AWS Best Practice

**TABLE OF CONTENTS**

[Best Practices 1](#_Toc1720439405)

[Naming 1](#_Toc563187559)

[General conventions 1](#_Toc1140858885)

[Service Users 1](#_Toc218798029)

[Resource and data source arguments 1](#_Toc896516024)

[Variables 2](#_Toc8454459)

[Outputs 2](#_Toc852868051)

[Resources: 2](#_Toc9478135)

# Best Practice

## User IAM

* NEVER use root user for day to day operators
* create custom users using "least privilege principle" for IDE and cli purposes  
  <https://en.wikipedia.org/wiki/Principle_of_least_privilege>
* create custom apikeys using "least privilege principle" for IDE and cli purposes  
  <https://en.wikipedia.org/wiki/Principle_of_least_privilege>
* use tags to limit and bound roles and policies
* Use the policy simulator to test policy before putting them into effect  
  [https://policysim.aws.amazon.com/home/index.jsp?#](https://policysim.aws.amazon.com/home/index.jsp)
* **MUST** have MFA enabled for all users

## Naming

* RiverSafe wide Roles SHOULD be prefixed "RS\_"
* RiverSafe wide Policies SHOULD be prefixed "RS\_"
* RiverSafe wide Usergroups SHOULD be prefixed "RS\_"
* RiverSafe users SHOULD be named after your riversafe user id aka for Anthony Mckale "mckalea"

## Tagging

* TBD

## Billing

* <https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/budgets-best-practices.html>

# Further Reading

## Snyk Serverless Security top 10

* list of 10 Serverless security best practices:  
  <https://snyk.io/blog/10-serverless-security-best-practices/>   
  <https://res.cloudinary.com/snyk/raw/upload/v1559295122/Cheat_Sheet-_10_Serverless_Security_Best_Practices.pdf>
* Patch function dependencies
* Adopt the principle of least privilege
* Maintain isolated function perimeters
* Sanitize event input to avoid injection
* Employ API gateways as a security buffer
* Monitor and log functions
* Follow secure coding conventions for application code
* Secure and verify data in transit
* Manage secrets in secure storage
* Deploy functions in minimal granularity

## Snyk Serverless Secrets Management top 10

* list of 10 Serverless security best practices:  
  <https://snyk.io/blog/best-practices-for-secrets-management-in-serverless-applications/>   
  <https://res.cloudinary.com/snyk/raw/upload/v1559295122/Cheat_Sheet-_10_Serverless_Security_Best_Practices.pdf>
* Rotate keys and credentials regularly
* secure storage solution for your secrets and sensitive credentials
* Do not store secrets embedded in dockerimages / or compiled source packages