Lab Exercise 18- Scanning IaC Templates for Vulnerabilities

Objective

- Learn how to scan Infrastructure as Code (IaC) templates for security vulnerabilities.
- Use open-source IaC security tools to detect misconfigurations.
- Understand common risks such as public access, unencrypted resources, and insecure network rules.

Prerequisites

- A Linux/Windows/Mac machine with:
    - Terraform installed (for sample IaC)
    - **Checkov** (pip install checkov) or **tfsec** (brew install tfsec or binary download)

```
C:\Users\hp\Desktop\Notes\DevSecOps-Lab\terraform-demo\lab-18>python3 -m pip install checkov
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: checkov in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.12_qbz5n2kfra8p0\localc
ache\local-packages\python312\site-packages (3.2.471)
Requirement already satisfied: bc-python-hcl2==0.4.3 in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.12_qbz5n2
kfra8p0\localcache\local-packages\python312\site-packages (from checkov) (0.4.3)
Requirement already satisfied: bc-detect-secrets==1.5.45 in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.12_qb
z5n2kfra8p0\localcache\local-packages\python312\site-packages (from checkov) (1.5.45)
```

- Git installed (optional, for version control of IaC templates)

Step 1: Create an Insecure IaC Template

Create a file named main.tf with the following Terraform code:

```
provider "aws" {
  region = "us-east-1"
}
resource "aws_s3_bucket" "insecure_bucket" {
  bucket = "my-insecure-bucket-lab"
  acl    = "public-read"
}
resource "aws_security_group" "insecure_sg" {
  name        = "insecure-sg"
  description = "Allow all inbound traffic"
  ingress {
    from_port   = 0
    to_port     = 65535
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }
```

```
}
```

Step 2: Scan the Template with Checkov

Run Checkov on the current directory:

```
checkov -d .
```

Expected Findings:

- Public S3 bucket access (public-read)
- Security group open to all inbound traffic

Expected Findings:

- Warns about S3 bucket without encryption
- Flags open Security Group rules

Step 4: Review the Report

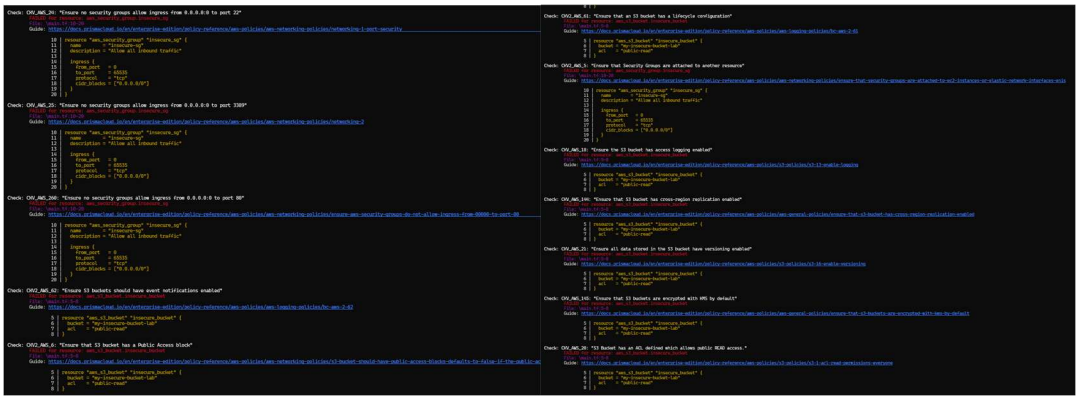
Example output (Checkov):

```
Check: CKV_AWS_20: "S3 Bucket allows public read access"

    FAILED for resource: aws_s3_bucket.insecure_bucket

Check: CKV_AWS_260: "Security group allows ingress from 0.0.0.0/0"

    FAILED for resource: aws_security_group.insecure_sg
```

