Lab 8
Deploy weathermood to AWS

Software Studio

DataLab, CS, NTHU

2021 spring

What AWS services we would use

- **IAM**
- **Elastic Beanstalk**

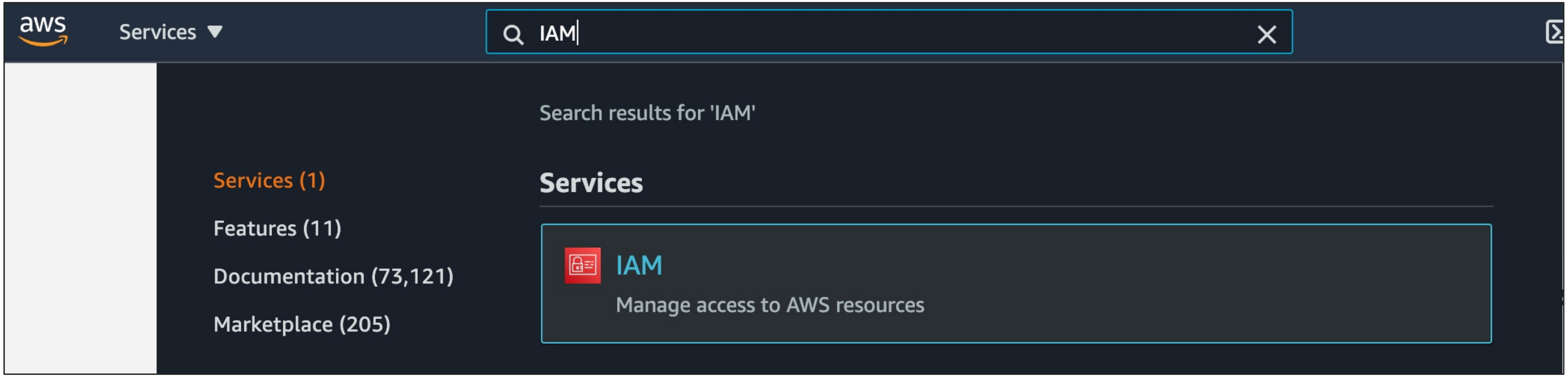
Manage access to AWS using IAM

IAM (Identity and Access Management)

- Enable you to manage access to AWS services and resources securely.
- Create and manage AWS users and groups, and use permissions to allow/deny their access to AWS resources.

IAM – Steps to create a new user

Step 1: Find IAM Services



IAM

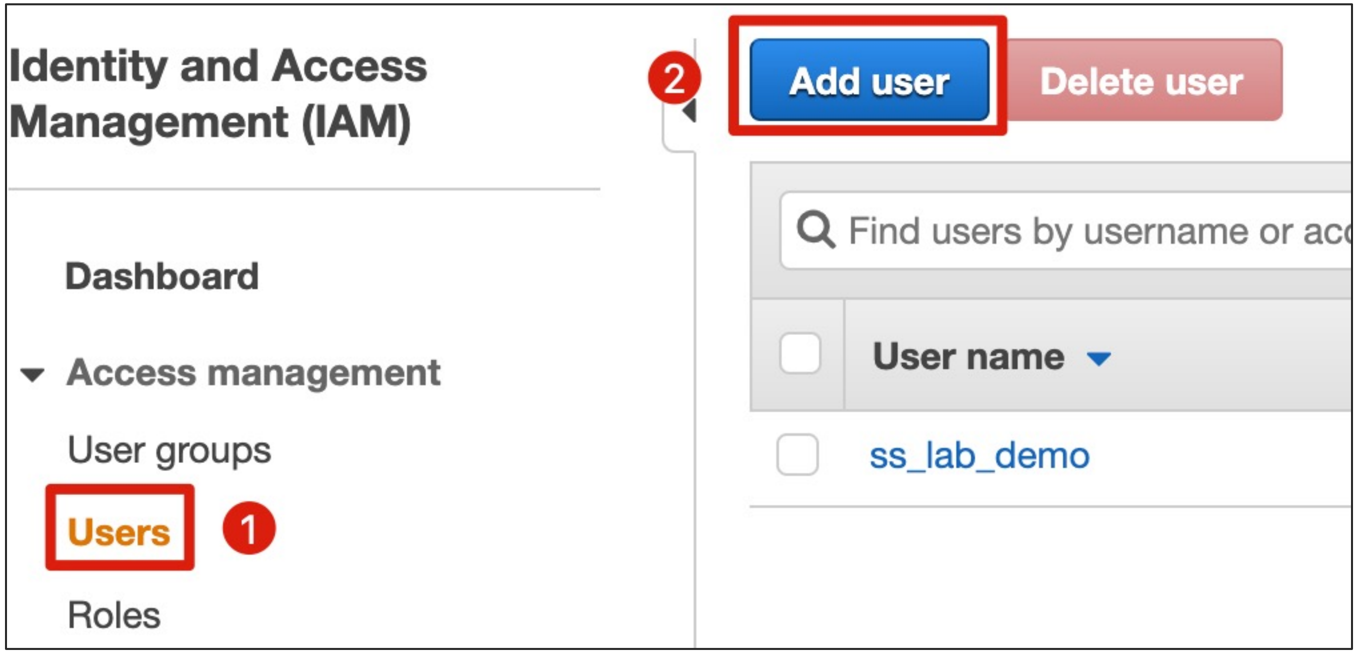
Search results for 'IAM'

IAM

Manage access to AWS resources

IAM – Steps to create a new user

Step 2: Go to “Users”, Click “Add user”

