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)

```
mohdanas@Mohds-MacBook-Air ~ % python3 -m pip install checkov

Collecting checkov
  Downloading checkov-3.2.470-py3-none-any.whl.metadata (26 kB)
Collecting bc-python-hcl2==0.4.3 (from checkov)
  Downloading bc_python_hcl2-0.4.3-py3-none-any.whl.metadata (4.2 kB)
Collecting bc-detect-secrets==1.5.45 (from checkov)
  Downloading bc_detect_secrets-1.5.45-py3-none-any.whl.metadata (23 kB)
Collecting bc-jsonpath-ng==1.6.1 (from checkov)
```

- 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
```

```
Check: CKV_AWS_260: "Ensure no security groups allow ingress from 0.0.0.0:0 to port 80"
        FAILED for resource: aws_security_group.insecure_sg
        File: /main.tf:8-17

                8  | resource "aws_security_group" "insecure_sg" {
                9  |   name        = "insecure-sg"
                10 |   description = "Allow all inbound traffic"
                11 |   ingress {
                12 |     from_port   = 0
                13 |     to_port     = 65535
                14 |     protocol    = "tcp"
                15 |     cidr_blocks = ["0.0.0.0/0"]
                16 |   }
                17 | }
Check: CKV_AWS_24: "Ensure no security groups allow ingress from 0.0.0.0:0 to port 22"
        FAILED for resource: aws_security_group.insecure_sg
        File: /main.tf:8-17

                8  | resource "aws_security_group" "insecure_sg" {
                9  |   name        = "insecure-sg"
                10 |   description = "Allow all inbound traffic"
                11 |   ingress {
                12 |     from_port   = 0
                13 |     to_port     = 65535
                14 |     protocol    = "tcp"
                15 |     cidr_blocks = ["0.0.0.0/0"]
                16 |   }
                17 | }
Check: CKV_AWS_23: "Ensure every security group and rule has a description"
        FAILED for resource: aws_security_group.insecure_sg
        File: /main.tf:8-17

                8  | resource "aws_security_group" "insecure_sg" {
                9  |   name        = "insecure-sg"
                10 |   description = "Allow all inbound traffic"
                11 |   ingress {
                12 |     from_port   = 0
                13 |     to_port     = 65535
                14 |     protocol    = "tcp"
                15 |     cidr_blocks = ["0.0.0.0/0"]
                16 |   }
                17 | }
Check: CKV_AWS_25: "Ensure no security groups allow ingress from 0.0.0.0:0 to port 3389"
        FAILED for resource: aws_security_group.insecure_sg
        File: /main.tf:8-17

                8  | resource "aws_security_group" "insecure_sg" {
                9  |   name        = "insecure-sg"
                10 |   description = "Allow all inbound traffic"
                11 |   ingress {
                12 |     from_port   = 0
                13 |     to_port     = 65535
                14 |     protocol    = "tcp"
                15 |     cidr_blocks = ["0.0.0.0/0"]
                16 |   }
                17 | }
```

```
Check: CKV_AWS_20: "S3 Bucket has an ACL defined which allows public READ access."
        FAILED for resource: aws_s3_bucket.insecure_bucket
        File: /main.tf:4-7

                4 | resource "aws_s3_bucket" "insecure_bucket" {
                5 |   bucket = "my-insecure-bucket-lab"
                6 |   acl    = "public-read"
                7 | }

Check: CKV2_AWS_61: "Ensure that an S3 bucket has a lifecycle configuration"
        FAILED for resource: aws_s3_bucket.insecure_bucket
        File: /main.tf:4-7

                4 | resource "aws_s3_bucket" "insecure_bucket" {
                5 |   bucket = "my-insecure-bucket-lab"
                6 |   acl    = "public-read"
                7 | }

Check: CKV_AWS_18: "Ensure the S3 bucket has access logging enabled"
        FAILED for resource: aws_s3_bucket.insecure_bucket
        File: /main.tf:4-7

                4 | resource "aws_s3_bucket" "insecure_bucket" {
                5 |   bucket = "my-insecure-bucket-lab"
                6 |   acl    = "public-read"
                7 | }

Check: CKV_AWS_145: "Ensure that S3 buckets are encrypted with KMS by default"
        FAILED for resource: aws_s3_bucket.insecure_bucket
        File: /main.tf:4-7

