

## SUMMARY:

This Lab covers S3 fundamentals: Static website hosting, KMS, Encryption, S3 Replication, S3 Pre-signed URLs

## Contents

1. Creating a static website with S3 .....	1
2. Encrypting with KMS .....	4
3. Encrypting S3 Objects .....	7
4. S3 Replication for Disaster Recovery .....	9
5. Pre-Signed URLs .....	11

### 1. Creating a static website with S3

Create public bucket for animals 4 life:

Name ▲	AWS Region ▼	Access ▼	Creation date ▼
<input type="radio"/> <a href="#">top.10.animals4lifeivan.click</a>	US East (N. Virginia) us-east-1	Objects can be public	July 14, 2021, 19:29:07 (UTC-05:00)

Enable static hosting:

### Static website hosting

Use this bucket to host a website or redirect requests. [Learn more](#)

Static website hosting

Enabled ←

Hosting type

Bucket hosting

Bucket website endpoint

When you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. [Learn more](#)

<http://top.10.animals4lifeivan.click.s3-website-us-east-1.amazonaws.com> ←

Upload website files:

<input type="checkbox"/>	Name ▲	Folder ▼	Type ▼	Size ▼
<input type="checkbox"/>	annnndmerlin.jpg	img/	image/jpeg	53.9 KB
<input type="checkbox"/>	anothermerlin.jpg	img/	image/jpeg	53.9 KB
<input type="checkbox"/>	boris.jpg	img/	image/jpeg	14.9 KB
<input type="checkbox"/>	differentcat1.jpg	img/	image/jpeg	53.9 KB
<input type="checkbox"/>	differentcat2.jpg	img/	image/jpeg	53.9 KB
<input type="checkbox"/>	error.html	-	text/html	233.0 B
<input type="checkbox"/>	index.html	-	text/html	2.6 KB
<input type="checkbox"/>	merlin.jpg	img/	image/jpeg	53.9 KB
<input type="checkbox"/>	merlinagain.jpg	img/	image/jpeg	53.9 KB
<input type="checkbox"/>	samson.jpg	img/	image/jpeg	47.0 KB


## Destination

Destination

[s3://top.10.animals4lifeivan.click](#) 

Create bucket policy:

Bucket ARN

 `arn:aws:s3:::top.10.animals4lifeivan.click`

## Policy

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Sid": "PublicRead",  
6       "Effect": "Allow",  
7       "Principal": "*",  
8       "Action": ["s3:GetObject"],  
9       "Resource": ["arn:aws:s3:::top.10.animals4lifeivan.click/*"]  
10    }  
11  ]  
12 }  
13
```



JSON used:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "PublicRead",  
      "Effect": "Allow",  
      "Principal": "*",  
      "Action": ["s3:GetObject"],  
      "Resource": ["arn:aws:s3:::top.10.animals4lifeivan.click/*"]  
    }  
  ]  
}
```

```
{
  "Sid": "PublicRead",
  "Effect": "Allow",
  "Principal": "*",
  "Action": ["s3:GetObject"],
  "Resource": ["arn:aws:s3:::examplebucket/*"]
}
```

Create DNS record in Route53 and point to bucket:

**Define simple record**

**Record name**  
To route traffic to a subdomain, enter the subdomain name. For example, to route traffic to blog.example.com, enter *blog*. If you leave this field blank, the default record name is the name of the domain.

top.10 .animals4lifeivan.click

Valid characters: a-z, 0-9, ! " # \$ % & ' ( ) \* + , - / : ; < = > ? @ [ \ ] ^ \_ ` { | } . ~

**Record type**  
The DNS type of the record determines the format of the value that Route 53 returns in response to DNS queries.

A – Routes traffic to an IPv4 address and some AWS resources

Choose when routing traffic to AWS resources for EC2, API Gateway, Amazon VPC, CloudFront, Elastic Beanstalk, ELB, or S3. For example: 192.0.2.44.

**Value/Route traffic to**  
The option that you choose determines how Route 53 responds to DNS queries. For most options, you specify where you want to route internet traffic.

