# Terraform cơ sở hạ tầng SAP của bạn trên AWS

## **Giới thiệu**

# Hệ thống SAP rất quan trọng đối với hoạt động kinh doanh, điều quan trọng là phải tuân thủ các thông lệ tốt nhất của SAP khi lập kế hoạch triển khai và vận hành. Trong khi khách hàng ngày càng lựa chọn RISE trên AWS, nhiều khách hàng đang chạy SAP trên AWS theo cách gốc, do đó, khi yêu cầu về khối lượng công việc thay đổi, các nhóm phải lặp lại và thích ứng nhanh chóng để cung cấp cơ sở hạ tầng cần thiết, đồng thời lưu ý đến tất cả các nguyên tắc bảo mật, thông lệ tốt nhất về hiệu suất và các yêu cầu không chức năng khác.

# Trong bài đăng trên blog này, chúng tôi sẽ giúp bạn tạo cơ sở hạ tầng SAP của riêng mình trên AWS bằng [Terraform](https://www.terraform.io/) và các khối xây dựng được cung cấp trong [kho lưu trữ GitHub này](https://github.com/aws-samples/aws-sap-terraform) .

# Các tài nguyên tối thiểu cần thiết để tạo ra một kiến ​​trúc như trong Hình 1 và lưu trữ một hệ thống SAP duy nhất thường bao gồm:

# [Đám mây riêng ảo](https://aws.amazon.com/vpc/) (VPC) và [mạng con](https://docs.aws.amazon.com/vpc/latest/userguide/configure-subnets.html)

# [Cấu hình nhóm bảo mật](https://docs.aws.amazon.com/vpc/latest/userguide/vpc-security-groups.html) cụ thể cho ứng dụng và cơ sở dữ liệu SAP

# [Các trường hợp EC2](https://docs.aws.amazon.com/sap/latest/general/ec2-instance-types-sap.html)

# [Khối lượng EBS](https://docs.aws.amazon.com/ebs/latest/userguide/ebs-volumes.html) cho [SAP HANA](https://docs.aws.amazon.com/sap/latest/sap-hana/hana-ops-storage-config.html)

# [Hệ thống tệp đàn hồi Amazon](https://aws.amazon.com/efs/) (EFS) cho các thư mục sapmnt và vận chuyển

# [Khóa KMS](https://aws.amazon.com/kms/?gclid=CjwKCAiA34S7BhAtEiwACZzv4dPiv_KR32pPWuDOtaPajoBqUGa_i1xuN92T39hhqDcAoyTgakV_EBoCWh8QAvD_BwE&trk=ced3f61b-256e-4936-a949-66a0aee40a02&sc_channel=ps&ef_id=CjwKCAiA34S7BhAtEiwACZzv4dPiv_KR32pPWuDOtaPajoBqUGa_i1xuN92T39hhqDcAoyTgakV_EBoCWh8QAvD_BwE:G:s&s_kwcid=AL!4422!3!629393326477!!!g!!!16080176321!133788127918) để mã hóa khi nghỉ ngơi

# [Thùng S3](https://aws.amazon.com/s3/) cho các tệp cài đặt, bản sao lưu SAP và lưu trữ tệp

# [Vai trò IAM](https://aws.amazon.com/iam/) cho các trường hợp, quản trị viên ứng dụng, kiểm toán viên và những người khác

# Kiến trúc SAP tối thiểu

# *Hình 1 – Kiến trúc SAP S/4HANA mẫu trên AWS*

# Tiêu chuẩn công nghiệp và các thông lệ tốt nhất để triển khai tài nguyên vào đám mây khuyến nghị sử dụng Cơ sở hạ tầng dưới dạng mã. Một số lợi ích mạnh mẽ hơn khi sử dụng Cơ sở hạ tầng dưới dạng mã là a) Cải thiện sự cộng tác và cấu hình được chia sẻ, b) dễ dàng phát triển và quản lý phiên bản cơ sở hạ tầng của bạn và c) tự động hóa cung cấp.

## **Cơ sở hạ tầng như mã**

# Cơ sở hạ tầng dưới dạng mã (IaC) là một hoạt động trong đó cơ sở hạ tầng được cung cấp và quản lý bằng các kỹ thuật phát triển mã và phần mềm, chẳng hạn như kiểm soát phiên bản và tích hợp liên tục. Đám mây AWS cung cấp một mô hình do API điều khiển cho phép các nhà phát triển và quản trị viên hệ thống tương tác với cơ sở hạ tầng theo chương trình, ở quy mô lớn. Khi các tài nguyên và dịch vụ được xác định là mã, cơ sở hạ tầng và máy chủ có thể nhanh chóng được triển khai bằng các mẫu chuẩn hóa, được cập nhật bằng các bản vá/phiên bản mới nhất và được sao chép theo những cách có thể lặp lại.