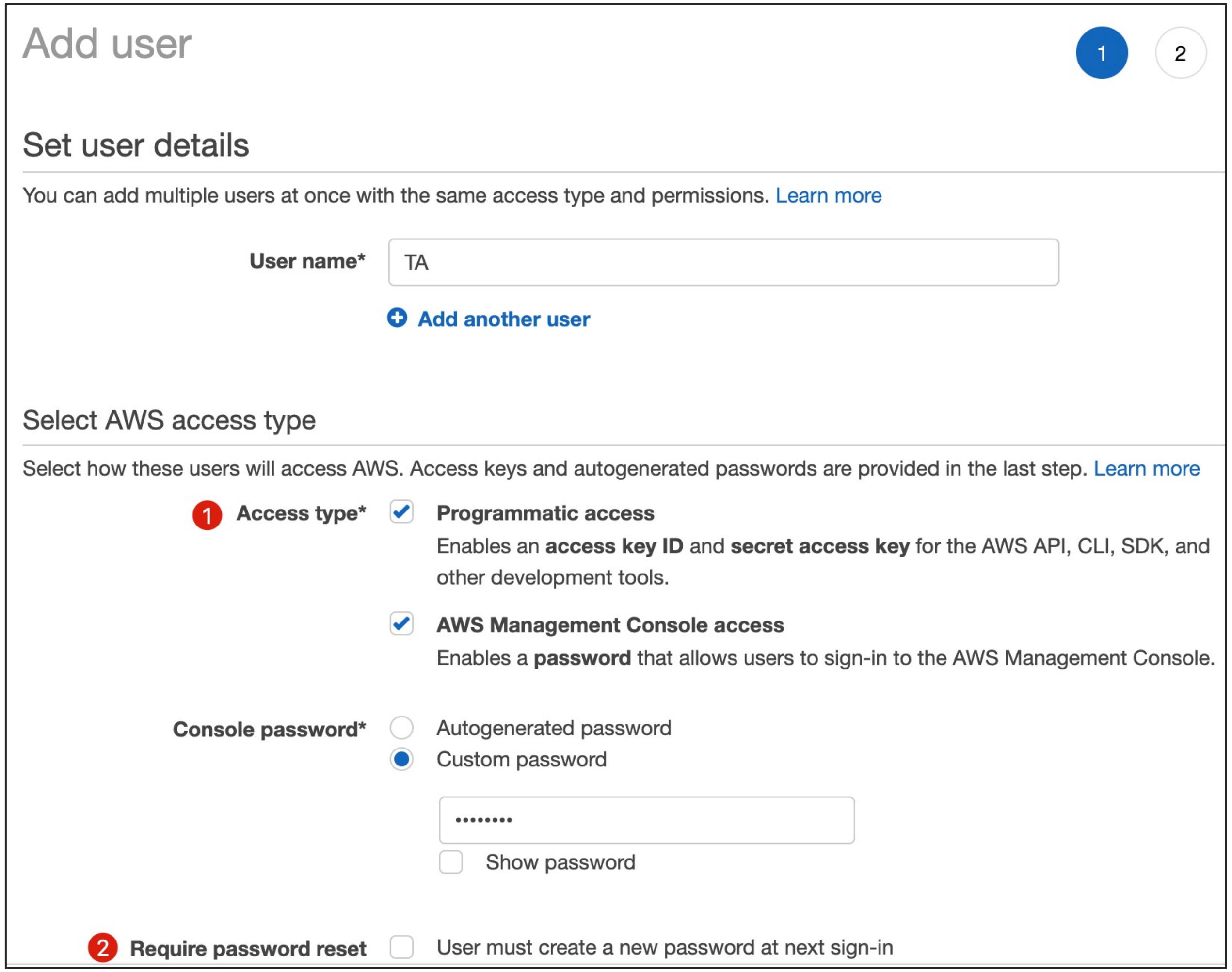


IAM – Steps to create a new user

Step 3: Set user details

1. Check ”Programmatic access” & “AWS Management Console access”

2. Uncheck “Require password reset”



Add user

1 2

Set user details

You can add multiple users at once with the same access type and permissions. Learn more

User name* TA

Add another user

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. Learn more

1 Access type* ☑ **Programmatic access**
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

☑ **AWS Management Console access**
Enables a **password** that allows users to sign-in to the AWS Management Console.