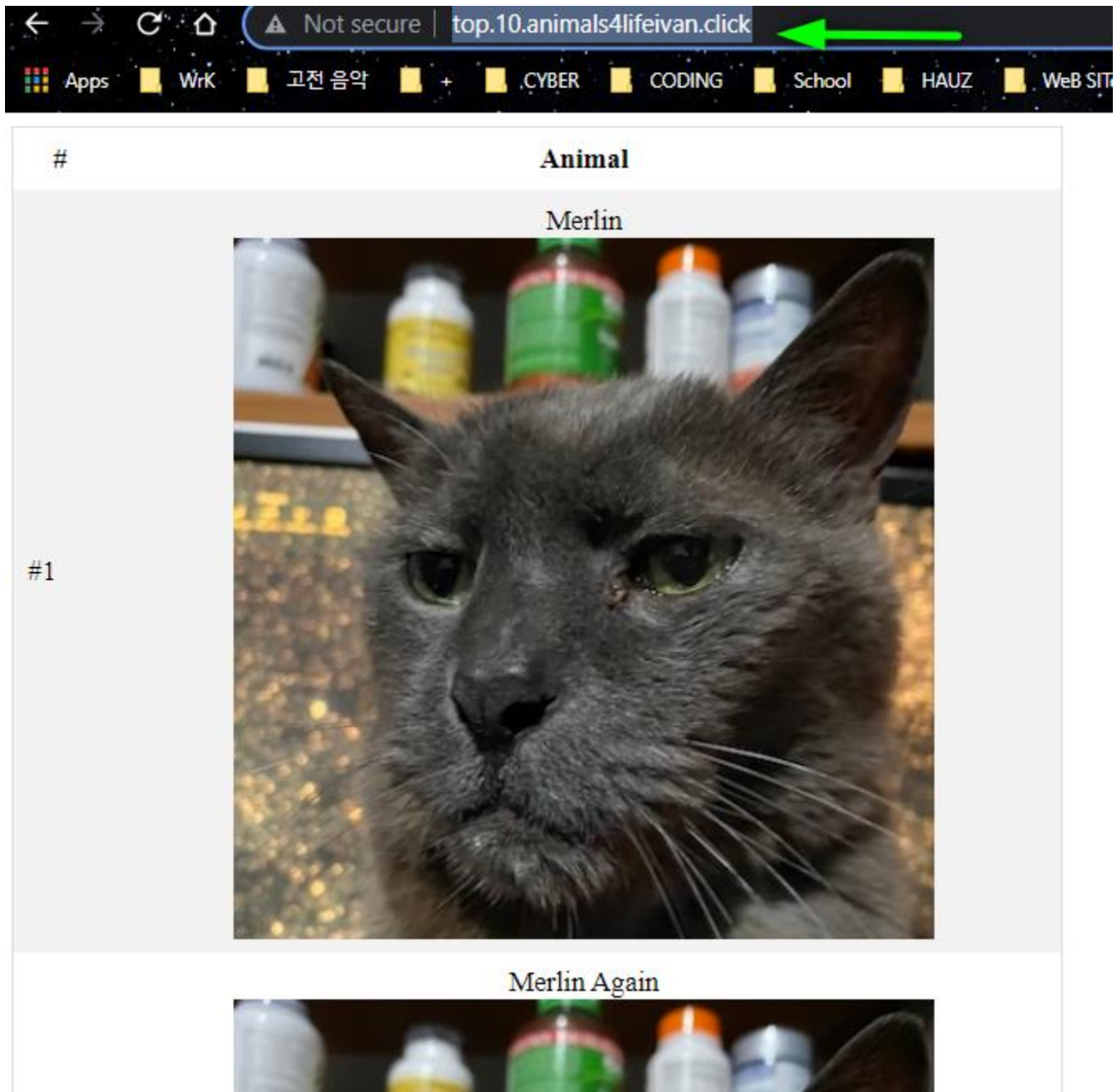
Alias to S3 website endpoint

US East (N. Virginia) [us-east-1]

s3-website-us-east-1.amazonaws.com

Cancel Define simple record

Test FQDN and verify it points to bucket site:



## 2. Encrypting with KMS

In this [DEMO] lesson we run through the practical steps of creating and configuring a Customer Managed CMK, an Alias and we use that CMK and the CLI tools to encrypt and decrypt some data.