# Điều này cho phép các công ty triển khai nhanh chóng cơ sở hạ tầng SAP trên AWS bằng cách sử dụng các công cụ như [AWS Launch Wizard](https://aws.amazon.com/launchwizard/) . Với AWS Launch Wizard, các nhóm khách hàng có thể xây dựng các hệ thống SAP phù hợp với các biện pháp thực hành tốt nhất của AWS một cách nhanh chóng từ AWS Console theo trải nghiệm có hướng dẫn được thiết kế cho quản trị viên SAP. Khi sử dụng AWS Launch Wizard, các dịch vụ AWS khác được sử dụng ở phía sau, chẳng hạn như AWS CloudFormation và AWS Systems Manager, để thực hiện công việc cung cấp và cấu hình. Mặc dù AWS Launch Wizard cung cấp một cách dễ sử dụng và hiệu quả để cung cấp cơ sở hạ tầng liên quan đến SAP trên AWS, nhưng trong một số trường hợp, bạn có thể thích sử dụng công cụ IaC không phụ thuộc vào đám mây. Trong những tình huống như thế này, Terraform của HashiCorp có thể được tận dụng để tự động hóa các triển khai SAP này trên AWS.

### **DevOps cho cơ sở hạ tầng SAP**

# Việc định nghĩa cơ sở hạ tầng của bạn dưới dạng mã mở ra một góc nhìn mới về lợi ích. Nó cho phép bạn triển khai mô hình DevOps trong thế giới SAP Basis.

# Đường ống cấp cao

# *Hình 2 – Tổng quan về quy trình DevOps*

# Việc tự động cung cấp SAP ở mọi lớp mang lại những lợi ích sau:

# Giao hàng nhanh chóng

# Tỉ lệ

# Độ tin cậy

# Cải thiện sự cộng tác

# Bảo vệ

# Bạn có thể tìm hiểu sâu hơn và tìm thêm thông tin về [DevOps là gì](https://aws.amazon.com/devops/what-is-devops/#:~:text=DevOps%20is%20the%20combination%20of,development%20and%20infrastructure%20management%20processes.) ?.

# Là một phần trong hoạt động AWS Professional Services SAP của chúng tôi, chúng tôi giúp khách hàng xây dựng nhà máy tự động hóa của họ bằng nhiều công cụ DevOps như Terraform và CloudFormation, kết hợp [SAP và AWS Best Practices](https://docs.aws.amazon.com/sap/latest/general/welcome.html) vào loại tự động hóa này. Trong blog này, chúng tôi cung cấp một giải pháp Terraform mẫu để triển khai kiến ​​trúc được mô tả trong Hình 1.

## **Bắt đầu**

# Fork kho lưu trữ GitHub [aws-sap-terraform](https://github.com/aws-samples/aws-sap-terraform) vào tài khoản của bạn. [Làm thế nào để fork?](https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/working-with-forks/fork-a-repo)

# Điều hướng qua các thư mục theo thứ tự cụ thể: KMS, EFS, Security Groups, IAM, Instances và điền vào các tệp có tên “dev.tfvars” trên mỗi thư mục theo yêu cầu của môi trường và các ví dụ bên dưới. Để biết thêm thông tin về cách điền vào các tệp này bên dưới trong phần Cấu hình cơ sở hạ tầng của bạn, cũng như từng trang README trong kho lưu trữ.

# QUAN TRỌNG: Tên tệp cũng như tên nhánh phải khớp với môi trường bạn đang triển khai. Nếu bạn đang triển khai dev, hãy giữ dev.tfvars và sử dụng nhánh dev. Nếu bạn đang triển khai qa, hãy tạo một tệp mới có tên qa.tfvars và tạo một nhánh qa. Nếu bạn đang triển khai prod, hãy tạo tệp prod.tfvars và tạo/sử dụng một nhánh có tên prod. Chiến lược nhánh được đề xuất cho các triển khai cơ sở hạ tầng SAP là nhánh môi trường. Hãy xem [Chiến lược nhánh môi trường là gì](https://www.wearefine.com/news/insights/env-branching-with-git/) nếu bạn đang tìm kiếm thêm thông tin chi tiết, cũng như [Cách làm việc với các nhánh môi trường](https://docs.dataops.live/docs/environment-management/branching-strategies/#overview-git-branches) .

