

AWS Config - Hands-on Exercise

er

- [Introduction](#)
- [Part 1 – IAM Users](#)
 - [Goal](#)
 - [Part 1A - Login and get to Config](#)
 - [Part 1B – Find "your" IAM user](#)
 - [Part 1C – Whitelist "your" IAM user](#)
- [Part 2 – S3 Buckets](#)
 - [Goal](#)
 - [Part 2A - Login and go to Config](#)
 - [Part 2B – Find "your" S3 bucket](#)
 - [Part 2C – Whitelist "your" bucket](#)
- [Part 3 – Wait for Config Rule re-evaluation](#)
 - [Goal](#)
 - [Part 3A – Find "your" resources in Config again](#)
 - [Part 3B – Be prepared for delayed gratification](#)

Introduction

This hands-on exercise shows how to work with non-compliant resources in AWS Config and how to whitelist them for Config.

Part 1 – IAM Users

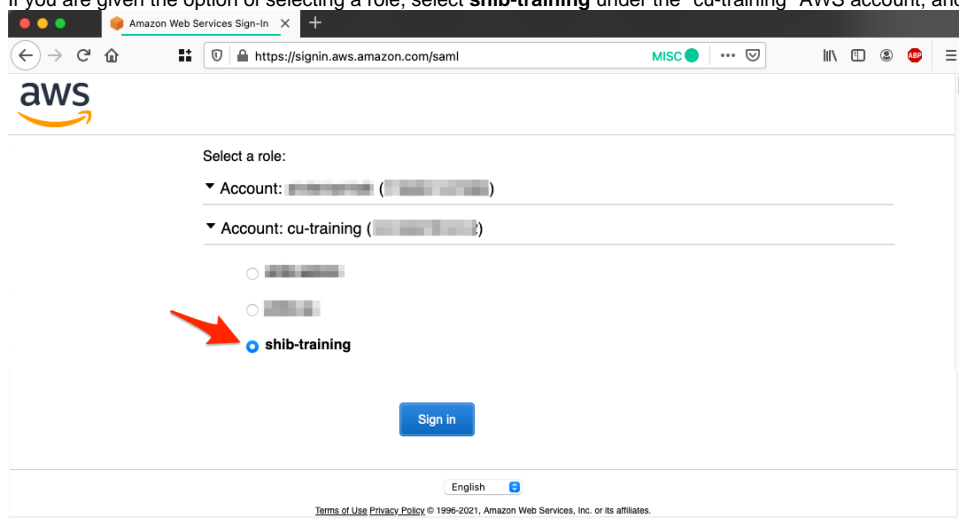
Goal

Although the use of IAM users in Cornell AWS accounts is discouraged in most situations, there are some valid use-cases where IAM users are necessary. The Cornell AWS Config rule **251-MED-no-iam-users-except-whitelist** labels all IAM users as non-compliant unless they are specifically whitelisted.

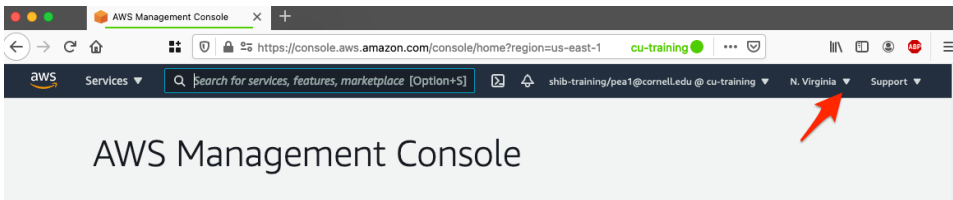
For this exercise, an IAM user was previously created in cu-training for each training participant. In this exercise, you will find "your" IAM user there and whitelist it for the **251-MED-no-iam-users-except-whitelist** Config rule.

