AWS Prescriptive Guidance

Defining S3 bucket and path names for data lake layers on the AWS Cloud

AWS Prescriptive Guidance: Defining S3 bucket and path names for data lake layers on the AWS Cloud

Table of Contents

Introduction 1
Targeted business outcomes 1
Recommended data layers 3
Naming S3 buckets in your data layers 4
Landing zone S3 bucket 4
Raw layer S3 bucket 5
Stage layer S3 bucket 6
Analytics layer S3 bucket 6
Mapping S3 buckets to IAM policies in your data lake 8
Handling sensitive data 9
Using a landing zone to mask sensitive data 9
FAQ 11
What name should I use for a multi-Region Amazon Simple Storage Service (Amazon S3) bucket? 11
Do I need to use raw, stage, and analytics as the names for my data lake layers? 11
Is it possible to rename an S3 bucket? 11
What happens if I delete an S3 bucket and want to reuse the name? 11
Are there limitations on what I can include in my S3 bucket or path's name? 11
Can I use more layers than the landing zone, raw, stage, and analytics layers in my data lake? 12
What happens if I have not defined my parameters? 12
How can I track costs at the business unit level? 12
What features should I consider when creating an S3 bucket naming standard? 12
Resources 13
Document history 14

Defining S3 bucket and path names for data lake layers on the AWS Cloud

Isabelle Imacseng, Samuel Schmidt, and Andrés Cantor, Amazon Web Services (AWS)

November 2021 (document history (p. 14))

This guide helps you create a consistent naming standard for Amazon Simple Storage Service (Amazon S3) buckets and paths in data lakes hosted on the Amazon Web Services (AWS) Cloud. The guide's naming standard for S3 buckets and paths helps you to improve governance and observability in your data lakes, identify costs by data layer and AWS account, and provides an approach for naming AWS Identity and Access Management (IAM) roles and policies.

We recommend that you use at least three data layers in your data lakes and that each layer uses a separate S3 bucket. However, some use cases might require an additional S3 bucket and data layer, depending on the data types that you generate and store. For example, if you store sensitive data, we recommend that you use a landing zone data layer and a separate S3 bucket. The following list describes the three recommended data layers for your data lake:

- **Raw data layer** – Contains raw data and is the layer in which data is initially ingested. If possible, we recommend that you retain the original file format and turn on versioning in the S3 bucket.
- **Stage data layer** – Contains intermediate, processed data that is optimized for consumption (for example CSV to Apache Parquet converted raw files or data transformations). An AWS Glue job reads the files from the raw layer and validates the data. The AWS Glue job then stores the data in an Apache Parquet-formatted file and the metadata is stored in a table in the AWS Glue Data Catalog.
- **Analytics data layer** – Contains the aggregated data for your specific use cases in a consumption-ready format (for example, Apache Parquet).

This guide's recommendations are based on the authors' experience in implementing data lakes with the serverless data lake framework (SDLF) and are intended for data architects, data engineers, or solutions architects who want to set up a data lake on the AWS Cloud. However, you must make sure that you adapt this guide's approach to meet your organization's policies and requirements.

The guide contains the following sections:

- Recommended data layers (p. 3)
- Naming S3 buckets in your data layers (p. 4)
- Mapping S3 buckets to IAM policies in your data lake (p. 8)
- Handling sensitive data (p. 9)

Targeted business outcomes

You should expect the following five outcomes after implementing a naming standard for S3 buckets and paths in data lakes on the AWS Cloud:

- Improved governance and observability in your data lake.

- Increased visibility into your overall costs for individual AWS accounts by using the relevant AWS account ID in the S3 bucket name and for data layers by using cost allocation tags for the S3 buckets.
- More cost-effective data storage by using layer-based versioning and path-based lifecycle policies.
- Meet security requirements for data masking and data encryption.
- Simplify data source tracing by enhancing developer visibility to the AWS Region and AWS account of the underlying data storage.