# Tại mỗi thư mục, hãy xem và cập nhật theo yêu cầu tại tệp locals.tf (ví dụ [kms/locals.tf](https://github.com/aws-samples/aws-sap-terraform/blob/main/kms/locals.tf) ). Các tệp này chứa tất cả các thẻ được đính kèm vào tài nguyên của cấu hình đó. Cập nhật chúng theo yêu cầu.

# Đẩy mã đã cập nhật vào kho lưu trữ của bạn.

# Thêm khóa thẻ “sap\_relevant” và giá trị “true” vào VPC mà bạn đang sử dụng cho lần triển khai này.

# Thêm khóa thẻ “sap\_relevant” và giá trị “true” vào các Mạng con bạn đang sử dụng cho lần triển khai này. Điều này sẽ ảnh hưởng đến nơi tài nguyên sẽ được tạo. Đảm bảo bạn áp dụng thẻ này cho tất cả các mạng con sẽ nhận được ứng dụng SAP.

# Tiếp tục đến phần bên dưới “Cách triển khai”

## **Cấu hình cơ sở hạ tầng của bạn**

### **1. Cấu hình KMS**

# Cấu hình các khóa KMS bạn muốn tạo cho môi trường của mình theo mẫu bên dưới bằng cách sử dụng tệp kms/dev.tfvars. Bạn có thể tìm thêm thông tin về cấu hình này trong [README](https://github.com/aws-samples/aws-sap-terraform/blob/main/kms/README.md) .

# environment = "dev"

# aws\_region = "us-east-1"

# 

# keys\_to\_create = {

# "ebs" = {

# alias\_name = "kms-alias-ebs"

# }

# "efs" = {}

# "cloudwatch" = {}

# "s3" = {}

# }

# 

# HCL

### **2. Cấu hình EFS**

# Cấu hình các ổ đĩa EFS cần thiết cho môi trường của bạn theo mẫu bên dưới bằng cách sử dụng tệp efs/dev.tfvars. Bạn có thể tìm thêm thông tin về cấu hình này trong [README](https://github.com/aws-samples/aws-sap-terraform/blob/main/efs/README.md) .

# environment = "dev"

# aws\_region = "us-east-1"

# sap\_discovery\_tag = "sap\_relevant"

# 

# efs\_to\_create = {

# "D01-sapmnt" = {

# access\_point\_info = {

# posix\_user = {

# gid = 5001,

# uid = 3001

# },

# root\_directory = {

# creation\_info = {

# owner\_gid = 5001,

# owner\_uid = 3001,

# permissions = 0775

# },

# path : "/",

# }

# }

# }

# }

# HCL

### **3. Cấu hình nhóm bảo mật**

# Cấu hình Nhóm bảo mật cần thiết cho triển khai của bạn theo mẫu bên dưới bằng cách sử dụng tệp security\_group/dev.tfvars. Bạn có thể tìm thêm thông tin về cấu hình này trong [README](https://github.com/aws-samples/aws-sap-terraform/blob/main/security_group/README.md) .

# environment = "dev"

# aws\_region = "us-east-1"

# sap\_discovery\_tag = "sap\_relevant"

# 

# app\_sg\_list = {

# app1 = {

# description = "App SG"

# efs\_to\_allow = ["sapmedia", "sapmnt", "saptrans"]

# rules = {

# "app1" = {

# source = "vpc"

# ports = [4237]

# }

# "app2" = {

# source = "vpc"

# ports = [8443]

# }

# "app3" = {

# source = "self"

# protocol = "all"

# ports = [0, 0]

# }

# "app4" = {

# source = "self"

# ports = [1, 65535]

# }

# }

# }

# }

# HCL

### **4. Cấu hình Chính sách và Vai trò IAM**

# Trước tiên, hãy cấu hình các chính sách cần thiết cho môi trường của bạn bằng cách sử dụng mẫu bên dưới bằng tệp iam/dev.tfvars. Bạn có thể tìm thêm thông tin về cách cấu hình này trong [README](https://github.com/aws-samples/aws-sap-terraform/blob/main/iam/README.md) .

