

# Stratus object storage system

Stratus, the CISL object storage disk system described here, is for long-term data storage.

Some documents attached below include the name of the vendor – Active Scale, a division of Western Digital – and some refer to the system with the name "Data Commons S3."

## Page contents

- [System overview](#)
  - [Policies](#)
  - [Requesting account](#)
  - [Documentation and additional information](#)
- 

## System overview

Stratus does NOT have POSIX file system access. In fact, it differs from other file systems in many ways:

- There is no directory structure, only a flat hierarchy with a single level (bucket and content of the bucket).
  - The data and metadata are accessed programmatically (rather than at the command line) with get/put commands, via an HTTP REST API.
  - Data and metadata can be accessed either via a library (such as Python's boto3) or a web browser (either directly for the HTTP calls or via web interface).
  - The system uses an API that is similar but not identical to the Amazon Web Services S3.
  - Accounts are identified by a key pair: access key and secret key, as in these examples:
    - **Access key:** AK0IYXKCCIA63BMNCOUN
    - **Secret key:** Joeke2uHHebQdKJBgTVUzp+j7uRDthPdIBI5YaLE
  - Accounts are associated with email, and each email address can have only a single account with a single role. A person who needs two roles must use two separate emails.
  - Two roles exist:
    1. Admin – An admin can create buckets and users, set up read/write access control for users, and do everything a user can do; owns data created by users.
    2. Users – Users may access buckets and read or write data inside buckets if the admin granted access. Users **cannot** create buckets.
- 

## Policies

- The system is not backed up.
  - Support will be provided during business hours on business days.
  - CISL will create only one admin account per lab. The admin will be able to create accounts for other users. Because the secret key-based logins do not expire, the admin will also delete accounts as appropriate – for example, when a user leaves NCAR.
- 

## Requesting account

Contact CISL to request an account. You will be asked to:

- Specify how much disk space you need.
  - Give a brief description (one sentence) of your intended use case.
  - Acknowledge that you will be the admin and will manage buckets and users.
- 

## Documentation and additional information

This related page will help you get started as an object storage admin: [Getting started with object storage admin account](#).

- Additional documentation is attached below.
- The system is accessible only via the NCAR VPN. This is important mostly for the browser-based access, since CISL anticipates that the server-based access will be from an internal server anyway.
- The access and secret credentials will be sent via email. They are all it takes a user to login (there is no UCAS, CIT, or Duo login). The NCAR username is irrelevant for this system.
- The way that these credentials are (unlike username/password) seems to nudge users towards nonoptimal patterns, such as hardcoding them into the source code. Users are strongly advised to **NOT** do that. Instead, use a separate file (outside of version control) similar to the following and source that file before running your code. This applies to both admin and user accounts.

```
export AWS_ACCESS_KEY_ID='xxx'
export AWS_SECRET_ACCESS_KEY='yyy'
```

- Admins might want to create a separate user account for themselves with just reading (and perhaps writing) capabilities and not admin capabilities. This would require use of a different email address, since the system does not allow re-use of existing emails. Admins might use a personal email, or a (group) alias setup in PeopleDB.

**Click to download**[Object\\_Storage\\_S3\\_API\\_Reference.pdf](#)[Object\\_Storage\\_View\\_User\\_Guide.pdf](#)