Console password* ○ Autogenerated password
● Custom password

••••••••

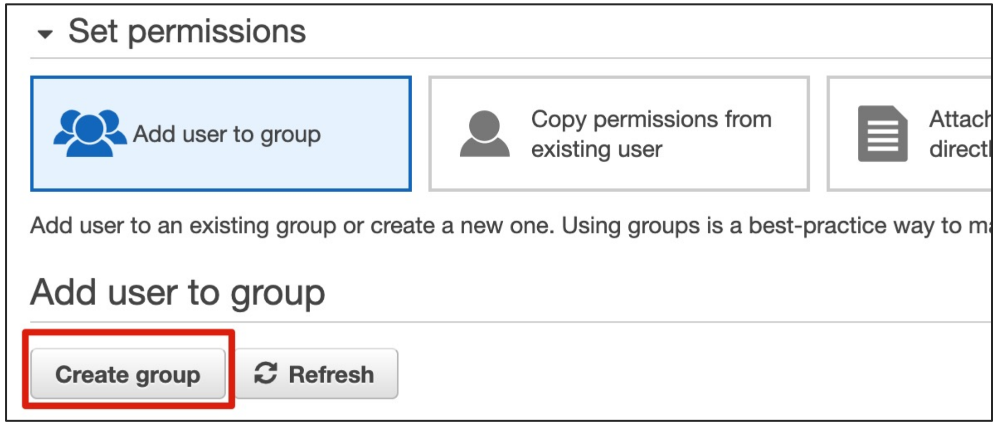
□ Show password

2 Require password reset □ User must create a new password at next sign-in

IAM – Steps to create a new user

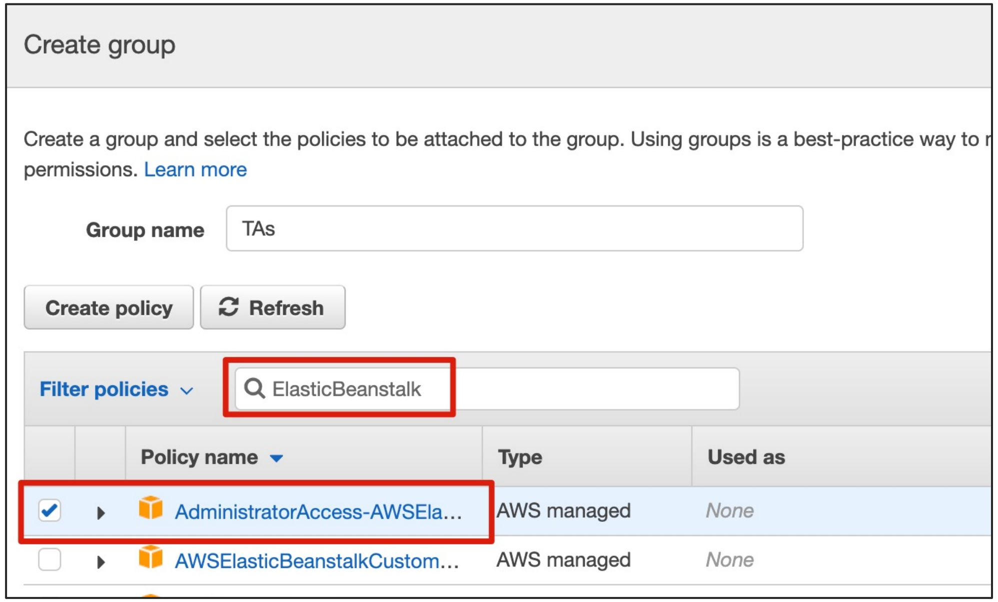
Step 4: Set permissions

1. Click “Create group”
2. Search “AdministratorAccess-AWSElasticBeanstalk”



Add user to an existing group or create a new one. Using groups is a best-practice way to ma

Add user to group



Create a group and select the policies to be attached to the group. Using groups is a best-practice way to r permissions. Learn more

Group name: TAs

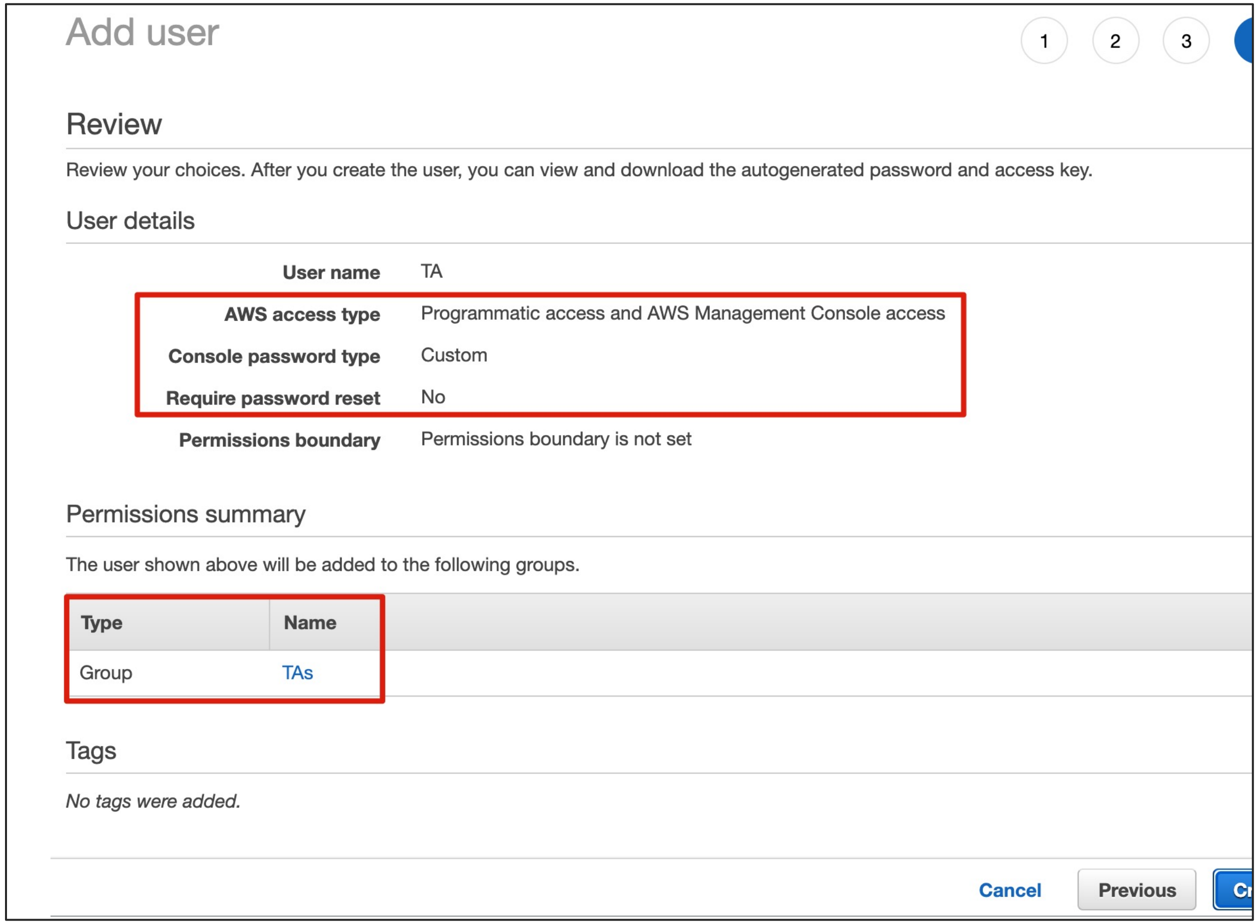
ElasticBeanstalk

		Policy name	Type	Used as
☑	▸	AdministratorAccess-AWSEla...	AWS managed	None
□	▸	AWSElasticBeanstalkCustom...	AWS managed	None