Recommended data layers

If you work with non-sensitive data, such as non-personally identifiable information (PII) data, we recommend that you use at least three different data layers in a data lake on the AWS Cloud.

However, you might require additional layers depending on the data's complexity and use cases. For example, if you work with sensitive data (for example, PII data), we recommend that you use an additional Amazon Simple Storage Service (Amazon S3) bucket as a landing zone and then mask the data before it is moved into the raw data layer. For more information about this, see the Handling sensitive data (p. 9) section of this guide.

Each data layer must have an individual S3 bucket; the following table describes our recommended data layers:

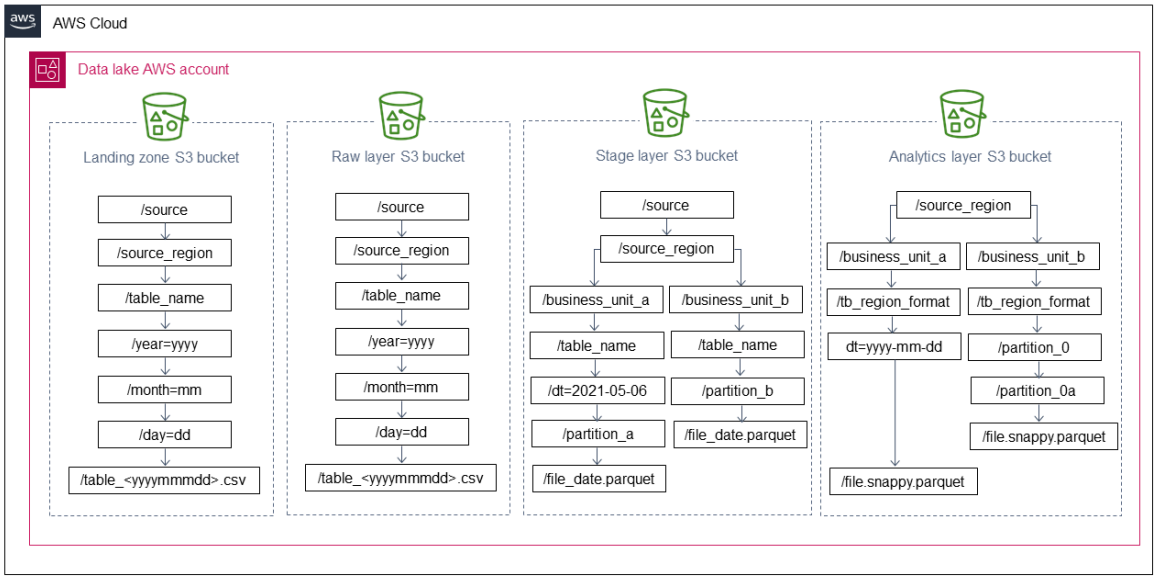
Data layer name	Description	Sample lifecycle policy strategy
Raw	Contains the raw, unprocessed data and is the layer in which data is ingested into the data lake. If possible, you should keep the original file format and turn on versioning in the S3 bucket.	After one year, move files into the Amazon S3 infrequent access (IA) storage class. After two years in Amazon S3 IA, archive them to Amazon S3 Glacier.
Stage	Contains intermediate, processed data that is optimized for consumption (for example CSV to Apache Parquet converted raw files or data transformations). An AWS Glue job reads the files from the raw layer and validates the data. The AWS Glue job then stores the data in an Apache Parquet-formatted file and the metadata is stored in a table in the AWS Glue Data Catalog.	Data can be deleted after a defined time period or according to your organization's requirements. Some data derivatives (for example, an Apache Avro transform of an original JSON format) can be removed from the data lake after a shorter amount of time (for example, after 90 days).
Analytics	Contains the aggregated data for your specific use cases in a consumption-ready format (for example, Apache Parquet).	Data can be moved to Amazon S3 IA and then deleted after a defined time period or according to your organization's requirements.