# iam\_policies = {

# ec2\_permissions = {

# name = "iam-policy-sap-ec2-others"

# statements = {

# stmt1 = {

# effect = "Allow"

# actions = [

# "s3:GetBucketPolicyStatus",

# "s3:GetBucketLocation",

# "s3:ListBucket",

# "s3:GetBucketAcl",

# "s3:GetBucketPolicy",

# "s3:PutObjectTagging",

# "s3:PutObject",

# "s3:GetObject",

# "s3:HeadObject",

# "s3:DeleteObject",

# ]

# resources = ["arn:aws:s3:::sap-media-bucket"]

# }

# }

# }

# }

# HCL

# Cấu hình các vai trò cần thiết cho môi trường của bạn bằng cách sử dụng mẫu bên dưới. Lưu ý rằng các chính sách được liệt kê trong "policies" bên dưới "name" trong ví dụ bên dưới phải sử dụng cùng tên chính xác mà bạn đã đặt cho chính sách ở trên. Theo cách này, tự động hóa sẽ đính kèm các chính sách bạn vừa tạo vào vai trò của mình.

# environment = "dev"

# aws\_region = "us-east-1"

# 

# iam\_roles = {

# role1 = {

# name = "iam-role-sap-ec2"

# policies = [

# "iam-policy-sap-data-provider",

# "iam-policy-sap-efs",

# "iam-policy-sap-ec2-others"

# ]

# managed\_policies = [

# "AmazonSSMManagedInstanceCore"

# ]

# permissions\_boundary\_arn = "arn:aws:iam::<<account nr>>:policy/example-permissions-boundary-rds"

# },

# role2 = {

# name = "iam-role-sap-ec2-ha"

# policies = [

# "iam-policy-sap-data-provider",

# "iam-policy-sap-efs",

# "iam-policy-sap-ec2-others",

# "iam-policy-sap-pacemaker-stonith",

# "iam-policy-sap-pacemaker-overlayip"

# ]

# managed\_policies = [

# "AmazonSSMManagedInstanceCore"

# ]

# permissions\_boundary\_arn = ""

# }

# }

# 

# HCL

### **5. Cấu hình các phiên bản EC2**

# Cuối cùng, hãy cấu hình các phiên bản EC2 cần thiết cho môi trường của bạn bằng cách sử dụng mẫu bên dưới bằng tệp ec2\_instance/dev.tfvars. Bạn có nhiều lựa chọn khác nhau để khai báo các phiên bản của mình, một lựa chọn động hơn và một lựa chọn chi tiết hơn cho các phiên bản tùy chỉnh. Bạn có thể tìm thêm thông tin về cách cấu hình này trong [README](https://github.com/aws-samples/aws-sap-terraform/blob/main/ec2_instance/README.md) .

# environment = "dev"

# aws\_region = "us-east-1"

# 

# instances\_to\_create = {

# sapd01db1 = {

# "private\_ip" = "10.237.40.144"

# "domain" = "mylab.com"

# "application\_code" = "hana"

# "application\_SID" = "D01"

# "ha" = false

# "ami\_ID" = "ami-12345678901234567"

# "subnet\_ID" = "subnet-12345678901234567"

# "key\_name" = "mycmk"

# "monitoring" = true

# "root\_ebs\_size" = 80

# "ec2\_tags" = {

# "tag\_key\_1" = "tag\_value\_1"

# }

# "instance\_type" = "x2iedn.xlarge"

# "hana\_data\_storage\_type" = "gp3"

# "hana\_logs\_storage\_type" = "gp3"

# "hana\_backup\_storage\_type" = "st1"

# "hana\_shared\_storage\_type" = "gp3"

# }

# }

# HCL

## **Cách triển khai**

# QUAN TRỌNG: Mã mẫu cung cấp một lộ trình dễ dàng giả sử bạn đang sử dụng GitLab để triển khai cơ sở hạ tầng của mình và nó đã được kết nối với tài khoản AWS của bạn. Nếu bạn không sử dụng điều đó, bạn sẽ phải viết lại/thay thế các tệp “.gitlab-ci.yml”, tất cả “<folder>/gitlab/pipeline.yaml” và đảm bảo bạn có cấu hình phù hợp trong “<folder>/configs.tf” theo yêu cầu của công cụ CI/CD của bạn.

