.on.aws">https://vpc-lextech-search-sdr75chqyc4xsooutqraalhi3m-aos.us-east-1.on.aws</a>
Deployment option(s)	Cluster health	Domain endpoint v2 (dual stack)	Domain endpoint v1 (dual stack)
3-AZ with standby	Green	<a href="https://vpc-lextech-search-sdr75chqyc4xsooutqraalhi3m-aos.us-east-1.on.aws/_search">https://vpc-lextech-search-sdr75chqyc4xsooutqraalhi3m-aos.us-east-1.on.aws/_search</a>	<a href="https://vpc-lextech-search-sdr75chqyc4xsooutqraalhi3m-aos.us-east-1.on.aws/_search_v1">https://vpc-lextech-search-sdr75chqyc4xsooutqraalhi3m-aos.us-east-1.on.aws/_search_v1</a>

## EC2 instance

- Configure instance, add to same VPC as OpenSearch dashboard.
- Open inbound rule for rdp port 3389 to remote into EC2c instance for OpenSearch dashboard access. Restricted 3389 access to my public ip 108.91.0.80/32

The screenshot shows the AWS EC2 Instances page with one instance listed:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability zone
OpenSearchPortal	i-0365c8d91a14be02c	Running	t3.micro	3/3 checks passed	View alarms +	us-east-1c

## OpenSearch dashboard

- Once accessible through EC2 jump
- Create lextech-documents index to receive PdfToTxt documents for search

The screenshot shows the OpenSearch Dashboards interface with the 'Indexes' tab selected. On the left, a sidebar lists management categories: State management policies, Policy managed indexes, Data streams, Templates, Aliases, Rollup jobs, Transform jobs, and Notification settings. Below these are Snapshot Management, Snapshot policies, Snapshots, and Repositories. The main area displays a table titled 'Indexes (13)' with columns: Index, Health, Managed by..., Status, Total size, Size of pri..., Total docu..., Deleted do..., Primaries, and Replicas. The table lists several indexes, including 'lextech-documents' (Green, No, Open), '.ql-datasources' (Green, No, Open), '.plugins-ml-task' (Green, No, Open), '.plugins-ml-model-group' (Green, No, Open), '.plugins-ml-model' (Green, No, Open), and '.plugins-ml-config' (Green, No, Open). Each row includes a checkbox and a small icon.

## Create S3 bucket

Purpose: Process pdf's for Lambda functions.

- S3 bucket ([lextech-content-store-219342442719](#)) //pdf upload bucket

The screenshot shows the AWS S3 Buckets page. At the top, there are buttons for 'Create bucket' (orange), 'Copy ARN', 'Empty', and 'Delete'. A search bar and a 'Find buckets by name' input field are also present. The main table lists 'General purpose buckets (13)'. One bucket is visible: 'lextech-content-store-219342442719', which was created on August 2, 2025, at 08:24:22 (UTC-05:00) in US East (N. Virginia). The table has columns for Name, AWS Region, and Creation date. To the right of the table, a sidebar displays 'activity trends.' and an 'External access summary - new' section with a note about updated daily findings for identifying bucket permissions.

## Create PdfToTxt Lambda function (lextech-pdf-txt)

- Automatically triggers when a PDF file is uploaded to the S3 content bucket ([lextech-content-store-219342442719](#))
- Uses S3 event notifications to detect new .pdf files
- Extracts ALL text content from every page in the PDF
- Saves the extracted text to the intermediary S3 bucket ([lextech-intermediary-store-219342442719](#))
- CloudWatch logs created automatically

The screenshot shows the AWS Lambda Function Overview page for the function 'lextech-pdf-to-txt'. The top navigation bar includes the AWS logo, search bar, and user information (United States (N. Virginia) and Patrick @ 2193-4244-2719). Below the navigation is a breadcrumb trail: Lambda > Functions > lextech-pdf-to-txt. The main title is 'lextech-pdf-to-txt'. A note below the title states: 'This function belongs to an application. [Click here](#) to manage it.' On the left, there's a 'Function overview' section with tabs for 'Diagram' (selected) and 'Template'. The diagram shows the function name 'lextech-pdf-to-txt' with a box labeled 'Layers (0)' underneath. Below the function box is an 'S3' box with a plus sign and 'Add destination'. At the bottom of this section are buttons for '+ Add trigger' and '+ Add destination'. To the right of the diagram, there are sections for 'Description' (Extract text from PDF files), 'Last modified' (23 hours ago), 'Function ARN' (arn:aws:lambda:us-east-1:219342442719:function:lextech-pdf-to-txt), and 'Application' (PDFtoTxt-Lambda-Stack-us-east-1). Buttons for 'Export to Infrastructure Composer' and 'Download' are also present.

