

Clean-up failure: Elasticache default.iam-user / default.iam-user-group

This page describes a specific account clean-up failure caused by AWS-managed default Elasticache resources, how to recognise it in the logs, and how to fix it.

Symptom

The account clean-up (AWS Nuke) CodeBuild job fails, and the account is moved to **Quarantine**. In the nuke log, the scan completes with only **two** nukeable resources, both of which are AWS-managed Elasticache defaults:

```
Scan complete: 279 total, 2 nukeable, 277 filtered.
```

```
would remove ElasticacheUserGroup default.iam-user-group
would remove ElasticacheUser      default.iam-user
```

Nuke then retries the deletion a few times and finally aborts with a fatal error:

```
InvalidParameterValue: User group id must begin with a letter; must contain only ASCII
letters, digits, and hyphens; and must not end with a hyphen or contain two consecutive
hyphens.
```

```
InvalidParameterValue: User id must begin with a letter; must contain only ASCII letters,
digits, and hyphens; and must not end with a hyphen or contain two consecutive hyphens.
```

```
fatal: failed
```

Cause

`default.iam-user` and `default.iam-user-group` are **AWS-managed default Elasticache resources** that are created automatically when ElastiCache (or its service-linked role) is first used in a region. They **cannot be deleted** — the ElastiCache API rejects the delete call because the resource name contains a dot (`.`), which violates the naming rules the delete API enforces.

AWS Nuke has built-in handling that automatically skips the plain `default` Elasticache user:

```
filtered: cannot delete default user    default    ElasticacheUser
```

However, nuke does **not** have built-in handling for the IAM-authentication defaults (`default.iam-user` / `default.iam-user-group`). These only get skipped if the **nuke configuration explicitly filters them**. If the config in effect for the account is missing those filters, nuke treats them as deletable, repeatedly fails, and the whole clean-up aborts — even when every other resource in the account filtered cleanly.

“ **Note:** This typically appears after AWS enables ElastiCache IAM-auth defaults in an account/region. Because they are created automatically, an account that cleaned up fine previously can start failing without any user action.

Fix

Add the following filters to the **Nuke configuration** used by the failing deployment (stored in AWS AppConfig as the *NukeConfig* hosted configuration). The filters can be placed in the relevant `presets` / global filter block or in the per-account `filters` block:

```
ElasticacheUser:
  # AWS-managed default Elasticache user; cannot be deleted (the delete API
  # rejects the "default.iam-user" name), causing nuke to fail fatally.
  - type: exact
    value: default.iam-user
  - property: UserName
    type: exact
    value: default.iam-user
ElasticacheUserGroup:
  # AWS-managed default Elasticache user group; not deletable.
  - type: exact
    value: default.iam-user-group
```

Once the filters are in place, the scan reports these resources as `filtered` instead of `nukeable`, and the clean-up completes successfully.

Verifying the fix

1. Confirm the live **NukeConfig** AppConfig hosted configuration version contains the `ElasticacheUser` and `ElasticacheUserGroup` filters above.
2. Re-run the clean-up for the quarantined account (see **Resolving clean-up failures**).

3. In the new nuke log, confirm the two resources now appear as filtered, for example:

```
filtered: filtered by config  default.iam-user      ElasticacheUser
filtered: filtered by config  default.iam-user-group ElasticacheUserGroup
```

4. Confirm the job finishes without `fatal: failed` and the account returns to the pool.

Revision #3

Created 2026-10-27 19:06:51 UTC by Doco

Updated 2026-10-27 19:09:54 UTC by Paul