

































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



Secrets static code analysis

Unique rules to find Vulnerabilities in your source code and language agnostic config files

All rules 7 Vulnerability 7

Tags ▾

Search by name... 🔍

- Amazon Web Services credentials should not be disclosed
Vulnerability
- Amazon MWS credentials should not be disclosed
Vulnerability
- Google API keys should not be disclosed
Vulnerability
- Google Cloud service accounts keys should not be disclosed
Vulnerability
- Alibaba Cloud AccessKeys should not be disclosed
Vulnerability
- IBM API keys should not be disclosed
Vulnerability
- Azure Storage Account Keys should not be disclosed
Vulnerability

Amazon Web Services credentials should not be disclosed

Analyze your code

Vulnerability Blocker cwe sans-top25-porous owasp-a3

AWS credentials are designed to authenticate and authorize requests to AWS.

If your application interacts with AWS then it requires AWS credentials to access all the resources it needs to function properly. Resources that can be accessed depend on the permission granted to the AWS account. Worst case scenario: the credential authenticates to the AWS account root user who has unrestricted access to all resources in your AWS account, including billing information.

Therefore only administrators should have access to the AWS credentials used by your application.

As a consequence, AWS credentials should not be stored along with the application code as they would grant special privilege to anyone who has access to the application source code.

Credentials should be stored outside of the code in a file that is never committed to your application code repository.

If possible, a better alternative is to use your cloud provider’s service for managing secrets. On AWS this service is called [Secret Manager](#).

When credentials are disclosed in the application code, consider them as compromised and revoke them immediately.

This rule flags instances of:

- AWS Secret Access Key
- AWS Access ID
- AWS Session Token

See

- [OWASP Top 10 2017 Category A2](#) - Broken Authentication
- [MITRE, CWE-798](#) - Use of Hard-coded Credentials
- [MITRE, CWE-259](#) - Use of Hard-coded Password
- [CERT, MSC03-J.](#) - Never hard code sensitive information
- [SANS Top 25](#) - Porous Defenses

Available In:

sonarlint