IAM – Steps to create a new user

Step 5: Review

- Check User details, Permissions summary are correctly set
- Click ”Create user”



Add user

Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name	TA
AWS access type	Programmatic access and AWS Management Console access
Console password type	Custom
Require password reset	No
Permissions boundary	Permissions boundary is not set

Permissions summary

The user shown above will be added to the following groups.

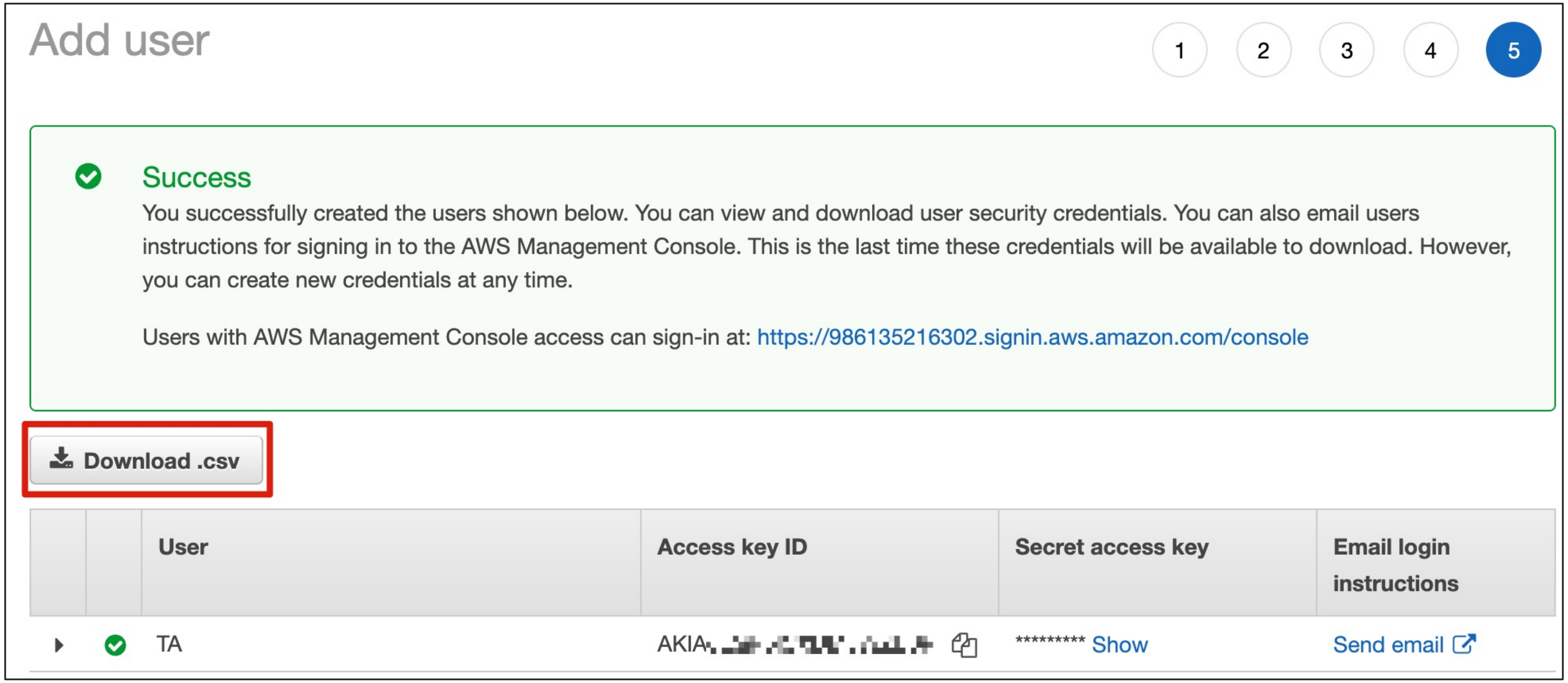
Type	Name
Group	TAs

Tags

No tags were added.

IAM – Steps to create a new user

Step 6: Download credentials file



Success

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: https://986135216302.signin.aws.amazon.com/console

		User	Access key ID	Secret access key	Email login instructions
▸	✓	TA	AKIA[redacted]	********* Show	Send email

Deploy to AWS using Elastic Beanstalk

Get project from Gitlab

- Clone project from GitLab
  - The client side code is in the weathermood_no_redux project
  - The server side code is in the weathermood-server_no_redux project
- Checkout branch
  - weathermood_no_redux -> server-file branch
  - weathermood-server_no_redux -> file branch

Elastic Beanstalk

How it works:

You simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, and automatic scaling to web application health monitoring, with ongoing fully managed patch and security updates.

