




 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 clear-text protocols is security-sensitive

 Security Hotspot


 Vulnerability


Google Cloud load balancers SSL policies should not offer weak cipher suites

 Vulnerability


 Security Hotspot


Azure custom roles should not grant subscription Owner capabilities

 Vulnerability


 Security Hotspot


Excluding users or groups activities from audit logs is security-sensitive

 Security Hotspot


 Security Hotspot

Defining a short log retention duration is security-sensitive

 Security Hotspot

 Security Hotspot

Enabling Attribute-Based Access Control for Kubernetes is security-sensitive

 Security Hotspot

 Security Hotspot

Creating custom roles allowing privilege escalation is security-sensitive

 Security Hotspot

 Security Hotspot

Creating App Engine handlers without requiring TLS is security-sensitive

 Security Hotspot

 Security Hotspot

Excessive granting of GCP IAM permissions is security-sensitive

 Security Hotspot




 Security Hotspot

Enabling project-wide SSH keys to access VM instances is security-sensitive

 Security Hotspot

### Granting access to S3 buckets to all or authenticated users is security-sensitive

Analyze your code

 Security Hotspot  Blocker  aws cwe owasp

Predefined permissions, also known as **canned ACLs**, are an easy way to grant large privileges to predefined groups or users.

The following canned ACLs are security-sensitive:

- PublicRead, PublicReadWrite grant respectively "read" and "read and write" privileges to everyone in the world (AllUsers group).
- AuthenticatedRead grants "read" privilege to all authenticated users (AuthenticatedUsers group).

#### Ask Yourself Whether

- The S3 bucket stores sensitive data.
- The S3 bucket is not used to store static resources of websites (images, CSS ...).

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

#### Recommended Secure Coding Practices

It's recommended to implement the least privilege policy, ie to grant necessary permissions only to users for their required tasks. In the context of canned ACL, set it to **private** (the default one) and if needed more granularity then use an appropriate S3 policy.

#### Sensitive Code Example

All users (ie: anyone in the world authenticated or not) have read and write permissions with the **public-read-write** access control:





```
resource "aws_s3_bucket" "mynoncompliantbucket" { # Sensitive
  bucket = "mynoncompliantbucketname"
  acl     = "public-read-write"
}
```

#### Compliant Solution

With the **private** access control (default), only the bucket owner has the read/write permissions on the buckets and its ACL.

```
resource "aws_s3_bucket" "mycompliantbucket" { # Compliant
  bucket = "mycompliantbucketname"
  acl     = "private"
}
```

See

<div>Granting public access to GCP resources is security-sensitive</div> <div> Security Hotspot</div>
<div>Creating GCP SQL instances without requiring TLS is security-sensitive</div> <div> Security Hotspot</div>
<div>Creating DNS zones without DNSSEC enabled is security-sensitive</div> <div> Security Hotspot</div>
<div>Creating keys without a rotation period is security-sensitive</div> <div> Security Hotspot</div>
<div>Granting highly privileged GCP resource rights is security-sensitive</div>

- [OWASP Top 10 2021 Category A1](#) - Broken Access Control
- [AWS Documentation](#) - Access control list (ACL) overview (canned ACLs)
- [AWS Documentation](#) - Controlling access to a bucket with user policies
- [MITRE, CWE-732](#) - Incorrect Permission Assignment for Critical Resource
- [MITRE, CWE-284](#) - Improper Access Control
- [OWASP Top 10 2017 Category A5](#) - Broken Access Control

Available In:



© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved.  
[Privacy Policy](#)