- 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

IBM API keys should not be disclosed

Analyze your code

Vulnerability | Blocker | cwe sans-top25-porous owasp-a3

IBM API keys are used to authenticate applications that consume IBM Cloud APIs.

If your application interacts with IBM then it requires credentials to access all the resources it needs to function properly. Resources that canbe accessed depend on the permissions granted to the account. These credentials may authenticate a user who has unrestricted access to all resourcesin your account, including billing information.

Recommended Secure Coding Practices

Only administrators should have access to the IBM API keys used by your application.

As a consequence, IBM API keys should not be stored along with the application code as they could be disclosed to a large audience or could be madepublic.

IBM API keys 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 IBM Cloud this service is called Secrets Manager.

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

In addition to secure storage, it's important to apply restrictions to API keys in order to mitigate the impacts whenthey are discovered by malicious actors.

See

- OWASP Top 10 2017 Category A3 - Sensitive Data Exposure
- 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