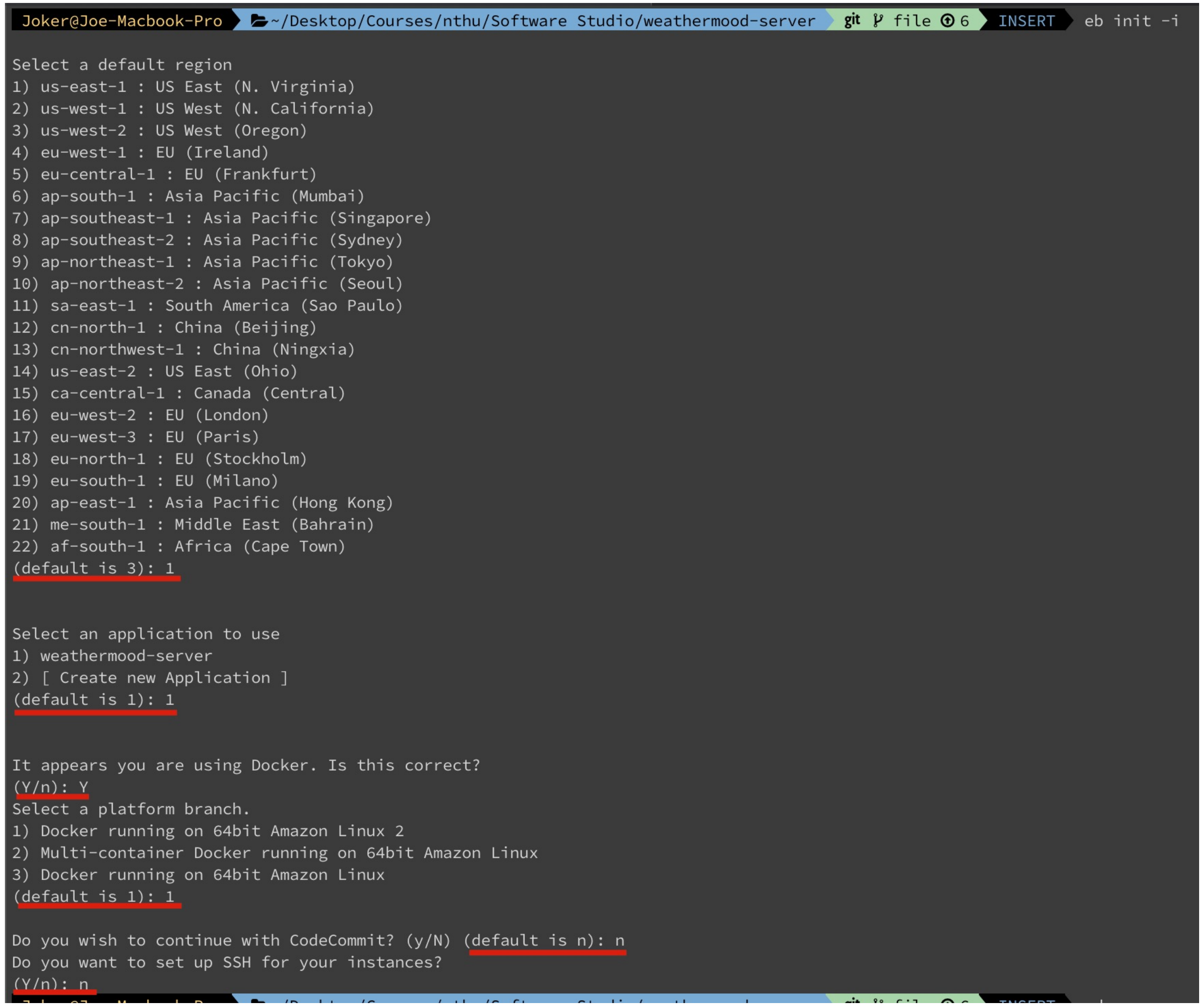
AWS EB CLI

- The AWS Elastic Beanstalk Command Line Interface (EB CLI) is a command line client that you can use to create, configure, and manage Elastic Beanstalk environments.
- To follow this lab, please install EB CLI first. Here is the link for more details

Deploy to AWS with AWS EB CLI

Step1: Setup application

- eb init –i

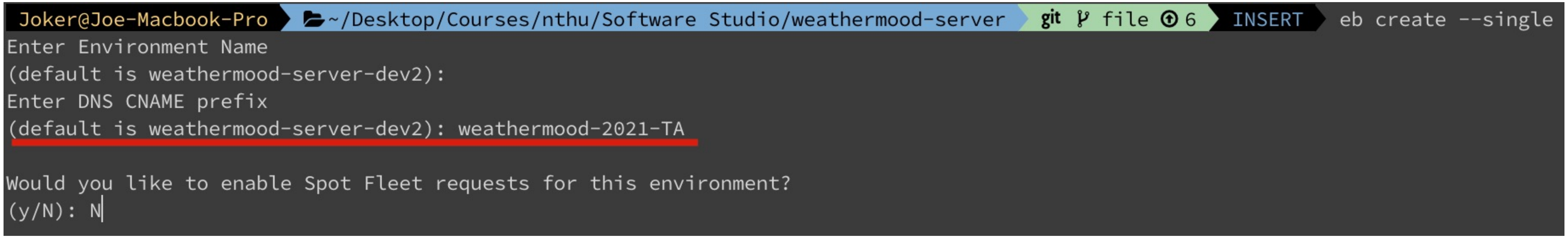


```
Joker@Joe-Macbook-Pro  ~/Desktop/Courses/nthu/Software Studio/weathermood-server  git  file ⊙6  INSERT  eb init -i

Select a default region
1) us-east-1 : US East (N. Virginia)
2) us-west-1 : US West (N. California)
3) us-west-2 : US West (Oregon)
4) eu-west-1 : EU (Ireland)
5) eu-central-1 : EU (Frankfurt)
6) ap-south-1 : Asia Pacific (Mumbai)
7) ap-southeast-1 : Asia Pacific (Singapore)
8) ap-southeast-2 : Asia Pacific (Sydney)
9) ap-northeast-1 : Asia Pacific (Tokyo)
10) ap-northeast-2 : Asia Pacific (Seoul)
11) sa-east-1 : South America (Sao Paulo)
12) cn-north-1 : China (Beijing)
13) cn-northwest-1 : China (Ningxia)
14) us-east-2 : US East (Ohio)
15) ca-central-1 : Canada (Central)
16) eu-west-2 : EU (London)
17) eu-west-3 : EU (Paris)
18) eu-north-1 : EU (Stockholm)
19) eu-south-1 : EU (Milano)
20) ap-east-1 : Asia Pacific (Hong Kong)
21) me-south-1 : Middle East (Bahrain)
22) af-south-1 : Africa (Cape Town)
(default is 3): 1


Select an application to use
1) weathermood-server
2) [ Create new Application ]
(default is 1): 1


It appears you are using Docker. Is this correct?
(Y/n): Y
Select a platform branch.
1) Docker running on 64bit Amazon Linux 2
2) Multi-container Docker running on 64bit Amazon Linux
3) Docker running on 64bit Amazon Linux
(default is 1): 1

Do you wish to continue with CodeCommit? (y/N) (default is n): n
Do you want to set up SSH for your instances?
(Y/n): n
```