## Create S3 bucket (lextech-intermediary-store-219342442719) //pdf upload bucket

- Automatically triggers when a .txt file is uploaded to the intermediary S3 bucket ([lextech-intermediary-store-219342442719](#))
- Uses S3 event notifications to detect new .txt files created by the PDFtoTxt function
- Downloads the text file from the intermediary S3 bucket
- Uploads the document to OpenSearch using REST API

The screenshot shows the AWS S3 Buckets page. The top navigation bar includes the AWS logo, search bar, and user information (United States (N. Virginia) and Patrick @ 2193-4244-2719). Below the navigation is a breadcrumb trail: Amazon S3 > Buckets. The main title is 'General purpose buckets (13)'. A note below the title says: 'Buckets are containers for data stored in S3.' There is a search bar with placeholder 'Find buckets by name' and a 'Create bucket' button. A table lists the buckets, with one row selected: 'lextech-intermediary-store-219342442719'. The table columns are 'Name', 'AWS Region', and 'Creation date'. The selected bucket row shows 'US East (N. Virginia)', 'us-east-1', and 'August 2, 2025, 08:24:32 (UTC-05:00)'. To the right of the table, there is a 'External access summary - new' section with a note: 'Updated daily' and 'External access findings help you identify bucket permissions that allow public access or access from other AWS accounts.'

## UploadToSearch Function (lextech-upload-to-search)

- Second step in the document processing pipeline, takes the text files created by PDFtoTxt and makes them searchable by indexing them in OpenSearch.
- Converts text files into searchable documents in OpenSearch.
- Automatically triggers when a .txt file is uploaded to the intermediary S3 bucket ([lextech-intermediary-store-219342442719](#)).
- Uses S3 event notifications to detect new .txt files created by the PDFtoTxt function.

The screenshot shows the AWS Lambda function configuration page for 'lextech-upload-to-search'. The top navigation bar includes 'aws', a search bar, and account information for Patrick @ 2193-4244-2719. The main header shows the function name 'lextech-upload-to-search'. Below the header, a message states 'This function belongs to an application. [Click here](#) to manage it.' The 'Function overview' section contains a diagram showing the function's layers and triggers. The diagram shows 'lextech-upload-to-search' (Lambda icon) with a 'Layers' box containing '(1)' and an 'S3' trigger box. To the right of the diagram are buttons for 'Export to Infrastructure Composer' and 'Download'. The 'Description' field is set to 'Upload extracted text content to OpenSearch for indexing'. The 'Last modified' field shows '1 day ago'. The 'Function ARN' field contains 'arn:aws:lambda:us-east-1:219342442719:function:lextech-upload-to-search'. The 'Application' field is set to 'UploadToSearchStack'.

## SearchGateway Function (lextech-search-gateway)

- Serves HTML content when users visit the main API Gateway URL
- Acts as a web server delivering a complete HTML page with CSS and JavaScript
- No database or file storage, generates the interface dynamically

The screenshot shows the AWS Lambda Functions interface. The top navigation bar includes 'aws', a search bar, and account information ('United States (N. Virginia) Patrick @ 2193-4244-2719'). Below the navigation is a breadcrumb trail: 'Lambda > Functions > lextech-search-gateway'. The main title is 'lextech-search-gateway'. On the right, there are buttons for 'Throttle', 'Copy ARN', and 'Actions'. A message box states: 'This function belongs to an application. Click here to manage it.' Below this is a 'Function overview' section with tabs for 'Diagram' (selected) and 'Template'. The diagram shows a Lambda function named 'lextech-search-gateway' with two triggers: 'API Gateway' (2 triggers) and '+ Add trigger'. To the right of the diagram are details: 'Description: Serve search web interface', 'Last modified: 1 day ago', 'Function ARN: arn:aws:lambda:us-east-1:219342442719:function:lextech-search-gateway', and 'Application: SearchGatewayStack'. There are also 'Export to Infrastructure Composer' and 'Download' buttons.

## LextechSearchFunction (lextech-search-function)

- Receives search requests from the SearchGateway web interface via API Gateway.

The screenshot shows the AWS Lambda Functions interface. The top navigation bar includes 'aws', a search bar, and account information ('United States (N. Virginia) Patrick @ 2193-4244-2719'). Below the navigation is a breadcrumb trail: 'Lambda > Functions > lextech-search-function'. The main title is 'lextech-search-function'. On the right, there are buttons for 'Throttle', 'Copy ARN', and 'Actions'. A message box states: 'This function belongs to an application. Click here to manage it.' Below this is a 'Function overview' section with tabs for 'Diagram' (selected) and 'Template'. The diagram shows a Lambda function named 'lextech-search-function' with two triggers: 'API Gateway' (2 triggers) and '+ Add trigger'. To the right of the diagram are details: 'Description: Execute search queries on OpenSearch', 'Last modified: 1 day ago', 'Function ARN: arn:aws:lambda:us-east-1:219342442719:function:lextech-search-function', and 'Application: SearchFunctionStack'. There are also 'Export to Infrastructure Composer' and 'Download' buttons.