Note
You must evaluate all the recommended lifecycle policy strategies against your organizational needs, regulatory requirements, query patterns, and cost considerations.

Naming S3 buckets in your data layers

The following sections provide naming structures for Amazon Simple Storage Service (Amazon S3) buckets in your data lake layers. However, you can customize the S3 bucket and path names according to your organization's requirements. We recommend that you create separate S3 buckets for each individual layer because archiving, versioning, access, and encryption requirements can vary for each layer.

The following diagram shows the recommended naming structure for S3 buckets in the three recommended data lake layers, including separating multiple business units, file formats, and partitions. You can adapt data partitions according to your organization's requirements, but you should use lowercase and key-value pairs (For example, `year=yyyy`, not `yyyy`) so that you can update the catalog with the `MSCK REPAIR TABLE` command.



Important
S3 buckets must follow the naming guidelines from Bucket naming rules in the Amazon S3 documentation.

Landing zone S3 bucket

You require an Amazon Simple Storage Service (Amazon S3) bucket for your landing zone if sensitive datasets contain elements that must be masked before data is moved to the raw bucket.

The following table provides the naming structure, a description of the naming structure, and a name example for the S3 bucket in your landing zone layer.

Naming format	Example

`s3://companyname-landingzone-awsregion-awsaccount|uniqid-env/source/source_region/table/year=yyyy/month=mm/day=dd/table_<yearmonthday>.avro|csv` • `companyname` – The organization's name (optional). • `awsregion` – The AWS Region (for example, `us-east-1`, or `sa-east-1`). • `awsaccount|uniqid` – The unique identifier or AWS account ID. • `env` – The deployment environment (for example, `dev`, `test`, or `prod`). • `source` – The source or content (for example, MySQL database, ecommerce, or SAP). • `source_region` – For example, `us` or `asia`. • `table` – `tb_customer`, `tb_transactions`, or `tb_products`.	`s3://anycompany-landingzone-useast1-12345-dev/socialmedia/us/tb_products/year=2021/month=03/day=01/products_20210301.csv`

Raw layer S3 bucket

The raw data layer contains ingested data that has not been transformed and is in its original file format (for example, JSON or CSV). This data is typically organized by data source and the date that it was ingested into the raw data layer's Amazon Simple Storage Service (Amazon S3) bucket.

The following table provides the naming structure, a description of the naming structure, and a name example for the S3 bucket in your raw data layer.

Naming format	**Example**
`s3://companyname-raw-awsregion-awsaccount|uniqid-env/source/source_region/table/year=yyyy/month=mm/day=dd/table_<yearmonthday>.avro|csv` • `companyname` – The organization's name (optional). • `awsregion` – The AWS Region (for example, `us-east-1`, or `sa-east-1`). • `awsaccount|uniqid` – The unique identifier or AWS account ID. • `env` – The deployment environment (for example, `dev`, `test`, or `prod`). • `source` – The source or content (for example, MySQL database, ecommerce, or SAP). • `source_region` – For example, `us` or `asia`. • `table` – `tb_customer`, `tb_transactions`, or `tb_products`.	`s3://anycompany-raw-useast1-12345-dev/socialmedia/us/tb_products/year=2021/month=03/day=01/products_20210301.csv`

Stage layer S3 bucket

Data in the stage layer is read and transformed from the raw layer (for example, by using an AWS Glue or Amazon EMR job). This process validates the data (for example, by checking data types and headers) and then stores it in a consumption-ready file format such as Apache Parquet. The metadata is stored in a table in the AWS Glue Data Catalog.

The following table provides the naming structure, a description of the naming structure, and a name example for the S3 bucket in your stage data layer.