Commands used:

```
# Shared
echo "find all the doggos, distract them with the yumz" > battleplans.txt
```

Windows Commands

```
aws kms encrypt --key-id alias/catrobot --plaintext fileb://battleplans.txt --output text --
profile iamadmin-general --query CiphertextBlob > battleplans.base64

certutil -decode battleplans.base64 not_battleplans.enc

aws kms decrypt --ciphertext-blob fileb://not_battleplans.enc --output text --
profile iamadmin-general --query Plaintext > decreyptedplans.base64

certutil -decode decreyptedplans.base64 decryptedplans.txt
```

Keypolicy & Key creation:

Customer managed keys (1)

Key actions ▼

Create key

Filter keys by properties or tags

< 1 >

⚙

<input type="checkbox"/>	Aliases ▼	Key ID ▼	Status	Key spec ⓘ	Key usage
<input type="checkbox"/>	ivancatrobot	f92b033b-11ee-4842-93c3-f9e6037c3638	Enabled	SYMMETRIC_DEFAULT	Encrypt and decrypt

Key Policy details:

```
{
  "Id": "key-consolepolicy-3",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Enable IAM User Permissions",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::361618461803:root"
      },
      "Action": "kms:*",
      "Resource": "*"
    },
    {
      "Sid": "Allow access for Key Administrators",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::361618461803:user/iamadmin"
      },
      "Action": [
        "kms:Create*",
        "kms:Describe*",
        "kms:Enable*",
        "kms:List*",
        "kms:Put*",
        "kms:Update*",
        "kms:Revoke*",
        "kms:Disable*",
        "kms:Get*",
        "kms>Delete*"
      ]
    }
  ]
}
```

```
        "kms:TagResource",
        "kms:UntagResource",
        "kms:ScheduleKeyDeletion",
        "kms:CancelKeyDeletion"
    ],
    "Resource": "*"
},
{
    "Sid": "Allow use of the key",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::361618461803:user/iamadmin"
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
},
{
    "Sid": "Allow attachment of persistent resources",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::361618461803:user/iamadmin"
    },
    "Action": [
        "kms:CreateGrant",
        "kms:ListGrants",
        "kms:RevokeGrant"
    ],
    "Resource": "*",
    "Condition": {
        "Bool": {
            "kms:GrantIsForAWSResource": "true"
        }
    }
}
]
```

Using key to encrypt & decrypt test file in windows 10 cli:

```
Microsoft Windows [Version 10.0.19042.1083]
(c) Microsoft Corporation. All rights reserved.

C:\Users\IvanVlad>echo "ivan test kms lab 1" > battleplans.txt

C:\Users\IvanVlad>aws kms encrypt --key-id alias/ivancatrobot --plaintext fileb://battleplans.txt --output text --profile iamadmin-general --query CiphertextBlob > battleplans.base64

C:\Users\IvanVlad>type battleplans.base64
AQICAHgWu1rSzEHVb+/L3JwLDdH4DnJvfHyswmb2xWgeTJEKAG+enppnQ2scnXd6Z2FqQgHAAAAdjB08gkqhkiG9w0BBwagZzBlAgEAMGAGCSqGSIB3DQEHATAeBgIghkgB8ZQMEAS4wEQQM39oTyfmiSlvCkXFVAgEQgDMP
HfYyVmbB4mvyDTlyhOW2qY9zjdlSc9ys0DxGtUit58CTDI+7TbW6+TbPfwXkcPbpfuQ=

C:\Users\IvanVlad>certutil -decode battleplans.base64 not_battleplans.enc
Input Length = 238
Output Length = 176
CertUtil: -decode command completed successfully.

C:\Users\IvanVlad>aws kms decrypt --ciphertext-blob fileb://not_battleplans.enc --output text --profile iamadmin-general --query Plaintext > decryptedplans.base64

C:\Users\IvanVlad>type decryptedplans.base64
ivan test kms lab 1

C:\Users\IvanVlad>type decryptedplans.txt
ivan test kms lab 1
```

Annotations in the image:

- Encryption command**: Points to the `aws kms encrypt` command.
- File now unreadable**: Points to the base64 encoded output of the encryption command.
- Decrypting**: Points to the `aws kms decrypt` command.
- File now readable**: Points to the output of the decryption command, which matches the original file content.