## API Gateway

- Search that receives requests with search queries and returns results from the OpenSearch backend.
- The API Gateway acts as a proxy to invoke the appropriate Lambda functions (SearchGateway for web interface, SearchFunction for search queries)

The screenshot shows the AWS API Gateway interface. On the left, there's a sidebar with 'API Gateway' and 'APIs' sections. Below that, under 'API: LexTech-Search-API', there are 'Resources', 'Stages', 'Authorizers', 'Gateway responses', 'Models', and 'Resource policy'. The main area is titled 'Resources' and shows a tree structure for a POST method at '/api/search'. To the right, a detailed view of the '/api/search - POST - Method execution' is shown, including the ARN (arn:aws:execute-api:us-east-1:219342442719:lmf57jlga/\*/POST/api/search), Resource ID (497yjf), and a flow diagram illustrating the request-response cycle between a Client, Method request, Integration request, Method response, Integration response, and a Lambda integration.

## Example 1

Search ‘the’ and multiple documents returned successfully that had the word ‘the’...

The screenshot shows a web browser window with the URL 'lmf57jlga.execute-api.us-east-1.amazonaws.com/prod/search'. The title bar says 'Search Page'. The main content area is titled 'Search Page' and contains a form with 'Enter Search Terms' and a 'Submit' button. Below the form, two search results are displayed:

- Title :** Communication  
**Author :** Unknown Author  
**Date :** Unknown Date  
**Summary :** --- Page 1 --- Communication is a continuous and dynamic process. It involves dissemination and understanding of information in the right context. In day to day communication, people express ideas, emotions, opinions and thoughts in a casual manner by using colloquial language and non-verbal cues. In such communication, there is a great possibility of misunderstanding and misinterpretation. Whether it is a formal or informal situation, communication should be meaningful, effective and correct.  
**E...**  
**Score :** 8.27398
- Title :** kerberos  
**Author :** Unknown Author  
**Date :** Unknown Date  
**Summary :** --- Page 1 --- Kerberos provides a centralized authentication server whose function is to authenticate users to servers and servers to users. In Kerberos Authentication server and database is used for client authentication. Kerberos runs as a third-party trusted server known as the Key Distribution Center (KDC). Each user and service on the network is a principal. The main components of

At the bottom, a footer bar says '© Powered by: pointernext.com'

## Example 2

Search word ‘biology’ and document returned successfully that had the word ‘biology’...

The screenshot shows a web browser window with the URL `lmpf57jlga.execute-api.us-east-1.amazonaws.com/prod/search`. The title bar has icons for back, forward, refresh, and a password entry field labeled "Enter Passphrase". Below the title bar is a dark header bar with the text "Search Page". The main content area is titled "Search Page" and contains a form with a text input field labeled "Enter Search Terms" containing the word "biology". Below the input field is a blue "Submit" button. To the right of the input field, there is a summary of search results:  
Title : Biology  
Author : Unknown Author  
Date : Unknown Date  
Summary : --- Page 1 --- 1. Biology Bio mean life and Logy mean study Definition The study of living organisms is called Biology Or The science of life and living organisms is called Biology. > Biology is divided into several specific fields that cover their morphology, physiology, anatomy, behavior, origin and distribution. > An organism is a living entity containing of one cell e.g. bacteria > An organism is a living entity containing of several cells e.g. animals, plants and fungi. 2. What ...  
Score : 19.218985

## Lessons and Observations

This search engine project was very hard and complex for me. I worked on this project daily for almost 2 weeks. I learned a lot along the way through trial and error to the point of almost giving up. I would make progress then hit a brick wall, then get past the brick wall to run into it all over again consistently. Through perseverance, I was able to finally see the vision of this project and work methodically through each step to completion. Looking back now, I have great deal of satisfaction for completing this project.

### A few complex issues throughout this project included:

- Misconfigured Lambda environment variables and VPC settings that broke OpenSearch.
- CI/CD pipeline setup was very challenging, failure after failure, mostly permissions and VPC related
- OpenSearch initial setup was rather simple, but the search, indexing and data upload was challenging. Main issue was the local roles security permissions, and mapping (mapped users) additional AWS roles to resolve permissions/access issues.
- Experienced multiple failures with pipeline implementation, trial and error took a great deal of time.

- VPC configuration preventing Lambda-to-OpenSearch communication - Configured security groups and subnets to allow Lambda functions to reach the VPC-based OpenSearch.
- Incomplete CI/CD pipeline deployment, "PLACEHOLDER\_TOKEN" instead of valid GitHub personal access tokens was the issue, preventing source code retrieval.
- PDF upload not appearing in search results - S3 upload → PDF processing → text extraction → OpenSearch indexing.
- Lambda function deployment inconsistencies, resolved through persistent trial and error.