Part 1A - Login and get to Config

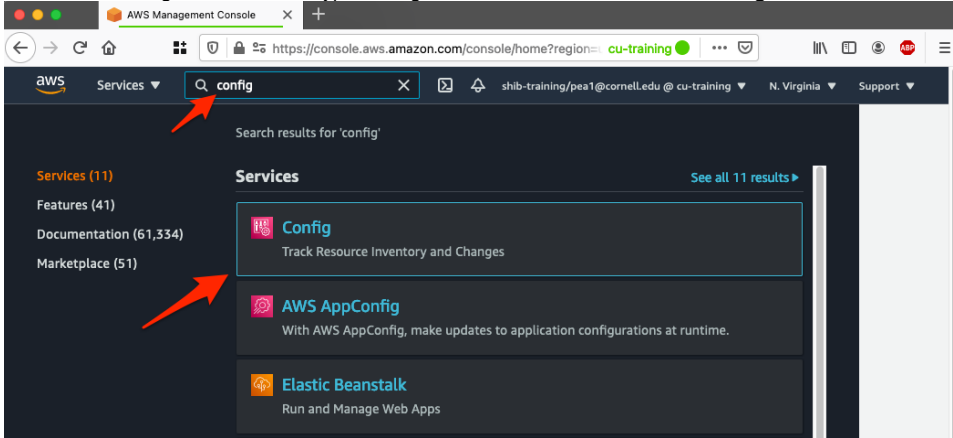
1. Login to the cu-training AWS account using traditional Shibboleth login.
 - a. Use this link to initiate login: <https://signin.aws.cucloud.net/>
 - This will start the usual process for Cornell Two-Step Login process. Complete your two-step login.
 - Once you have finished with DUO, you will be in one of two places. Take your next steps based on where you end up.
 - b. If you are given the option of selecting a role, select **shib-training** under the "cu-training" AWS account, and click on **Sign in**.



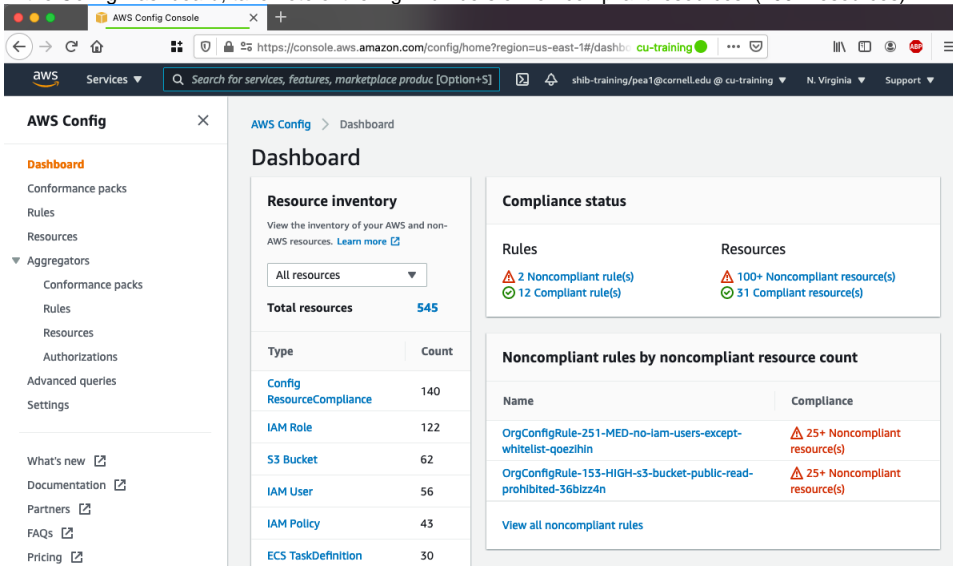
1. Once in the AWS Management Console, check which AWS region your console is pointed at. You want "N. Virginia". If your console is in any other region, change it to "US East (N. Virginia) us-east-1".



2. In the AWS Management Console, type "config" in the search box and click on **Config** under **Services**.



3. In the Config Dashboard, take note of the high numbers of non-compliant resources. (100+ resources)



Part 1B – Find "your" IAM user

1. Click on **Resources** from the left-hand navigation panel in the Config console.
2. Enter the "netid" form of your Cornell email address (e.g., netid@cornell.edu) in the **Resource identifier** search field and hit "enter" on your keyboard. This will start the search for "your" IAM user.

Resource Inventory

Search existing or deleted resources recorded by AWS Config. For a specific resource, view the resource details, configuration timeline, or compliance timeline. The resource configuration timeline allows you to view all the configuration items captured over time for a specific resource. The resource compliance timeline allows you to view compliance status changes. To query your resource configurations, use the [advanced SQL query editor](#).

Resources

View detailsResource Timeline

Resource categoryAll resource categoriesResource typeAll resource typesComplianceAny compliance status

Resource identifier - optionalpea1@cornell.eduXInclude deleted resources

< 1 > ⚙

Resource Identifier	Type	Compliance
pea1@cornell.edu	IAM User	⚠ Noncompliant

3. Config should show one search result, listing an IAM user named like "netid@cornell.edu". That IAM User resource will be labelled as non-compliant.