Naming format	Example
`s3://companyname-stage-awsregion-awsaccount|uniqid-env/source/source_region/business_unit/table/<partitions>/table_<table_name>_<yearmonthday>.snappy.parquet` • `companyname` – The organization's name (optional). • `awsregion` – The AWS Region (for example, `us-east-1`, or `sa-east-1`). • `awsaccount|uniqid` – The unique identifier or AWS account ID. • `env` – The deployment environment (for example, `dev`, `test`, or `prod`). • `source` – The source or content (for example, MySQL database, ecommerce, or SAP). • `source_region` – For example, `us` or `asia`. • `business_unit` – The business unit that the data is processed for. • `table` – `tb_customer`, `tb_transactions`, or `tb_products`. • `partitions` – Partitions that provide the best performance for the consumer, allowing the query engine to avoid full data scans.	`s3://anycompany-stage-saeast1-12345-dev/sap/br/customers/validated/dt=2021-03-01/table_customers_20210301.snappy.parquet`

Analytics layer S3 bucket

The analytics layer is similar to the stage layer because the data is in a processed file format, but the data is then aggregated according to your organization's requirements.

The following table provides the naming structure, a description of the naming structure, and a name example for the S3 bucket in your analytics data layer.

Naming format	Example
`s3://companyname-analytics-awsregion-awsaccount|uniqid-env/source_region/business_unit/`	`s3://anycompany-analytics-useast1-12345-dev/us/sales/`

`tb_<region>_<table_name>_<file_format>/<partition_0>/<partition_1>/.../<partition_n>/xxxxx.<compression>.<file_format>` • `companyname` – The organization's name (optional). • `awsregion` – The AWS Region (for example, `us-east-1`, or `sa-east-1`). • `awsaccount|uniqid` – The unique identifier or AWS account ID. • `env` – The deployment environment (for example, `dev`, `test`, or `prod`). • `source` – The source or content (for example, MySQL database, ecommerce, or SAP). • `source_region` – For example, `us` or `asia`. • `business_unit` – The business unit that the data is processed for. • `table` – `tb_customer`, `tb_transactions`, or `tb_products`. • `partitions` – Partitions that provide the best performance for the consumer, allowing the query engine to avoid full data scans.	`tb_us_customers_parquet/<partitions>/part-000001-20218c886790.c000.snappy.parquet`

Mapping S3 buckets to IAM policies in your data lake

We recommend that you map the data lake's Amazon Simple Storage Service (Amazon S3) buckets and paths to AWS Identity and Access Management (IAM) policies and roles by using the bucket names or paths in the IAM policy or role name. The following table shows a sample S3 bucket name and a sample IAM policy that is used to access this S3 bucket.

Sample Amazon S3 object path	Sample IAM policy
S3 bucket name – `<companyname>-raw-<aws_region>-<aws_accountid>-dev` **S3 bucket path** – `nosql/us/customers/year=2020/month=03/day=01/table_customers_20210301.csv`	`{ “Version” : “2012-10-17", “Statement” : [ { “Sid” : “s3-nosql-us-customers-get-list", “Effect” : “Allow”, “Principal” : “*”, “Action” : [ “s3:GetObject”, “s3:ListBucket” ], “Resource” : [ “arn:aws:s3:::<companyname>-raw-<aws_region>-<aws_accountid>-dev/*” ] } ] }`

Note
This is a sample IAM policy that shows our recommended naming standard for S3 buckets; however, you should ensure that you correctly configure S3 bucket policies according to your organization's policies and requirements.

Handling sensitive data

Typically, sensitive data contains PII or confidential information that must be secured for compliance or legal reasons. If encryption is only required on a row or column level, we recommend that you use a landing zone layer. This is *partially-sensitive* data.