Patrick M. Jones

8/4/2025

## Project Cleanup

The screenshot shows the AWS CloudWatch Metrics interface. A single metric named "OpenSearch Requests" is displayed. The value is 1.0, and the timestamp is 0. The chart has a single data series with one data point.

**Metric Details:**

Metric Name	Value	Timestamp
OpenSearch Requests	1.0	0

**Domain Overview:**

Domains (1) [Info](#)

Name	Status	Engine	Version	Deploy...	Endpoint	Cluster
lextech-search	Deleting Applying changes	OpenSearch	2.19	3-AZ with ...	VPC	-

OpenSearch includes certain Apache-licensed Elasticsearch code from Elasticsearch B.V. and other source code. Elasticsearch B.V. is not the source of that other source code. ELASTICSEARCH is a registered trademark of Elasticsearch B.V.

S3

EC2 > Instances

Successfully initiated termination (deletion) of i-0365c8d91a14be02c

Instances (1/1) Info

Last updated less than a minute ago

Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive)

All states

Name Instance ID Instance state Instance type Status check Alarm status Available

OpenSearchPortal i-0365c8d91a14be02c Terminated t3.micro - View alarms + us-east

S3

CloudFormation > Stacks

Stacks (2)

C Delete Update stack Stack actions Create stack

Filter status

Filter by stack name Active View nested

Stack name	Status	Created time	Description
UploadToSearch-pipeline-stack	DELETE_IN_PROGRESS	2025-08-03 10:38:08 UTC-0500	CI/CD Pipeline for UploadToSearch Lambda Function
SearchFunctionStack	DELETE_IN_PROGRESS	2025-08-02 08:48:00 UTC-0500	Lambda function for search execution

S3

Developer Tools > CodePipeline > Pipelines

CodePipeline

- Source • CodeCommit
- Artifacts • CodeArtifact
- Build • CodeBuild
- Deploy • CodeDeploy
- Pipeline • CodePipeline
  - Getting started
  - Pipelines**
  - Account metrics

Pipelines Info

C View history Release change Delete pipeline Create pipeline

Name	Latest execution status	Latest source revisions	Latest execution started	Most recent executions
No results There are no results to display.				

S3

Developer Tools X

**CodeBuild**

- ▶ Source • CodeCommit
- ▶ Artifacts • CodeArtifact
- ▼ Build • CodeBuild
  - Getting started
  - Build projects**
  - Build history
  - Report groups
  - Report history
  - Compute fleets New
  - Account metrics

Developer Tools > [CodeBuild](#) > Build projects

**Build projects** Info

[Actions](#) [Create trigger](#) [View details](#) [Debug build](#) [Start build](#)

[Create project](#)

[Your projects](#)

Name	Source provider	Repository	Latest build status	Description	Last Modified
No results There are no results to display.					

S3

aws | S3 | [Search](#) [Option+S] | [Actions](#) | [Bell](#) | [Help](#) | [Settings](#) | United States (N. Virginia) | Patrick @ 2193-4244-2719

**API Gateway** > APIs

**API Gateway** X

**APIs**

- Custom domain names
- Domain name access associations
- VPC links

---

- Usage plans
- API keys
- Client certificates
- Settings

**APIs (0/0)** [Delete](#) [Create API](#)

[Find APIs](#)

Name	Description	ID	Protocol	API endpoint type	Created
No API You don't have any apis.					

[Create API](#)

S3

aws | S3 | [Search](#) [Option+S] | [Actions](#) | [Bell](#) | [Help](#) | [Settings](#) | United States (N. Virginia) | Patrick @ 2193-4244-2719

**Lambda** > Functions

**Lambda** X

**Functions**

- Dashboard
- Applications
- Functions**

▼ Additional resources

- Code signing configurations
- Event source mappings
- Layers
- Replicas

**Functions (0)** Last fetched 4 seconds ago [Actions](#) [Create function](#)

[Filter by attributes or search by keyword](#)

Function name	Description	Package type	Runtime	Last modified
There is no data to display.				

AWS | [Option+S] | Search | United States (N. Virginia) | Patrick @ 2193-4244-2719

S3

Amazon S3 > Buckets

Amazon S3

General purpose buckets

- Directory buckets
- Table buckets
- Vector buckets [Preview](#)
- Access Grants
- Access Points (General Purpose Buckets, FSx file systems)
- Access Points (Directory Buckets)
- Object Lambda Access Points
- Multi-Region Access Points

General purpose buckets All AWS Regions Directory buckets

General purpose buckets (0) [Info](#)

[Create bucket](#)

Buckets are containers for data stored in S3.

Find buckets by name

Name	AWS Region	Creation date
No buckets You don't have any buckets.		