sonar RULES

Products

Secrets
ABAP
Apex
C
C++
CloudFormation
COBOL
C#
CSS
Flex
Go
HTML
Java
JavaScript
Kotlin
Objective C
PHP
PL/I
PL/SQL
Python
RPG
Ruby
Scala
Swift
Terraform
Text
TypeScript
T-SQL
VB.NET
VB6
XML

Terraform static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your TERRAFORM code

All rules 50 Vulnerability 5 Security Hotspot 43 Code Smell 2

Tags Search by name...

Security Hotspot

Using unencrypted EFS file systems is security-sensitive

Security Hotspot

Using unencrypted SQS queues is security-sensitive

Security Hotspot

Using unencrypted SNS topics is security-sensitive

Security Hotspot

Using unencrypted SageMaker notebook instances is security-sensitive

Security Hotspot

Using unencrypted Elasticsearch domains is security-sensitive

Security Hotspot

Using unencrypted RDS databases is security-sensitive

Security Hotspot

Using unencrypted EBS volumes is security-sensitive

Security Hotspot

Disabling logging is security-sensitive

Security Hotspot

Administration services access should be restricted to specific IP addresses

Vulnerability

Unversioned Google Cloud Storage buckets are security-sensitive

Security Hotspot

Disabling S3 bucket MFA delete is security-sensitive

Enabling Azure resource-specific admin accounts is security-sensitive

Analyze your code

Security Hotspot Major azure

Enabling Azure resource-specific admin accounts can reduce an organization’s ability to protect itself against account or service account thefts.

Full Administrator permissions fail to correctly separate duties and create potentially critical attack vectors on the impacted resources.

In case of abuse of elevated permissions, both the data on which impacted resources operate and their access traceability are at risk.

Ask Yourself Whether

- This Azure resource is essential for the information system infrastructure.
- This Azure resource is essential for mission-critical functions.
- Compliance policies require this resource to disable its administrative accounts or permissions.

There is a risk if you answered yes to any of those questions.

Recommended Secure Coding Practices

Disable the administrative accounts or permissions in this Azure resource.

Sensitive Code Example

For Azure Batch Pools:

```
resource "azurerm_batch_pool" "example" {
  name = "sensitive"

  start_task {
    command_line = "echo 'Hello World'"
    max_task_retry_count = 1
    wait_for_success = true

    user_identity {
      auto_user {
        elevation_level = "Admin" # Sensitive
        scope = "Task"
      }
    }
  }
}
```

For Azure Container Registries:

Security Hotspot

Disabling versioning of S3 buckets is security-sensitive

Security Hotspot

Disabling server-side encryption of S3 buckets is security-sensitive

Security Hotspot

AWS tag keys should comply with a naming convention

Code Smell

Terraform parsing failure

Code Smell

```
resource "azurerm_container_registry" "example" {
  name = "example"
  admin_enabled = true # Sensitive
}
```

Compliant Solution

For Azure Batch Pools:

```
resource "azurerm_batch_pool" "example" {
  name = "example"

  start_task {
    command_line = "echo 'Hello World'"
    max_task_retry_count = 1
    wait_for_success = true

    user_identity {
      auto_user {
        elevation_level = "NonAdmin"
        scope = "Task"
      }
    }
  }
}
```

For Azure Container Registries:

```
resource "azurerm_container_registry" "exemple" {
  name = "example"
  admin_enabled = false
}
```

See

- OWASP Top 10 2021 Category A1 - Broken Access Control
- OWASP Top 10 2017 Category A5 - Broken Access Control
- MITRE, CWE-284 - Improper Access Control

Available In:

sonarcloud sonarqube