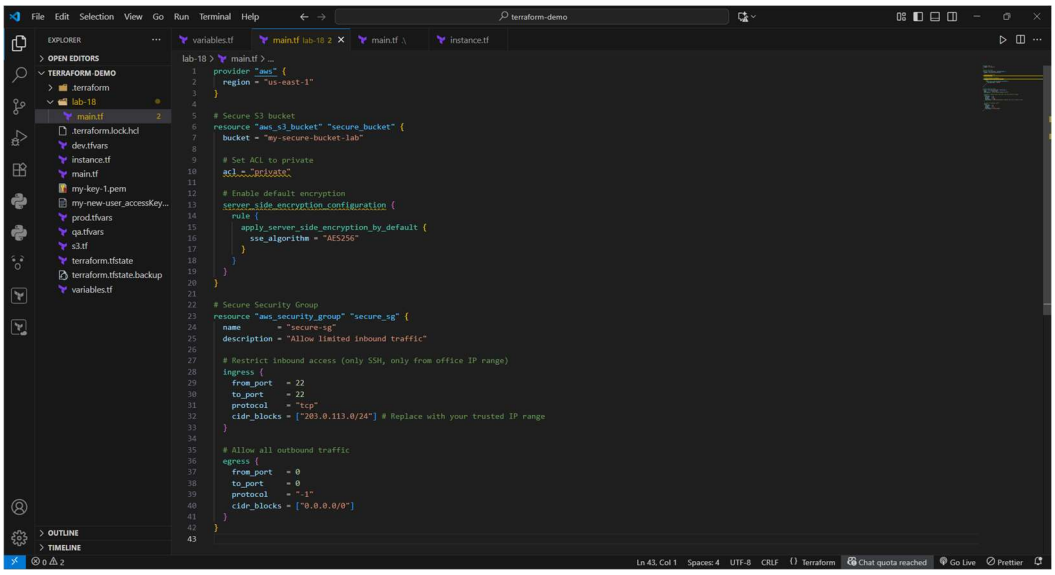


Step 5: Apply Fixes (Optional)

Modify the IaC template to:

- Set S3 bucket ACL to private
- Enable encryption (AES256)

- Restrict Security Group to specific IP ranges

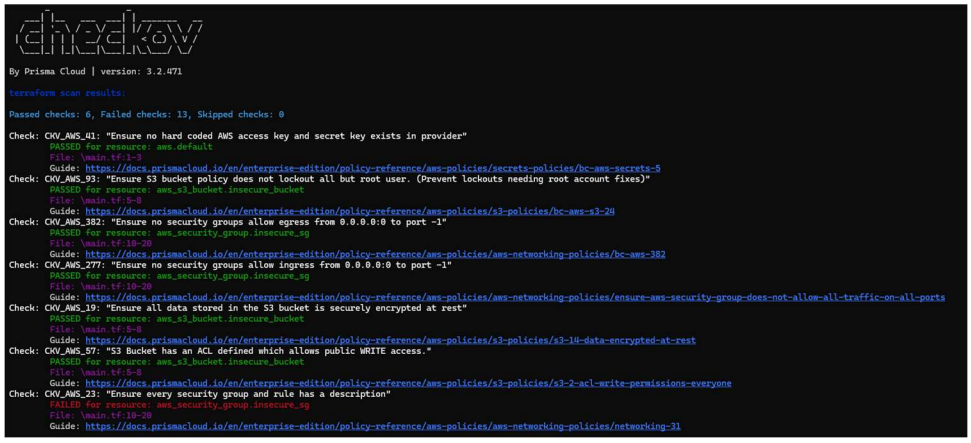


Step 6: Rescan the Template

Run the scan again:

```
checkov -d .
```

Now the findings should be **resolved or reduced**.



Step 7: Document Findings

Findings Log:

Fixed Vulnerabilities:

ID	Resource	Issue Detected	Risk	Fix Applied
1	aws_s3_bucket.secure_bucket	S3 bucket ACL was public-read	Public exposure of data	Changed ACL to **private**
2	aws_s3_bucket.secure_bucket	No encryption configured	Data at rest not protected	Enabled **AES256 server-side encryption**
3	aws_security_group.secure_sg	Ingress allowed 0.0.0.0/0 on all TCP ports	Full internet exposure (critical risk)	Restricted ingress to specific CIDR (203.0.113.0/24) and limited to **port 22 (SSH)**

Remaining Issues:

Check ID	Description	Status	Notes
CKV_AWS_23	**Missing SG rule descriptions**	❌ **Failed**	**Add descriptions to each rule**
CKV_AWS_382	**SG allows egress 0.0.0.0/0 on all ports**	❌ **Failed**	**Restrict egress to only required destinations**
CKV2_AWS_5	**SG not attached to a resource**	❌ **Failed**	**Attach SG to EC2 or relevant resource**
CKV_AWS_18	**S3 bucket logging not enabled**	❌ **Failed**	**Enable server access logging**
CKV_AWS_21	**S3 bucket versioning not enabled**	❌ **Failed**	**Enable versioning for recovery**
CKV2_AWS_6	**No public access block on S3 bucket**	❌ **Failed**	**Add aws_s3_bucket_public_access_block resource**

CKV2_AWS_61	**No lifecycle configuration on S3 bucket**	✗ **Failed**	**Add lifecycle rules for storage classes**
CKV_AWS_145	**S3 not using KMS encryption**	✗ **Failed**	**Switch from AES256 to KMS CMK**
CKV_AWS_144	**No cross-region replication configured**	✗ **Failed**	**Configure replication if needed**
CKV2_AWS_62	**No event notifications configured**	✗ **Failed**	**Add event notification configuration**