

# Tagging AWS Resources in Substrate

---

Downloaded from Epic Games Confluence

Date: 2025-07-12 04:07:58

Original URL: <https://confluence-epicgames.atlassian.net/wiki/spaces/CDE/pages/81068305>

## Document Level Classification

100

- [What are tags?](#)
- [Why you should tag?](#)
- [What happens if tags are not applied?](#)
- [Required Tags](#)
- [Tag Descriptions](#)
  - [Service Tag](#)
    - [Examples](#)
  - [Owner Tag](#)
    - [Examples](#)
  - [Contact Tag](#)
    - [Examples](#)
  - [EUID Tag](#)
    - [Examples](#)
      - [Services in Backstage](#)
      - [Service/Resource with no Backstage Entry](#)
- [Name](#)
  - [Examples](#)
  - [Note](#)

- [IdleScanExempted](#)
  - [Example](#)
  - [Note](#)
- [NonGravitonUseCase](#)
  - [Example](#)
- [IntentionallyPublic](#)
  - [Example](#)
  - [Note](#)
- [SecurityTooling](#)
  - [Example](#)

## What are tags?

Tags are key and value pairs that act as metadata for organizing your AWS resources. Tags can help you manage, identify, organize, search for, and filter resources. You can create tags to categorize resources by purpose, owner, environment, or other criteria.

Each tag has two parts:

- A tag key (for example, CostCenter, Environment, or Project). Tag keys are case sensitive.
- A tag value (for example, 111122223333 or Production). Like tag keys, tag values are case sensitive.

You can use tags to categorize resources by purpose, owner, environment, or other criteria.

Reference AWS Docs: <https://docs.aws.amazon.com/tag-editor/latest/userguide/tagging.html>

## Why you should tag?

Standardizing on a set of specific tags will make it easier for Epic to perform cost attribution across team and product boundaries, identify and remediate unused and orphaned resources, as well as better enable

account and resource ownership for operations and incident response across the company.

## What happens if tags are not applied?

Any cloud resources that do not adhere to this policy and are missing required tags will be removed. Notifications will be sent to account stewards and resource owners, if identifiable through SSO or contact tags.

**Note:** Existing tags do not need to be removed. We are aware that some teams were using other tags ex. "Role", "Tier", "EpicFinance:Environment", "EpicFinance:Product", "EpicFinance:Owner", etc. These tags do not need to be deleted.

## Required Tags

InfoSec proposes the following tags be used on all AWS resources that support tagging:

<b>Service</b>	Canonical name of your service which should tie back to the product in the product registry (i.e. aws-event-monitoring, uam, aop)
<b>Owner</b>	The team the resource belongs to (i.e. epicgames.infosec.cloudsec)
<b>Contact</b>	Team email who owns the resource (i.e. <a href="mailto:infosec@epicgames.com">infosec@epicgames.com</a> )
<b>EUID</b>	Backstage EUID for service (i.e. euid-w2gys2pn)

For resources not in backstage please provide an explanation for why it's not (i.e. "Bucket holds terraform state for multiple projects")

This is a deliberate attempt to not use Epic anywhere in the tag name. Since Epic regularly acquires other companies via mergers and acquisitions we want to be able to roll these standards out to those companies as well. Furthermore, special characters in the tags such as a colon make it difficult to query for those tags in systems like AWS Config.

Reference <https://pages.github.ol.epicgames.net/Cloud-Governance/cloud-gov-docs/docs/cloud-governance/required-tagging/> for Required tags.

Reference <https://pages.github.ol.epicgames.net/Cloud-Governance/cloud-gov-docs/docs/cloud-governance/optional-tagging/> for Optional tags.

## Optional Tags

<b>Name</b>	The name tag will identify the resource (i.e. root-login-monitor-lambda, uam-frontend-webserver, aop-config-bucket)
<b>IdleScanExempted</b>	Some cloud resources can appear idle from an outsider perspective. This results in active resources appearing in idle resource reports. Teams can use the <code>IdleScanExempted</code> tag to inform the org that a resource is not truly idle. This will stop systems such as Savvy from identifying the resource as idle. Set the tag value to a description about why it is not a real idle resource.