Deploy to AWS with AWS EB CLI

Step2: Create environment

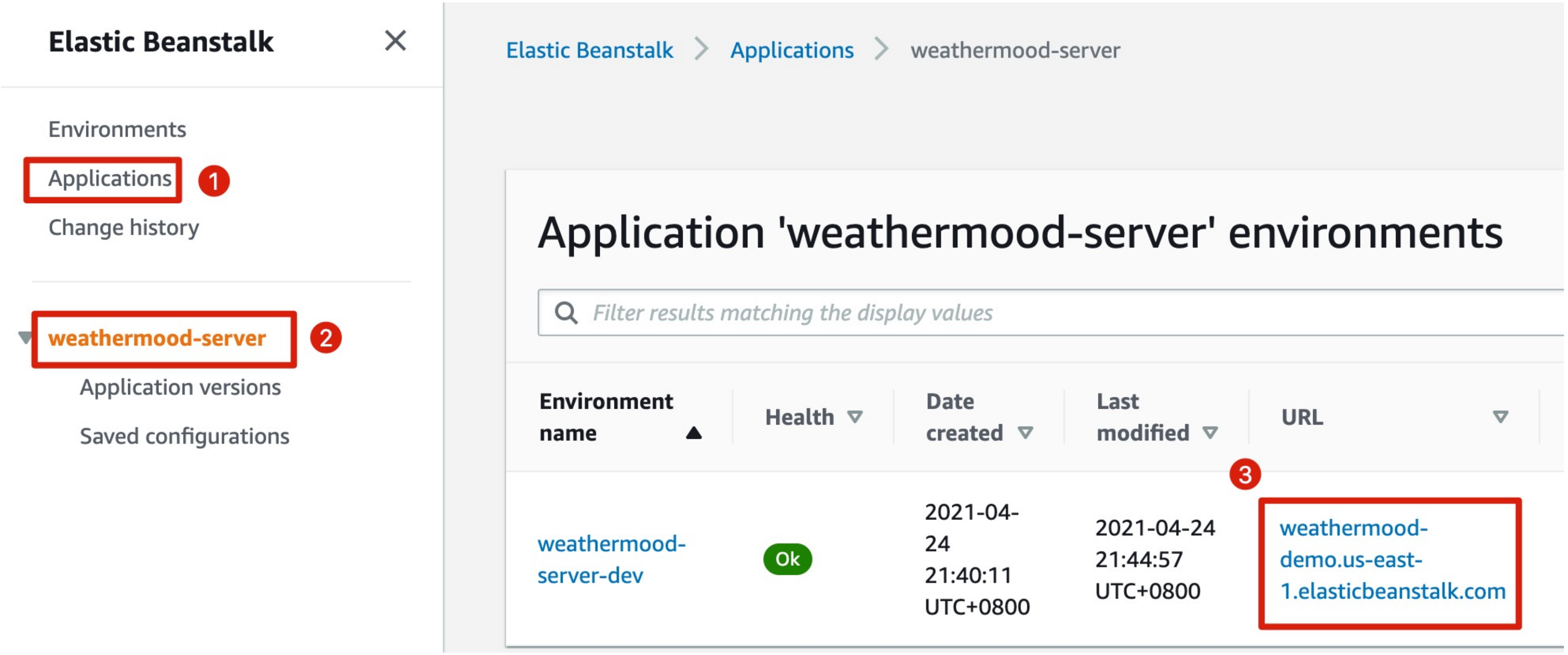
- eb create --single
- Enter weathermood-2021-{group_id} for DNS CNAME prefix
  - E.g. weathermood-2021-1 for group 1



```
Joker@Joe-Macbook-Pro  ~/Desktop/Courses/nthu/Software Studio/weathermood-server  git  file ⬆ 6  INSERT  eb create --single
Enter Environment Name
(default is weathermood-server-dev2):
Enter DNS CNAME prefix
(default is weathermood-server-dev2): weathermood-2021-TA

Would you like to enable Spot Fleet requests for this environment?
(y/N): N
```