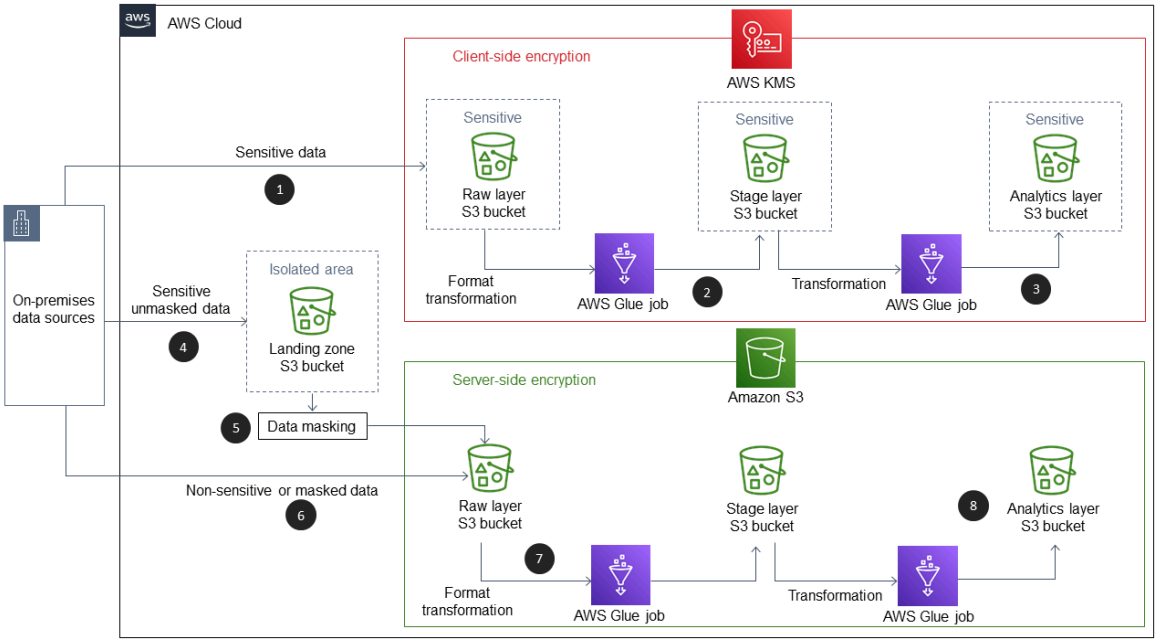
However, if the entire dataset is considered sensitive, we recommend using separate Amazon Simple Storage Service (Amazon S3) buckets to contain the data. This is *highly-sensitive* data. These separate S3 buckets must be used for each data layer and "*sensitive*" should be included in the bucket's name. We recommend that you encrypt sensitive buckets with AWS Key Management Service (AWS KMS) using Client-Side Encryption. You must also use client-side encryption to encrypt the AWS Glue jobs that transform your data.

Using a landing zone to mask sensitive data

You can use a landing zone layer for partially-sensitive datasets (for example, if encryption is only required at the row or column level). This data is ingested into the landing zone's S3 bucket and is then masked. After the data is masked, it is ingested into the raw layer's S3 bucket that is encrypted with Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3). If required, you can tag data at the object level.

Any data that is already masked can bypass the landing zone and be directly ingested into the raw layer's S3 bucket. There are two access levels in the stage and analytics layers for partially-sensitive datasets; one level has full access to all data and the other level only has access to non-sensitive rows and columns.

The following diagram shows a data lake where partially-sensitive datasets use a landing zone to mask the sensitive data but highly-sensitive datasets use separate, encrypted S3 buckets. The landing zone is isolated using restrictive IAM and S3 bucket policies, and the encrypted buckets use client-side encryption with AWS KMS.



The diagram shows the following workflow:

1. Highly-sensitive data is sent to an encrypted S3 bucket in the raw data layer.
2. An AWS Glue job validates and transforms the data into a consumption-ready format and then places file into an encrypted S3 bucket in the stage layer.
3. An AWS Glue job aggregates data according to business requirements and places the data into an encrypted S3 bucket in the analytics layer.
4. Partially-sensitive data is sent to landing zone bucket.
5. Sensitive rows and columns are masked and data is then sent to the S3 bucket in the raw layer.
6. Non-sensitive data is directly sent to the S3 bucket in the raw layer.
7. An AWS Glue job validates and transforms the data into a consumption-ready format and places the files into the S3 bucket for the stage layer.
8. An AWS Glue job aggregates the data according to your organization's requirements and places the data into an S3 bucket in the analytics layer.