Resource Identifier	Type	Compliance
pea1@cornell.edu	IAM User	⚠ Noncompliant

4. Click on the IAM user name (i.e., netid@cornell.edu) to drill into that resource.
5. Review the **Rules** at the bottom to confirm that "your" IAM user is indeed non-compliant with respect to the **251-MED-no-iam-users-except-whitelist** rule.

AWS Config > Resources > pea1@cornell.edu

pea1@cornell.edu

Resource TimelineManage Resource

▼ Details

Resource name
pea1@cornell.edu

Resource type
AWS::IAM::User

Resource ID
AID: [REDACTED]

Amazon resource name
arn:aws:iam:: [REDACTED] :user/example/delete/pea1@cornell.edu

Availability zone
Not Applicable

Created on
April 12, 2021 8:16 AM

User Name
pea1@cornell.edu

Inline Policy Details
-

► View Configuration Item (JSON)

Rules appliedTags

Rules

View detailsEdit ruleActionsAdd rule

Any status

< 1 > ⚙

Name	Remediation action	Type	Compliance
OrgConfigRule-152-HIGH-no-iam-users-with-password-jwhckxhj	Not set	-	✔ Compliant
OrgConfigRule-304-LOW-iam-user-unused-credentials-check-ywaxmu8d	Not set	-	✔ Compliant
OrgConfigRule-252-MED-access-keys-rotated-jxtdpgwk	Not set	-	✔ Compliant
OrgConfigRule-251-MED-no-iam-users-except-whitelist-qoezihh	Not set	-	⚠ Noncompliant

6. In the top right of that page, click on **Manage Resource**.

Part 1C – Whitelist "your" IAM user

1. You should now be viewing "your" IAM user in the the IAM console.
2. Click on the **Tags** tab.

The screenshot shows the AWS IAM console for user `pea1@cornell.edu`. The **Tags** tab is selected, indicated by a red arrow. The summary shows the user's ARN, path, and creation time. Below the summary, the **Permissions** section shows one policy applied: `AWSDenyAll`.

3. Click on **Add tags**.
4. Add a tag with the following settings, and click **Save changes**:
 - a. Key: `cit:config:251-MED-no-iam-users-except-whitelist`
 - b. Value: `exception`

The screenshot shows the **Tags for pea1@cornell.edu** page. A new tag is being added with the key `cit:config:251-MED-no-iam-users-except-whitelist` and the value `exception`. The **Save changes** button is highlighted with a red arrow.

The screenshot shows the **Summary** tab for user `pea1@cornell.edu`. The **Tags** section shows the newly added tag with the key `cit:config:251-MED-no-iam-users-except-whitelist` and the value `exception`.

⚠ At this point in a typical Config workflow, you would find the **251-MED-no-iam-users-except-whitelist** Config Rule and trigger re-evaluation of the rule to confirm that your whitelisting had the desired effect (i.e. making the IAM user compliant for that rule). However, the Config API has a very, very threshold for the number of times that you can invoke reevaluations. Therefore, the exercise leader will trigger re-evaluation just a few times during this hands-on session.

Part 2 – S3 Buckets

Goal

To be conservative, the Cornell **153-HIGH-s3-bucket-public-read-prohibited** Config Rule flags any S3 buckets that are publicly readable. However, there are valid use cases where you want S3 bucket contents to be publicly readable (e.g., public web resources).

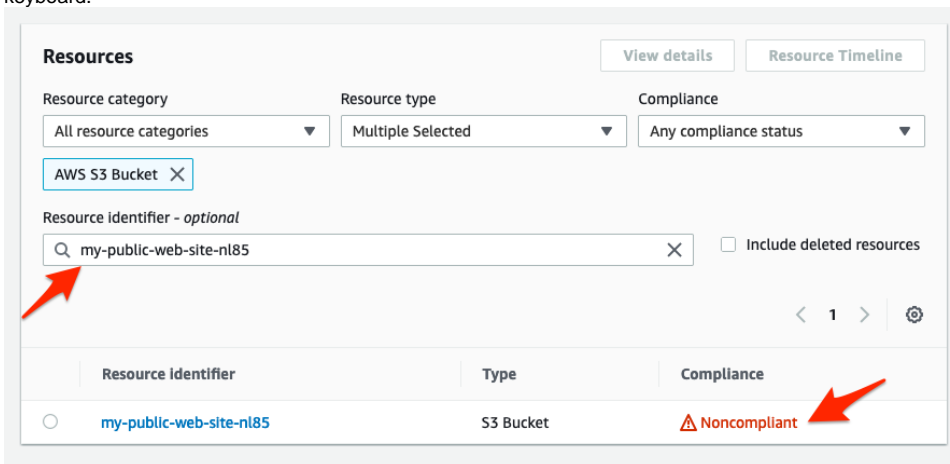
For this exercise, an S3 bucket was previously created in cu-training for each training participant. In this exercise, you will find "your" S3 web site bucket and whitelist it for the **153-HIGH-s3-bucket-public-read-prohibited** Config rule.