Deploy to AWS with AWS EB CLI

Step3: Get the URL of your environment on AWS Console



Elastic Beanstalk

Environments

Applications 1

Change history

weathermood-server 2

Application versions

Saved configurations

Elastic Beanstalk > Applications > weathermood-server

Application 'weathermood-server' environments

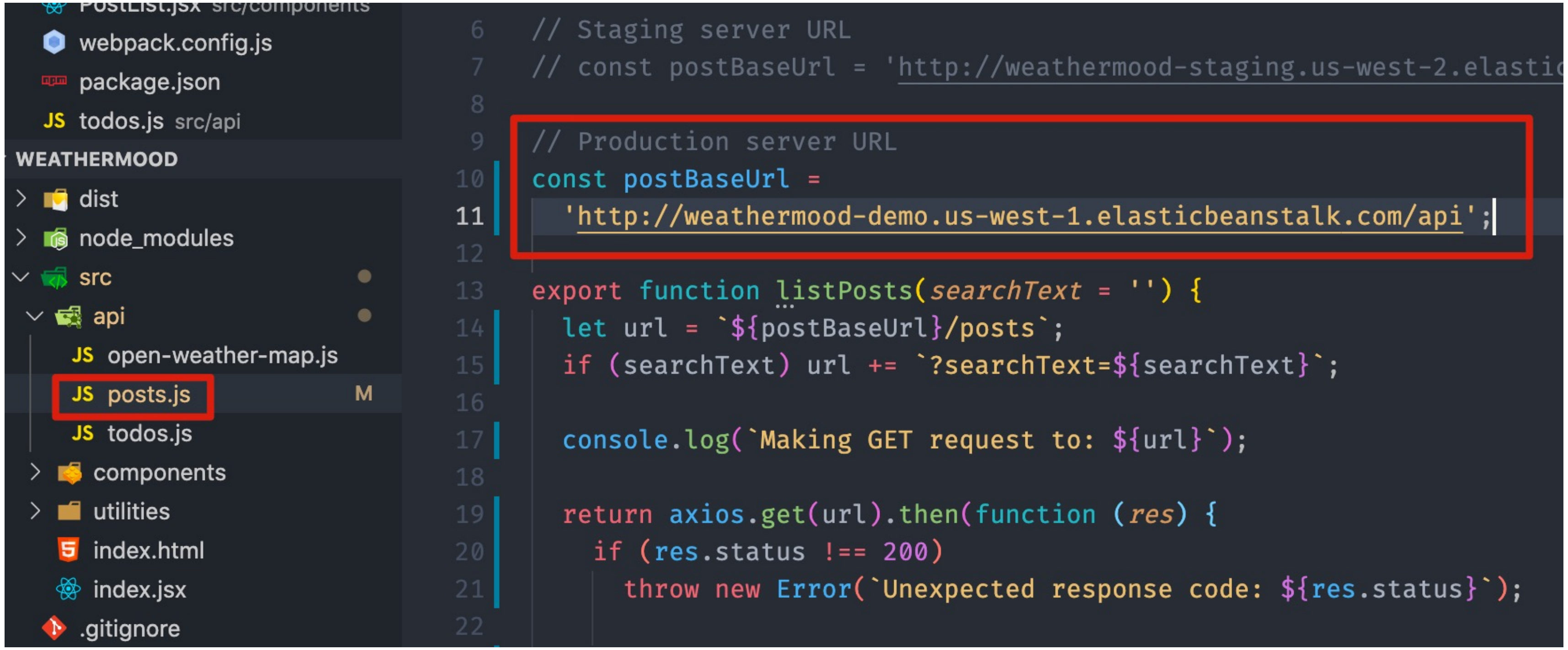
Filter results matching the display values

Environment name ▲	Health ▽	Date created ▽	Last modified ▽	URL ▽
weathermood-server-dev	Ok	2021-04-24 21:40:11 UTC+0800	2021-04-24 21:44:57 UTC+0800	3 weathermood-demo.us-east-1.elasticbeanstalk.com

Deploy to AWS with AWS EB CLI

Step4: Go to client project and paste the URL to postBaseUrl

- Reminder: Don’t forget the /api part of the URL



```
// Staging server URL
// const postBaseUrl = 'http://weathermood-staging.us-west-2.elastic

// Production server URL
const postBaseUrl =
  'http://weathermood-demo.us-west-1.elasticbeanstalk.com/api';

export function listPosts(searchText = '') {
  let url = `${postBaseUrl}/posts`;
  if (searchText) url += `?searchText=${searchText}`;

  console.log(`Making GET request to: ${url}`);

  return axios.get(url).then(function (res) {
    if (res.status !== 200)
      throw new Error(`Unexpected response code: ${res.status}`);

```

Deploy to AWS with AWS EB CLI

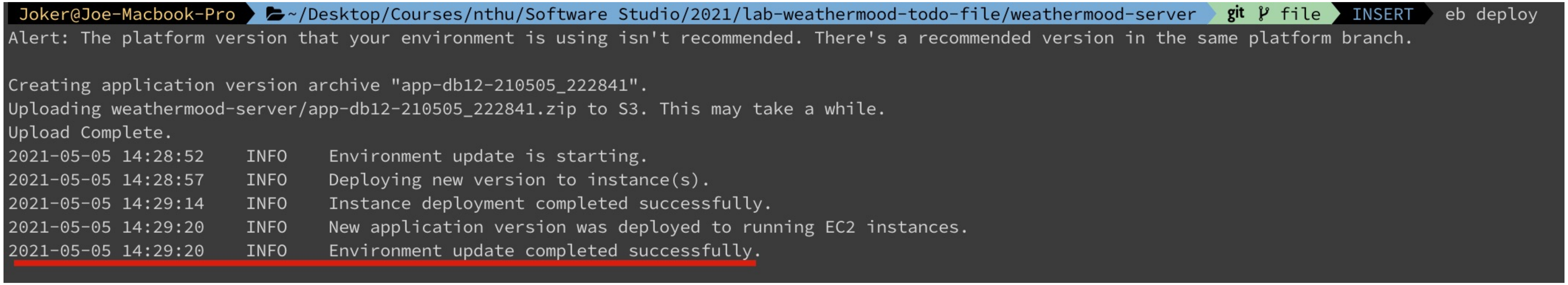
Step5: Combine the client and server project

