CS 240
Computer Systems

#17: Data Storage - File Systems to Cloud

March 24, 2022 · Wade Fagen-Ulmschneider

Data Storage

Throughout every program you have ever written, you have had to handle data storage in some way. Let's explore our options for data storage:

[1]: ________________________
Why? How?

[2]: ________________________
Why? How?

[3]: ________________________
Why? How?

[4]: ________________________
Why? How?

[5]: ________________________
Why? How?

[6]: ________________________
Why? How?

[7]: ________________________
Why? How?

File Systems

All modern systems utilize an Operating System to facilitate the storage of data in units called "files":

```
waf@sp22-cs240-001:~$ ls -la
drwxr-xr-x 7 waf  csvm240-cls 4096 Mar 22 11:25 .
drwxr-xr-x 3 root root        4096 Mar 10 13:42 ..
-rw------- 1 waf  csvm240-cls   19 Mar 10 13:56 .bash_history
-rw-r--r-- 1 waf  csvm240-cls  220 Mar 10 13:42 .bash_logout
-rw-r--r-- 1 waf  csvm240-cls 3771 Mar 10 13:42 .bashrc
drwx------ 2 waf  csvm240-cls 4096 Mar 10 13:42 .cache
drwxr-xr-x 2 waf  csvm240-cls 4096 Mar 22 11:22 cs240
drwxr-xr-x 2 waf  csvm240-cls 4096 Mar 21 14:35 docker
```

Permission Bits [1]	[3]	File Owner and Group [2]	File Size (bytes) [4] and Date Modified [5]	File Name [6]

[1]: Permission Bits:

d	r	w	x	r	w	x	r	w	x
Dir	User			Group			Other		

[2]: File Owner and File Group

[5]: Last Modified Date:

- Almost all modern operating systems store three different date fields for every single file:
  a.

  b.

  c.

- The date/time fields are always based on **your local computer clock** -- easily modified, easily faked.

[6]: File Name

- "dot" files and directories:

Q: Why does local file storage not work on a cloud-scale system?

Cloud Object Storage

Instead of using local file storage, large data storage in the cloud-based systems are commonly stored as **"objects"**. These objects (files) are organized into ____________________:

Public Cloud Providers	Private Cloud Solutions

Example: AWS

Amazon AWS S3 CreateBucket REST API
https://docs.aws.amazon.com/AmazonS3/latest/API/API_CreateBucket.html

```
PUT / HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-acl: ACL
x-amz-grant-read: GrantRead : UserList
x-amz-grant-write: GrantWrite : UserList
x-amz-grant-full-control: GrantFullControl : UserList
x-amz-grant-read-acp: GrantReadACP : UserList
x-amz-grant-write-acp: GrantWriteACP : UserList
[...]
```

Bucket:	Name of the bucket. *[Required]*
ACL:	The canned Access Control to apply to the bucket. `private | public-read | public-read-write | authenticated-read`
UserList:	You specify each grantee (user) as a type=value pair, where the type is one of the following: **id** – if the value specified is the canonical user ID of an AWS account **uri** – if you are granting permissions to a predefined group **emailAddress** – if the value specified is the email address of an AWS account **Ex:** `x-amz-grant-read: id="11112222333",id="444455556666"`
ACP:	**x-amz-grant-read** grants permission for the file itself; **x-amz-grant-read-acp** grants permissions for the access control policies.

+ Lots of Language-level Libraries

Private Cloud Solution:

MinIO: https://docs.min.io/docs/python-client-api-reference.html#make_bucket

OpenStack/Swift:
https://docs.openstack.org/api-ref/object-store/index.html?expanded=create-container-detail#create-container

Adding files to storage are also HTTP endpoints:

Amazon AWS S3 PutObject REST API
https://docs.aws.amazon.com/AmazonS3/latest/API/API_PutObject.html

```
PUT /Key HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-tagging: Tagging
x-amz-acl: ACL
x-amz-grant-full-control: GrantFullControl : UserList
x-amz-grant-read: GrantRead : UserList
x-amz-grant-read-acp: GrantReadACP : UserList
x-amz-grant-write-acp: GrantWriteACP : UserList
[...]
Content-Length: ContentLength

Body
```

Q: Is there a directory structure similar to traditional file systems?

Q: In both traditional file systems and S3, names must be unique. However, tagging allows for multiple files to have the same tag. What design possibilities does this open up for us?

Structured Storage Solutions

	Public Cloud	Private Cloud
Key-Value Stores		
Document Stores		
Relational Stores		