### 3. Encrypting S3 Objects

In this [DEMO] lesson we create an S3 bucket, and upload 4 images to the bucket using different encryption methods.

After adjusting the permissions of the IAMADMIN user we review what access changes occur, and why.

This DEMO focusses on the role separation aspect of S3 encryption using KMS.

Create:

### Server-side encryption settings

Server-side encryption protects data at rest. [Learn more](#)

**Server-side encryption**

☐ Disable

☒ Enable

**Encryption key type**

To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.

☐ Amazon S3 key (SSE-S3)

An encryption key that Amazon S3 creates, manages, and uses for you. [Learn more](#)

☒ AWS Key Management Service key (SSE-KMS)

An encryption key protected by AWS Key Management Service (AWS KMS). [Learn more](#)

**AWS KMS key**

☐ AWS managed key (aws/s3)

arn:aws:kms:us-east-1:945690336440:alias/aws/s3

☒ Choose from your KMS master keys

☐ Enter KMS master key ARN

**KMS master key**

arn:aws:kms:us-east-1:945690336440:key/6664f... [↻](#) [Create key](#)

**Bucket Key is disabled for objects uploaded, modified, or copied in this bucket**

Uploaded, modified, or copied objects inherit their Bucket Key settings from the bucket default encryption configuration unless they already have Bucket Key configured. [Learn more](#)


JSON for deny KMS policy for IAM admin:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Deny",
      "Action": "kms:*",
      "Resource": "*"
    }
  ]
}
```

Access Denied with new policy:

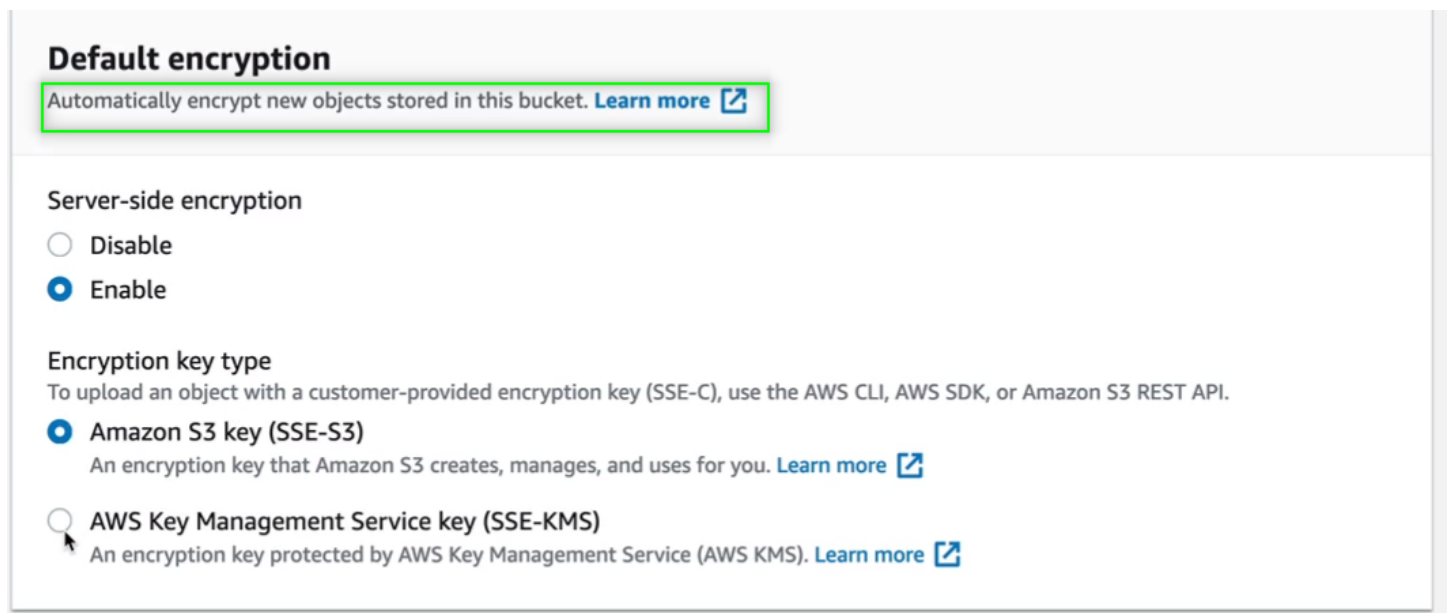


This XML file does not appear to have any style information associated with it. The document tree is shown below.