- Build client project - `npm run build`
- Copy *dist folder* from clint to server

Deploy to AWS with AWS EB CLI

Step6: Deploy the app to the environment

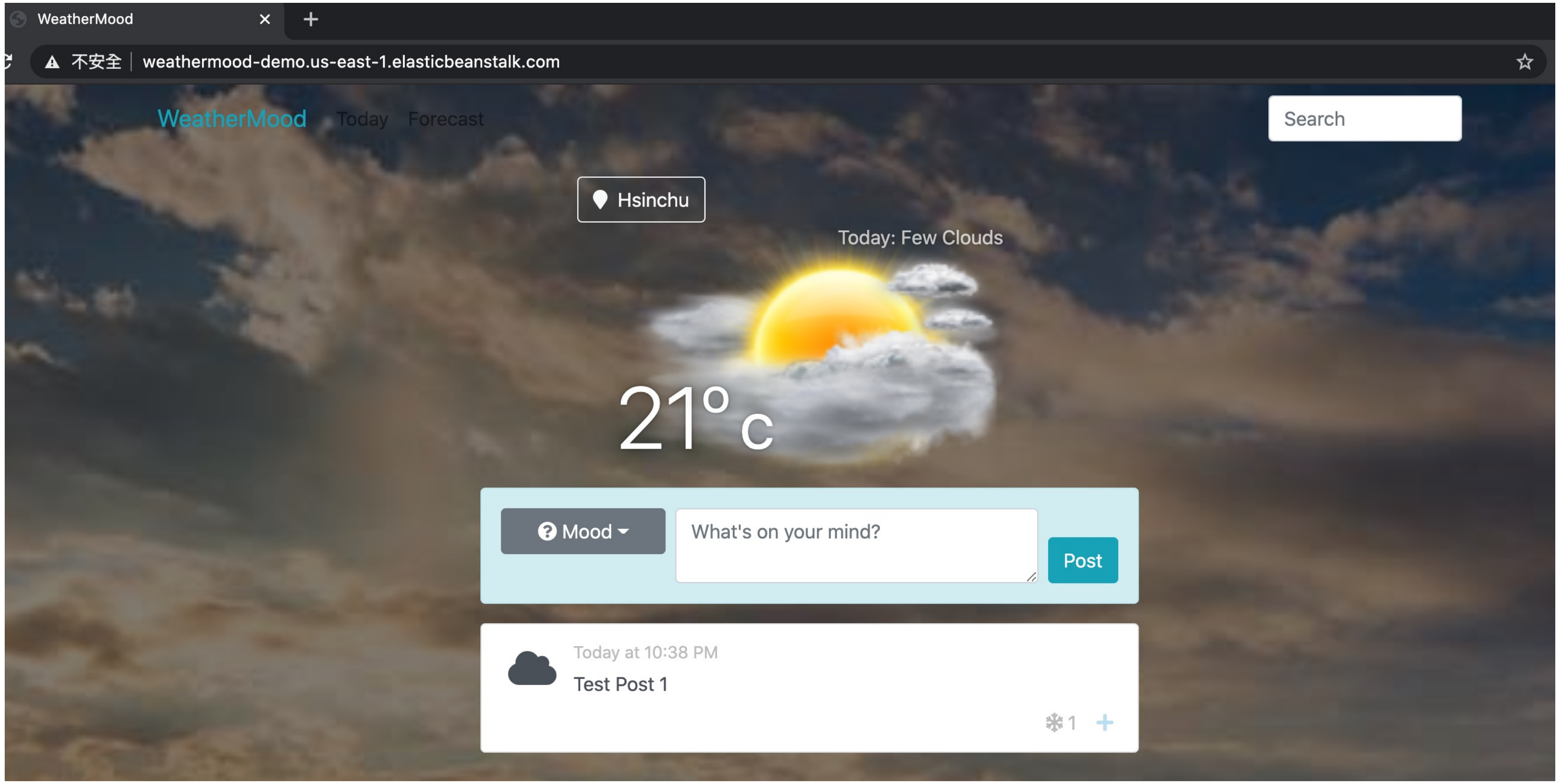
- Commit before deploy: Only committed change would be deployed
- eb deploy <env>



```
 Joker@Joe-Macbook-Pro  ~/Desktop/Courses/nthu/Software Studio/2021/lab-weathermood-todo-file/weathermood-server  git  file  INSERT  eb deploy
Alert: The platform version that your environment is using isn't recommended. There's a recommended version in the same platform branch.

Creating application version archive "app-db12-210505_222841".
Uploading weathermood-server/app-db12-210505_222841.zip to S3. This may take a while.
Upload Complete.
2021-05-05 14:28:52    INFO    Environment update is starting.
2021-05-05 14:28:57    INFO    Deploying new version to instance(s).
2021-05-05 14:29:14    INFO    Instance deployment completed successfully.
2021-05-05 14:29:20    INFO    New application version was deployed to running EC2 instances.
2021-05-05 14:29:20    INFO    Environment update completed successfully.
```

Deploy to AWS with AWS EB CLI

Check result: Enter URL into browser



weathermood-demo.us-east-1.elasticbeanstalk.com

What's on your mind?

Today at 10:38 PM

Test Post 1