Part 2A - Login and go to Config

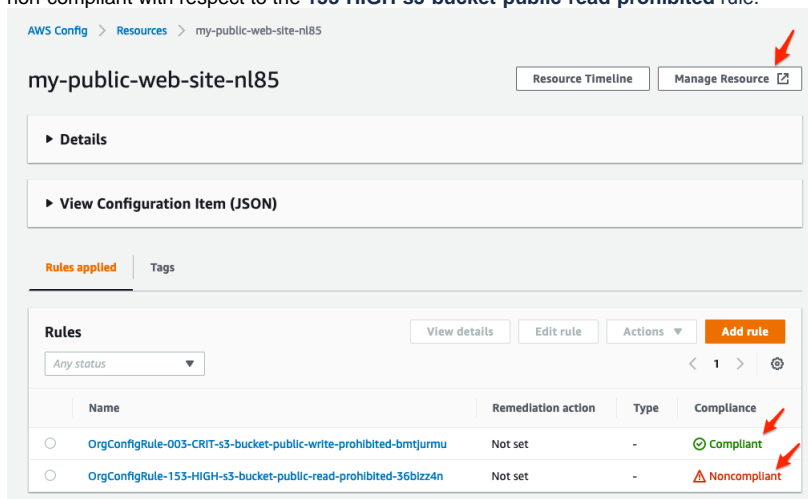
1. If you aren't logged in to the cu-training account with role **shib-training**, follow the instructions in **Part 1A** above to login and navigate to the Config console.

Part 2B – Find "your" S3 bucket

1. Click on **Resources** from the left-hand navigation panel in the Config console.
2. In the **Resource identifier** search field, enter "your" S3 bucket name using this pattern **my-public-web-site-NETID**, and hit "enter" on your keyboard.



3. Config should show one search result, listing an S3 bucket user named like **my-public-web-site-NETID**. That bucket will be labelled as non-compliant.
4. Click on the bucket name to drill into the Config details for that resource.
5. Review the **Rules** at the bottom of the page. They will show that the bucket is
 - a. compliant with respect to the **003-CRIT-s3-bucket-public-write-prohibited** rule, but
 - b. non-compliant with respect to the **153-HIGH-s3-bucket-public-read-prohibited** rule.



6. In the top right of the details page, click on **Manage Resource**. This will take you to the S3 console for that bucket.

Part 2C – Whitelist "your" bucket

1. In the S3 console, with "your" bucket selected, click on the **Properties** tab and scroll down to **Tags**.

Amazon S3 > my-public-web-site-nl85

my-public-web-site-nl85

Publicly accessible

Objects | **Properties** | Permissions | Metrics | Management | Access Points

Bucket overview

AWS Region US East (N. Virginia) us-east-1	Amazon Resource Name (ARN) arn:aws:s3::my-public-web-site-nl85	Creation date April 12, 2021, 08:27:29 (UTC-04:00)
---	---	---

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning
Disabled

Multi-factor authentication (MFA) delete
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

Disabled

Tags (0)

Track storage cost or other criteria by tagging your bucket. [Learn more](#)

Key	Value
No tags associated with this resource.	

2. Click **Edit** in the tags section.
3. Click on **Add tag**.

Amazon S3 > my-public-web-site-nl85 > Edit bucket tagging

Edit bucket tagging

Tags

Track storage cost or other criteria by tagging your bucket. [Learn more](#)

No tags associated with this bucket.