<b>NonGravitonUseCase</b>	<p>This tag is used to bypass controls that enforce Graviton/ARM instance types. While Graviton is a less expensive option we want to enable developers to use other workloads if needed. When using the <code>NonGravitonUseCase</code> tag the value should briefly explain the use case.</p>
<b>IntentionallyPublic</b>	<p>Certain resources such as S3 buckets typically aren't supposed to be accessible by the general public on the internet. However there are valid use cases where exposing these types of resources is required. When needing to expose resources that are not typically public directly to the public please add the <code>IntentionallyPublic</code> tag and set the value to a brief description as to why.</p>
<b>SecurityTooling</b>	<p>Infosec maintains a number of security tools that can often can false positives within cloud environments. In order to reduce noise in our internal tooling and properly document this tag should be set. The value should be provided by the Infosec Security Engineering team as the value should not be arbitrarily set.</p>

# Tag Descriptions

## Service Tag

### Required

The **Service** tag identifies the name of your service. It should align with the name of your service in Epic's service catalog.

### Examples

The example services selected are arbitrary examples.

- [AWS Account Configurator](#)
- [EOS Identity Provider Service](#)
- [Creator Portal](#)

## Owner Tag

### Required

The **Owner** tag identifies the business, team, and sub-team that owns the resource. It must be structured in the following way:

**<business>.<team>.<sub-team>**

This structure is important as it enables upstream tooling to understand cost ownership.

### Examples

- epicgames.infosec.cloud-security
- superawesome.infrastructure.cloud
- epicgames.ecosec.anticheat

## Contact Tag

### Required

The **Contact** tag identifies the responsible party to contact regarding the resource. This tag must be a team email.

## Examples

- [core-tech-gov-eng@epicgames.com](mailto:core-tech-gov-eng@epicgames.com)
- [infosec-seceng@epicgames.com](mailto:infosec-seceng@epicgames.com)
- [pulse-rr@epicgames.com](mailto:pulse-rr@epicgames.com)

## EUID Tag

### Required

The **EUID** tag correlates to the EUID in Epic's service catalog. For resources that do not have a clear EUID may use this field to explain why the resource is not in backstage.

## Examples

### Services in Backstage

- [AWS Account Configurator](#)
  - EUID: euid-hvljzsf
- [EOS Identity Provider Service](#)
  - EUID: euid-w9i526tc
- [Creator Portal](#)
  - EUID: euid-vblgzdvk

### Service/Resource with no Backstage Entry

Resources without a proper EUID value should explain why instead

## Name

### Optional

The Name tag should identify the individual resource for the system.

## Examples

Example resources here do not really exist.

- prod-account-configurator-role-creator
- dev-eos-id-service-dynamo-lock-table
- prod-creator-portal-artifact-storage

## Note

This tag was previously required but based on the feedback received it was moved to optional.

## IdleScanExempted

### Optional

Some cloud resources can appear idle from an outsider perspective. This results in active resources appearing in idle resource reports. Teams can use the `IdleScanExempted` tag to inform the org that a resource is not truly idle. This will stop systems such as Savvy from identifying the resource as idle. Set the tag value to a description about why it is not a real idle resource.

### Example

`IdleScanExempted`: `RDS HA Node`

## Note

There used to be a companion tag `IdleScanExemptedUntil`. This tag is no longer required. Governance tooling will respect the `IdleScanExempted` tag regardless if the companion tag is present or not.

## NonGravitonUseCase

### Optional



This tag is used to bypass controls that enforce Graviton/ARM instance types. While Graviton is a less expensive option we want to enable developers to use other workloads if needed. When using the `NonGravitonUseCase` tag the value should briefly explain the use case.

## Example

```
NonGravitonUseCase : Graviton does not support burst throughput  
required
```

# IntentionallyPublic

## Optional

Certain resources such as S3 buckets typically aren't supposed to be accessible by the general public on the internet. However there are valid use cases where exposing these types of resources is required. When needing to expose resources that are not typically public directly to the public please add the `IntentionallyPublic` tag and set the value to a brief description as to why.

## Example

```
IntentionallyPublic : Serves public assets
```

## Note

As we review reasons for resources that use this tag we may reach out and suggest alternative implementations. This tag does not need to be applied to resources like load balancers that are typically used to support external ingress.

# SecurityTooling

## Optional

Infosec maintains a number of security tools that can often can false positives within cloud environments. In order to reduce noise in our

internal tooling and properly document this tag should be set. The value should be provided by the Infosec Security Engineering team as the value should not be arbitrarily set.

## Example

SecurityTooling: <\_provided\_by\_infosec\_>

- [How To Add Required Tags to Resources in Substrate](#)

---

### Page Information:

Page ID: 81068305

Space: Cloud Developer Platform

Downloaded: 2025-07-12 04:07:58