                4 | resource "aws_s3_bucket" "insecure_bucket" {
                5 |   bucket = "my-insecure-bucket-lab"
                6 |   acl    = "public-read"
                7 | }

Check: CKV_AWS_144: "Ensure that S3 bucket has cross-region replication enabled"
        FAILED for resource: aws_s3_bucket.insecure_bucket
        File: /main.tf:4-7

                4 | resource "aws_s3_bucket" "insecure_bucket" {
                5 |   bucket = "my-insecure-bucket-lab"
                6 |   acl    = "public-read"
                7 | }

Check: CKV_AWS_21: "Ensure all data stored in the S3 bucket have versioning enabled"
        FAILED for resource: aws_s3_bucket.insecure_bucket
        File: /main.tf:4-7

                4 | resource "aws_s3_bucket" "insecure_bucket" {
                5 |   bucket = "my-insecure-bucket-lab"
                6 |   acl    = "public-read"
                7 | }

mohdanas@Mohds-MacBook-Air LAB_18 %
```

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**.

```
      _               _
   ___| |__   ___  ___| | _______   __
  / __| '_ \ / _ \/ __| |/ / _ \ \ / /
 | (__| | | |  __/ (__|   < (_) \ V /
  \___|_| |_|\___|\___|_|\_\___/ \_/

By Prisma Cloud | version: 3.2.470

terraform scan results:

Passed checks: 7, Failed checks: 12, Skipped checks: 0

Check: CKV_AWS_93: "Ensure S3 bucket policy does not lockout all but root user. (Prevent lockouts ne
eding root account fixes)"
        PASSED for resource: aws_s3_bucket.insecure_bucket
        File: /main.tf:4-15
Check: CKV_AWS_277: "Ensure no security groups allow ingress from 0.0.0.0:0 to port -1"
        PASSED for resource: aws_security_group.insecure_sg
        File: /main.tf:17-26
Check: CKV_AWS_382: "Ensure no security groups allow egress from 0.0.0.0:0 to port -1"
        PASSED for resource: aws_security_group.insecure_sg
        File: /main.tf:17-26
Check: CKV_AWS_41: "Ensure no hard coded AWS access key and secret key exists in provider"
        PASSED for resource: aws.default
        File: /main.tf:1-3
Check: CKV_AWS_20: "S3 Bucket has an ACL defined which allows public READ access."
        PASSED for resource: aws_s3_bucket.insecure_bucket
        File: /main.tf:4-15
Check: CKV_AWS_19: "Ensure all data stored in the S3 bucket is securely encrypted at rest"
        PASSED for resource: aws_s3_bucket.insecure_bucket
        File: /main.tf:4-15
Check: CKV_AWS_57: "S3 Bucket has an ACL defined which allows public WRITE access."
        PASSED for resource: aws_s3_bucket.insecure_bucket
        File: /main.tf:4-15
Check: CKV_AWS_260: "Ensure no security groups allow ingress from 0.0.0.0:0 to port 80"
        FAILED for resource: aws_security_group.insecure_sg
        File: /main.tf:17-26
```

Step 7: Document Findings

Create a simple findings log:

Findings Log

Check ID	Description	Status	Notes / Remediation
CKV_AWS_260	SG allows ingress from 0.0.0.0/0 to port 80	❌ Failed	Restrict SG to specific IP ranges
CKV_AWS_24	SG allows ingress from 0.0.0.0/0 to port 22	❌ Failed	Restrict SSH access to admin IP only
CKV_AWS_25	SG allows ingress from 0.0.0.0/0 to port 3389	❌ Failed	Block or limit RDP access
CKV_AWS_23	Missing SG rule descriptions	❌ Failed	Add descriptions to each rule
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	S3 public access block not configured	❌ Failed	Add `aws_s3_bucket_public_acce`
CKV2_AWS_61	S3 bucket lifecycle not configured	❌ Failed	Add lifecycle rules for storage classes
CKV_AWS_145	S3 not using KMS encryption	❌ Failed	Switch from AES256 to KMS
CKV_AWS_144	S3 cross-region replication not enabled	❌ Failed	Configure replication if needed
CKV2_AWS_62	S3 event notifications not enabled	❌ Failed	Add event notification configuration