4. Add a tag with the following details, and click on **Save changes**:
 - a. Key: **cit:config:153-HIGH-s3-bucket-public-read-prohibited**

b. Value: **exception**

Amazon S3 > my-public-web-site-nl85 > Edit bucket tagging

Edit bucket tagging

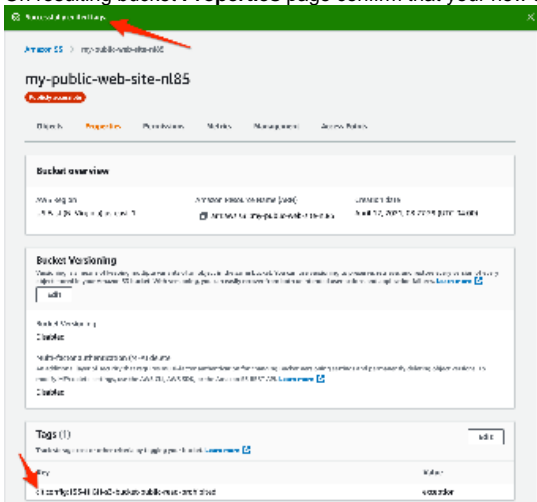
Tags
Track storage cost or other criteria by tagging your bucket. [Learn more](#)

Key: HIGH-s3-bucket-public-read-prohibited Value - optional: exception

Add tag Remove

Cancel Save changes

5. On resulting bucket **Properties** page confirm that your new tag is shown as one of the bucket tags.



Part 3 – Wait for Config Rule re-evaluation

Goal

In a typical Config workflow you would whitelist, reconfigure, or delete resources that Config has flagged and then tell Config to re-evaluate the relevant Rules. However, there are complicating factors.

Part 3A – Find "your" resources in Config again

1. Use the Config console to open two browser windows – one with Config details for "your" IAM user, and one with the Config details for "your" S3 bucket.
 - Each of those resources started out as non-compliant to one Config rule.
2. Wait until the exercise leader says that re-evaluation has been triggered and completed.
3. Once you get the go-ahead, refresh each of the resource Config details pages.

Part 3B – Be prepared for delayed gratification

- In all likelihood one, if not both, of "your" resources remain flagged as non-compliant. 😞
- One reason for this is that Config evaluations operate on snapshots of resource configurations, not from an instantaneous reading of the resource details.
- Another factor that seems to come into play is that Config evaluation results themselves are cached and sometimes you may be looking at results from the previous evaluation.
- The secret for maintaining your sanity with Config is to check back with Config for new results on the day after you make resource configuration changes. What you see after 24 hours is typically a meaningful set of results.



If you wanted to force Config to evaluate a Rule within your own AWS account, you would use the **Re-evaluate** action, as shown below.

[AWS Config](#) > [Rules](#) > OrgConfigRule-153-HIGH-s3-bucket-public-read-prohibited-36bizz4n



This rule has been created by [config-multiaccountsetup.amazonaws.com](#)

This is a service-linked AWS Config rule (SLR) and it's a unique type of managed config rule that supports other AWS services to create AWS Config rules in your account. You cannot edit or delete these rules if you are subscribed to AWS services that these rules are linked to. [Read more about Service-Linked AWS Config Rules.](#)

OrgConfigRule-153-HIGH-s3-bucket-public-read-prohibited-36bizz4n

Actions ▲

Manage remediation

Re-evaluate

Delete results

Delete rule

▼ Rule details

Description

[CIT custom rule] Ensure that S3 buckets are not publicly readable, except those explicitly whitelisted.

Config rule ARN

arn:aws:config:us-east-1:██████████:config-rule/aws-service-rule/config-multiaccountsetup.amazonaws.com/config-rule-l2dyn

Trigger type

Periodic: 24 hours

Scope of changes

Resources

Last successful evaluation

✓ April 12, 2021 1:58 PM

▼ Resources in scope

View details

Remediate



Noncompliant ▼

< 1 2 ... > ⚙

	ID	Type	Status	Annotation	Compliance
<input type="radio"/>	my-public-web-site-██████████	S3 Bucket	-	Trusted Advisor marks this bucket with status "Yellow" and it is not whitelisted.	⚠ Noncompliant
<input type="radio"/>	my-public-web-site-██████████	S3 Bucket	-	Trusted Advisor marks this bucket with status "Yellow" and it is not whitelisted.	⚠ Noncompliant
<input type="radio"/>	my-public-web-site-██████████	S3 Bucket	-	Trusted Advisor marks this bucket with status "Yellow" and it is not whitelisted.	⚠ Noncompliant
<input type="radio"/>	my-public-web-site-██████████	S3 Bucket	-	Trusted Advisor marks this bucket with status "Yellow" and it is not whitelisted.	⚠ Noncompliant