```
-<Error>
  <Code>AccessDenied</Code>
  <Message>Access Denied</Message>
  <RequestId>37F0CEDD6099029C</RequestId>
  <HostId>
    HYLYqWdUu96nmys1M8byU48xDfdNh1Az/215MBOdLdMIdRWrlk1+BLfa+odRS1KFCPH+HA/MZ2U=
  </HostId>
</Error>
```

Can select default encryption for auto enc for new objects:



**Default encryption**

Automatically encrypt new objects stored in this bucket. [Learn more](#) 

**Server-side encryption**


☐ Disable

☒ Enable


**Encryption key type**

To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.

☒ Amazon S3 key (SSE-S3)

An encryption key that Amazon S3 creates, manages, and uses for you. [Learn more](#) 

☐ AWS Key Management Service key (SSE-KMS)

An encryption key protected by AWS Key Management Service (AWS KMS). [Learn more](#) 

## 4. S3 Replication for Disaster Recovery

In this [DEMO] We create 2 S3 buckets - one in N. Virginia, the other in N. California and configure Cross-Region Replication (CRR) between the two.


Replication policy:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicRead",
      "Effect": "Allow",
      "Principal": "*",
      "Action": ["s3:GetObject"],
      "Resource": ["arn:aws:s3:::examplebucket/*"]
    }
  ]
}
```

```
]
}
```


Source Bucket:

Bucket ARN

 arn:aws:s3:::sourcebucketivan1234


### Policy

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "PublicRead",
6       "Effect": "Allow",
7       "Principal": "*",
8       "Action": ["s3:GetObject"],
9       "Resource": ["arn:aws:s3:::arn:aws:s3:::sourcebucketivan1234/*"]
10    }
11  ]
12 }
13
```



Destination Bucket:

Bucket ARN

 arn:aws:s3:::destinationbucketivan1234

### Policy

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "PublicRead",
6       "Effect": "Allow",
7       "Principal": "*",
8       "Action": ["s3:GetObject"],
9       "Resource": ["arn:aws:s3:::destinationbucketivan1234/*"]
10    }
11  ]
12 }
13
```



Replication rule for another region (DR):

**Replication rules (1)**  
Use replication rules to define options you want Amazon S3 to apply during replication such as server-side encryption, replica ownership, transitioning replicas to another storage class, and more. [Learn more](#)

[Refresh](#) [View details](#) [Edit rule](#) [Delete](#) [Actions](#) [Create replication rule](#)

	Replication rule name	Status	Destination bucket	Destination Region	Priority	Scope	Storage class	Replica owner	Re Ti C
<input type="radio"/>	staticwebsiteDR	Enabled	s3://destinationbucketivan1234	US West (N. California) us-west-1	0	Entire bucket	Same as source	Same as source	Di

## 5. Pre-Signed URLs

In this [DEMO] lesson you will create a bucket, upload an object and generate a presignedURL allowing access for any unauthenticated identities.

