## ■ Terraform Resource Re-Creation: taint vs replace

#### 1. Overview

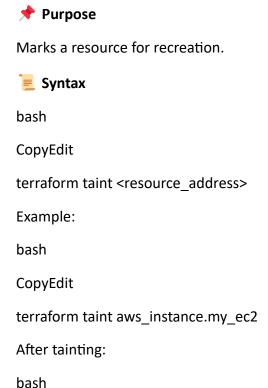
Sometimes, you need Terraform to **recreate** a resource without changing its configuration. This might happen when:

- The resource is corrupted.
- Manual changes outside Terraform broke it.
- You want to reset it for testing.

Terraform offers two main ways to do this:

- **terraform taint** marks a resource as "tainted" so it will be recreated in the next terraform apply.
- **terraform apply -replace** directly replaces a resource without marking it tainted first.

## 2. terraform taint (Legacy but still available)



Terraform will destroy and recreate the resource.

CopyEdit

terraform apply

# Advantages

- Explicitly marks a resource for replacement.
- Easy to understand and use.

### Limitations

- Requires two commands (taint then apply).
- Will be **deprecated in future versions** -replace is preferred.

## 3. terraform apply -replace



Directly tells Terraform to recreate a resource in one command.



bash

CopyEdit

terraform apply -replace=<resource\_address>

Example:

bash

CopyEdit

terraform apply -replace=aws\_instance.my\_ec2

Terraform will:

- 1. Destroy the resource.
- 2. Create a new one in a single apply run.

# Advantages

- Single step process.
- Preferred in modern Terraform workflows.
- No separate taint step.

#### 4. When to Use

#### Scenario

#### **Recommended Method**

Testing resource recreation -replace

Resource is corrupted -replace

Legacy scripts/workflows still using taint taint (temporary)

Debugging drift issues -replace

## 5. Example

## **Terraform Code (main.tf)**

```
CopyEdit
```

hcl

```
provider "aws" {
  region = "ap-south-1"
}

resource "aws_instance" "my_ec2" {
  ami = "ami-0e306788ff2473ccb"
  instance_type = "t2.micro"
```

#### **Recreate the Instance**

bash

}

CopyEdit

terraform apply -replace=aws\_instance.my\_ec2

Output:

vbnet

CopyEdit

Plan: 1 to add, 0 to change, 1 to destroy.

#### 6. Best Practices

- Always run terraform plan before replacing resources.
- Be cautious with **stateful** resources (databases, S3 buckets).
- Use -replace in CI/CD pipelines instead of taint.
- For multiple replacements:

bash

## CopyEdit

terraform apply -replace=aws\_instance.my\_ec2 -replace=aws\_s3\_bucket.my\_bucket

## 7. Summary Table

Feature terraform taint terraform apply -replace

Steps Needed 2 (taint + apply) 1 (apply -replace)

Modern Usage 

X Legacy 
✓ Preferred

Automation Friendly Less More

Explicit Marking Yes No (direct replace)