# Sau khi bạn đã cập nhật tất cả các tệp của mình từ các bước trên cũng như làm lại các thông số kỹ thuật CI/CD, hãy đẩy mã lên kho lưu trữ của bạn. Thực hiện theo thứ tự bên dưới để triển khai tài nguyên của bạn trên tài khoản AWS. Trình tự tạo:

| Sự liên tiếp | Sân khấu | Việc làm |
| --- | --- | --- |
| 1 | KMS | kms\_kế hoạch, kms\_áp dụng |
| 2 | Hệ thống quản lý an toàn | efs\_plan, efs\_apply |
| 3 | Nhóm bảo mật | kế hoạch nhóm bảo mật, áp dụng nhóm bảo mật |
| 4 | TÔI LÀ | Tôi\_kế\_hoạch, Tôi\_áp\_dụng |
| 5 | Các trường hợp EC2 | kế hoạch\_trường hợp, áp dụng\_trường hợp |

## **Làm thế nào để tự động hóa môi trường SAP của bạn hơn nữa**

# Để tối đa hóa hiệu quả, cải thiện sự xuất sắc trong hoạt động và tận dụng các dịch vụ AWS khác cho hoạt động SAP, hãy xem cách cấu hình:

# CloudWatch giám sát số liệu, cảnh báo và sự kiện cho SAP [HANA](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/appinsights-tutorial-sap-hana.html) , [NetWeaver](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/appinsights-tutorial-sap-netweaver.html) và ['](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/appinsights-tutorial-sap-ase.html)

# Công cụ sao lưu (ví dụ: AWS Backup và AWS [Backint Agent](https://aws.amazon.com/backint-agent/) )

# [EC2 Image Builder để tạo AMI cơ bản](https://aws.amazon.com/blogs/awsforsap/build-sap-golden-amis-with-ec2-image-builder-and-ansible/) cho các cài đặt SAP

# Sổ tay hướng dẫn quản lý hệ thống cho hoạt động tự động hóa

# [EC2 Start/Stop](https://aws.amazon.com/blogs/awsforsap/automate-start-or-stop-of-distributed-sap-hana-systems-using-aws-systems-manager/) và [EC2 Start/Stop 2](https://aws.amazon.com/blogs/awsforsap/start-stop-sap-systems-with-slack-using-aws-chatbot/)

# [Quản lý bản vá](https://docs.aws.amazon.com/sap/latest/sap-netweaver/net-win-operating-system-maintenance.html#net-win-patching)

# [Quy tắc AWS Config](https://aws.amazon.com/config/) để giám sát tuân thủ

# Các ứng dụng không có máy chủ để mở rộng khả năng của SAP

# [Tự động hóa các thử nghiệm HA (Khả năng sẵn sàng cao)](https://aws.amazon.com/blogs/awsforsap/automate-high-availability-tests-for-sap-hana/)

## **Dọn dẹp nhà cửa**

# Trong trường hợp bạn chỉ khám phá chức năng được mô tả trong blog này để làm quen hơn với các khả năng của Terraform IaC, chạy POC ngắn gọn hoặc chạy triển khai tạm thời vì bất kỳ lý do nào khác, hãy nhớ dọn dẹp các tài nguyên đã triển khai để tránh chi tiêu không cần thiết trong tài khoản AWS của bạn khi bạn không còn cần đến các tài nguyên đó nữa.

# Bạn có thể xóa thủ công các tài nguyên trong AWS Management Console hoặc, thay vào đó, bao gồm một bước trong quy trình của bạn để thực hiện việc đó (khuyến nghị). Trong mã mẫu, như một phần của cấu hình quy trình GitLab mẫu, chúng tôi đã bao gồm một giai đoạn Destroy. Sau khi thực hiện, nó sẽ sử dụng Terraform để xóa tất cả các tài nguyên đã tạo trước đó.

## **Phần kết luận**

# Như chúng tôi đã chứng minh với các mô-đun Terraform của mình tại [GitHub](https://github.com/aws-samples/aws-sap-terraform/tree/main) , bằng cách định nghĩa SAP là mã, cơ sở hạ tầng và máy chủ có thể được triển khai nhanh chóng bằng các mẫu chuẩn hóa, cập nhật các bản vá/phiên bản mới nhất và sao chép theo những cách có thể lặp lại để vượt qua một số rào cản ban đầu trong quá trình học.

# Terraform cùng với mô hình điều khiển API đám mây AWS cho phép các nhà phát triển và quản trị viên hệ thống tương tác với cơ sở hạ tầng theo chương trình, ở quy mô lớn, thay vì phải thiết lập và cấu hình tài nguyên theo cách thủ công.

# Giải pháp này giúp bạn dễ dàng thêm vào cấu hình Terraform hiện tại hoặc nếu bạn mới bắt đầu sử dụng Terraform, bạn có thể chỉ cần sao chép kho lưu trữ và bắt đầu với cấu hình Terraform mới.

# Bạn có muốn tìm hiểu thêm hay muốn hiểu rõ hơn về cách mở rộng giải pháp này cho dự án của mình không?

# Để biết thêm thông tin, hãy liên hệ với chúng tôi qua email sap-on-aws@amazon.com .

# 