Bucket object target for presignedURL:

**animals4lifemediaivan12345** [Info](#)

[Objects](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

**Objects (1)**  
Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

[Refresh](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	all5.jpg	jpg	July 14, 2021, 23:41:37 (UTC-05:00)	779.7 KB	Standard

URI for cloudshell presignedURL:

S3 URI copied

s3://animals4lifemediaivan12345/all5.jpg

Cloudshell presignedURL creation:

```
aws Services Search for services, features, marketplace products, and docs [Alt+S] iamadmin@ivan-general N. Virginia Support
```

AWS CloudShell

us-east-1

```
Preparing your terminal...
Try these commands to get started:
aws help or aws <command> help or aws <command> --cli-auto-prompt
[cloudshell-user@ip-10-1-165-118 ~]$ aws s3 ls
2021-07-15 04:40:31 animals4lifemediain12345
[cloudshell-user@ip-10-1-165-118 ~]$ aws s3 presign s3://animals4lifemediain12345/all5.jpg --expires-in 180
https://animals4lifemediain12345.s3.us-east-1.amazonaws.com/all5.jpg?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=ASTIAVIMRD2RVZMWT3UJ%2F20210715%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20210715T044645Z&X-Amz-Expires=180&X-Amz-SignedHeaders=host&X-Amz-Security-Token=IQoJb3JpZ2luZXZlEC8aCXVzLWVhc3QtMSJGMEQCIB8cPQjBDMTgkZextwman2WQz2fL%2Fc50rAbxKeOKTux2BA1ApAcMjMeVYFPDyz4gh08LQjfgtHpr%2FX2ByDBH4c4TK0RryqfAwgmEAAaDDM2HTYxODQ2MTgwMyIMDB10Pm34d2LYLakmKwCBvnnazg8jCurf%2B8X%2B%2F15JQc09e2uQVweat7en61MEwHgbH1MSE8PezCZA09%2FuuU51Fss%2BQ3N1GhkP1oLulCqLIvkLX2Btdxd2B091k1kHY5fgM7UJW1MkkL17yQUP2CykHtEPbjQ%2BmZ581w3n%2F2YV4v8WmMOQk3k3WYcKb8sUZRYuLAX2FQ6uGy4LZ0zgx4boH7GzW51X2B4kxuwV283H5qksVQ20ZE1QidmVD1Pe2HJq0v%2B8LvtZpt2ARFgyoRQFU1khEpgYUzVcJyAy3K1rOfwioee38BCJeoGV%2FbmcQNhHtkNf1R1cyJCQ50w7wJa0030y1V1G214Xn74UEFA5W5nKAgxON9cGaR21kze76Zh%2BnCK28re825ga1p4X7dHOVAHYVqbH1bsxtLK6ZvLCB%2Fax4YYvziZhkb2N8g4Lo4gv8gPKFQgm%2Bj4vYg6%2BNCgAUnkEBAV2B8sG2cn9emstv0%2FY8CcfKtUshf3h3H0aPJ2ay5wZ62Xj87Ex60w51CK2BhwY6tALOHuKBNwseU11ZVE85dyU2GugezWu96odtx8g0Nc1mTqspnuhakDCR12aUC1Vjk0K0mHqV1H1%2Frdw51GRMCPKEK%2B%2B5BT3dAIcgs6Wxy6dc210xLaf5yH1IX2BrTRn0csoS5x%2Bw4e8R7E1KbRdd11ktTCMnk5ZNGVmmX3Ndqm0sKAGFHRxK2%2Bmon1HASQK7aG443%2Bj8G%2BNFgN1eFqoEVDFW9B0pJyA60LRjplwP2hhmpu%2FQvrjpp9t0JX4h4KqDyQvRoxz4ohmhy8RFKFMdJwQFFAeyTwygkq25n5Mwez6txvT9pJ4P0WdPC061cywQTCjtwOfI7A6rXrbZH%2F8pJY2ctegL157IEp4041P70rG4TM5HQZeCXD77ToyDguFzq86h5Ny1K9Tr4DgbvNNAFuLl%3D%3D&X-Amz-Signature=5bf86c612b63c07ff03c4ed98bf9272b89a73816215019e563de4385e5cbd87
[cloudshell-user@ip-10-1-165-118 ~]$
```

Take presignedURL and check if it works:

