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

Azure role assignments that grant access to all resources of a subscription are security-sensitive

Analyze your code

Security Hotspot Major azure

Azure RBAC roles can be assigned to users, groups, or service principals. A role assignment grants permissions on a predefined set of resources called "scope".

The widest scopes a role can be assigned to are:

- Subscription: a role assigned with this scope grants access to all resources of this Subscription.
- Management Group: a scope assigned with this scope grants access to all resources of all the Subscriptions in this Management Group.

In case of security incidents involving a compromised identity (user, group, or service principal), limiting its role assignment to the narrowest scope possible helps separate duties and limits what resources are at risk.

Ask Yourself Whether

- The user, group, or service principal doesn’t use the entirety of the resources in the scope to operate on a day-to-day basis.
- It is possible to follow the Separation of Duties principle and split the scope into multiple role assignments with a narrower scope.

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

Recommended Secure Coding Practices

- Limit the scope of the role assignment to a Resource or Resource Group.
- Apply the least privilege principle by assigning roles granting as few permissions as possible.

Sensitive Code Example

```
resource "azurerm_role_assignment" "example" {
  scope                = data.azurerm_subscription.prim
  role_definition_name = "Reader"
  principal_id         = data.azuread_user.user.object_
}
```

Compliant Solution

```
resource "azurerm_role_assignment" "example" {
  scope                = azurerm_resource_group.example
  role_definition_name = "Reader"
  principal_id         = data.azuread_user.user.object_
}
```

See

- OWASP Top 10 2021 Category A1 - Broken Access Control
- OWASP Top 10 2021 Category A4 - Insecure Design
- OWASP Top 10 2017 Category A5 - Broken Access Control
- MITRE, CWE-266 - Incorrect Privilege Assignment
- Azure Documentation - Understand scope for Azure RBAC
- Azure Documentation - Best practices for Azure RBAC

Available In:

sonarcloud sonarqube

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