FAQ

This section provides answers to commonly raised questions about defining S3 bucket and path names for data lake layers on the AWS Cloud.

What name should I use for a multi-Region Amazon Simple Storage Service (Amazon S3) bucket?

You can use our recommended S3 bucket naming format and change the AWS Region identifier. For example, `examplecompany-raw-`**`useast1`**`-12345-dev` and `examplecoompany-raw-`**`uswest1`**`-12345-dev`.

Do I need to use raw, stage, and analytics as the names for my data lake layers?

No, you can name your layers according to your requirements. However, we strongly recommended that you use an S3 bucket for the data layer that contains the original file formats and that has versioning enabled.

Is it possible to rename an S3 bucket?

No. If you want to use a different S3 bucket name, you must create a new bucket with the new name. This one reason why we recommend having a clearly defined and consistent naming approach for S3 buckets.

What happens if I delete an S3 bucket and want to reuse the name?

If you delete an S3 bucket and want to create a new bucket with the same name, you must wait several minutes for the name to become available again. S3 bucket names are globally unique and all AWS accounts share the same namespace.

Are there limitations on what I can include in my S3 bucket or path's name?

Only lowercase letters, numbers, dashes, and dots are allowed in S3 bucket names. Bucket names must be three to 63 characters in length, must begin and end with a number or letter, and cannot be in an IP address format. The names must also be globally unique.

For S3 bucket paths, you can use uppercase letters, but we recommend that you only use lowercase letters. Paths can also include additional symbols, but we recommend that you only use underscores, dashes, slashes, and numbers.

Can I use more layers than the landing zone, raw, stage, and analytics layers in my data lake?

Yes, you can use as many layers as you want. However, we recommend having a landing zone and raw layer for your raw data, an intermediate layer for formatted data, and a layer for highly-modeled data.

What happens if I have not defined my parameters?

Certain parameters (for example, business units) don't need to be incorporated into the S3 bucket name but can be part of the path. This means that they don't need to be immediately determined because paths can be added after an S3 bucket is created.

How can I track costs at the business unit level?

This depends on your account strategy. If you have business units split up into different AWS accounts, you can assign cost allocation tags to S3 buckets that reflect the bucket costs for each business unit.

If your account strategy doesn't separate out business units into different AWS accounts, then you can use different buckets for each business unit by adding the business unit to the bucket name (for example, `exampleco-businessunit1-raw-useast1-12345-dev`). However, this means that you have to manage many S3 buckets.

What features should I consider when creating an S3 bucket naming standard?

You must ensure that your S3 bucket names use features that are only available at the bucket level. For example, cost tags, bucket encryption, and versioning are features that are only available for an entire S3 bucket. This means that they apply to all objects and paths in the S3 bucket.

Object versioning is also an important feature to consider. You should turn on versioning for your raw layer's S3 buckets, because you want to make sure that you can see previous versions if there are changes to the data. However, versioning might not be necessary for all the layers in your data lake and retaining multiple versions can cause unnecessary costs.

Resources

- AWS Glue developer guide (AWS Glue documentation)
- AWS Glue components (AWS Glue documentation)
- Bucket naming rules (Amazon Simple Storage Service (Amazon S3) documentation)
- Protecting data using Server-Side Encryption with KMS keys stored in AWS Key Management Service (SSE-KMS) (Amazon S3 documentation)
- Using cost allocation S3 bucket tags (Amazon S3 documentation)
- User-defined cost allocation tags (AWS Billing and Cost Management documentation)

Document history

The following table describes significant changes to this guide. If you want to be notified about future updates, you can subscribe to an RSS feed.

Change	Description	Date
— (p. 14)	Initial publication	November 18, 2021