Terraform static code analysis

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

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

Disabling Managed Identities for Azure resources is security-sensitive

Analyze your code

Security Hotspot   Major   azure

Disabling Managed Identities can reduce an organization's ability to protect itself against configuration faults and credentials leaks.

Authenticating via managed identities to an Azure resource solely relies on an API call with a non-secret token. The process is inner to Azure: secrets used by Azure are not even accessible to end-users.

In typical scenarios without managed identities, the use of credentials can lead to mistakenly leaving them in code bases. In addition, configuration faults may also happen when storing these values or assigning them permissions.

By transparently taking care of the Azure Active Directory authentication, Managed Identities allow getting rid of day-to-day credentials management.

Ask Yourself Whether

The resource:

- Needs to authenticate to Azure resources that support Azure Active Directory (AAD).
- Uses a different Access Control system that doesn't guarantee the same security controls as AAD, or no Access Control system at all.

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

Recommended Secure Coding Practices

Enable the Managed Identities capabilities of this Azure resource. If supported, use a System-Assigned managed identity, as:

- It cannot be shared across resources.
- Its life cycle is deeply tied to the life cycle of its Azure resource.
- It provides a unique independent identity.

Alternatively, User-Assigned Managed Identities can also be used but don't guarantee the properties listed above.

Sensitive Code Example

For Typical identity blocks:

```
resource "azurerm_api_management" "example" { # Sensiti
  name           = "example"
  publisher_name = "company"
}
```

For connections between Kusto Clusters and Azure Data Factory:

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_data_factory_linked_service_kusto" "e
  name                 = "example"
  use_managed_identity = false # Sensitive
}
```

Compliant Solution

For Typical identity blocks:

```
resource "azurerm_api_management" "example" {
  name           = "example"
  publisher_name = "company"

  identity {
    type = "SystemAssigned"
  }
}
```

For connections between Kusto Clusters and Azure Data Factory:

```
resource "azurerm_data_factory_linked_service_kusto" "e
  name                 = "example"
  use_managed_identity = true
}
```

See

- OWASP Top 10 2021 Category A05 - Security Misconfiguration
- OWASP Top 10 2017 Category A06 - Security Misconfiguration
- Azure AD Documentation - Managed Identities Overview
- Azure AD Documentation - Managed Identities Best Practices
- Azure AD Documentation - Services that support managed identities

Available In:

sonarcloud sonarqube