Breaking New Ground with OpenTofu Exclusive Features

Ronny Orot

OpenTofu

What is OpenTofu?

Open Source

IaC tool driven by the community.

Fork of Terraform

Supports configurations, providers and modules.

Values

- Openness
- Transparency
- Collaboration

OpenTofu

What is OpenTofu?

August 2023	September 2023	January 2024	March 2024	January 2025	...
Manifesto goes live	THE LINUX FOUNDATION	OpenTofu 1.6 GA	OpenTofu Day	OpenTofu 1.9	

OpenTofu

Community Support

Over 23.9K stars on GitHub and 150 contributors 🌟

OpenTofu support comes to JFrog Artifactory

JFrog

▲ Oracle dumps Terraform for OpenTofu (thestack.technology)
255 points by p1nkpineapple 8 months ago | hide | past | favorite | 183 comments

Add OpenTofu CI/CD components and deprecate Terraform CI/CD templates
Open Epic created 3 months ago by Timo Furrer

GitLab

The OpenTofu Manifesto

Terraform was open-sourced in 2014 under the Mozilla Public License (v2.0) (the "MPL"). Over the next ~9 years, it built up a community that included thousands of users, contributors, customers, certified practitioners, vendors, and an ecosystem of open-source modules, plugins, libraries, and extensions.

Then, on August 10th, 2023, with little or no advance notice or chance for much, if not all, of the community to have any input, HashiCorp switched the license for Terraform from the MPL to the Business Source License (v1.1) (the "BUSL"), a non-open source license. In our opinion, this change threatens the entire community and ecosystem that's built up around Terraform over the last 9 years.

Our concern: the BUSL license is a poison pill for Terraform.

Overnight, tens of thousands of businesses, ranging from one-person shops to the Fortune 500 woke up to a new reality where the underpinnings of their infrastructure suddenly

Supporters

Companies 161 | Projects 11 | Foundations 1 | Individuals 792

infrastructure, more and more, they'll pick alternatives that are genuinely open-source. Existing Terraform codebases will turn into outdated liabilities, independent tooling will all but disappear, and the community will fracture and disappear.

This sort of change also harms all similar open-source projects. Every company and every developer now needs to think twice before adopting and investing in an open-source

Tofu

The Need for Improved IaC Solutions

Community-Requested Improvements:

OpenTofu

The Need for Improved IaC Solutions

Community-Requested Improvements:

Security

Native solutions for state encryption that may include sensitive information.

OpenTofu

The Need for Improved IaC Solutions

Community-Requested Improvements:

Security

Native solutions for state encryption that may include sensitive information.

Flexibility

Simple methods for managing configurations across different environments.

OpenTofu

The Need for Improved IaC Solutions

Community-Requested Improvements:

Security
Native solutions for state encryption that may include sensitive information.

Flexibility
Simple methods for managing configurations across different environments.

Resource Management
Simple, built-in methods for excluding resources during deployment.

OpenTofu

The Need for Improved IaC Solutions

Community-Requested Improvements:

Security
Native solutions for state encryption that may include sensitive information.

Flexibility
Simple methods for managing configurations across different environments.

Resource Management
Simple, built-in methods for excluding resources during deployment.

Customization
Easy deployment to multiple regions by iterating over providers with different configurations.

OpenTofu

Ronny Orot

OpenTofu Core Team Member

Software Engineer at env0

OpenTofu

Features Tailored to Your Needs

State Encryption

Early Evaluation

Exclude Flag

Provider Iteration

OpenTofu

Before

State Encryption

Natively Securing Your Data

OpenTofu

Before

State Encryption

Natively Securing Your Data

Modules > terraform-aws-modules/rds-aurora

terraform-aws-modules/rds-aurora

Terraform module to create AWS RDS Aurora resources 🇺🇦

Owner	terraform-aws-modules
Latest version	v9.11.0
Published	12/19/2024

Inputs

Name	Description	Type
master_password	Password for the master DB user. Note that this may show up in logs, and it will be stored in the state file. Required unless `manage_master_user_password` is set to `true` or unless `snapshot_identifier` or `replication_source_identifier` is provided or unless a `global_cluster_identifier` is provided when the cluster is the secondary cluster of a global database	`string`

OpenTofu

Before

State Encryption

Natively Securing Your Data

Modules > terraform-aws-modules/rds-aurora

terraform-aws-modules/rds-aurora

Terraform module to create AWS RDS Aurora resources

Owner	terraform-aws-modules
Latest version	v9.11.0
Published	12/19/2024

Inputs

Name	Description	Type
master_password	Password for the master DB user. Note that this may show up in logs, and it will be stored in the state file. Required unless `manage_master_user_password` is set to `true` or unless `snapshot_identifier` or `replication_source_identifier` is provided or unless a `global_cluster_identifier` is provided when the cluster is the secondary cluster of a global database	`string`

Password for the master DB user. Note that this may show up in logs, and it will be stored in the state file.

OpenTofu

Before

State Encryption

Natively Securing Your Data

terraform.tfstate

```
{
  "module": "module.rds-aurora[0]",
  "type": "aws_rds_cluster",
  ...
  "instances": [
    {
      "index_key": 0,
      "schema_version": 0,
      "attributes": {
        "id": "dev-cluster",
        "master_password": "sensitive_secret123!",
      }
    }
  ]
}
```


OpenTofu

After

State Encryption

Natively Securing Your Data

OpenTofu

After

State Encryption

Natively Securing Your Data

main.tf

```
terraform {
  encryption {
    key_provider "aws_kms" "basic" {  ## Step 1: Add the desired key provider
      Kms_key_id = "some-id"
      key_spec = "AES_256"
    }

    method "aes_gcm" "new_method" {   ## Step 2: Set up your encryption method
      keys = key_provider.aws_kms.basic
    }

    state {                           ## Step 3: Link the desired encryption method
      method = method.aes_gcm.new_method
  }
}                                     ## Step 4: Run "tofu apply"
```

OpenTofu

Before

Early Evaluation

Flexible Management Across Different Environments

OpenTofu

Before

Early Evaluation

Flexible Management Across Different Environments

OpenTofu

Before

Early Evaluation

Flexible Management Across Different Environments

```
├── environments
│   ├── dev
│   │   ├── main.tf
│   │   ├── variables.tf
│   │   ├── outputs.tf
│   │   └── var.auto.tfvars
│   └── prod
│       ├── main.tf
│       ├── variables.tf
│       ├── outputs.tf
│       └── var.auto.tfvars
└── modules
    └── networking
        ├── main.tf
        └── variables.tf
```


OpenTofu

Before

Early Evaluation

Flexible Management Across Different Environments

```
├── environments
│   ├── dev
│   │   ├── main.tf
│   │   ├── variables.tf
│   │   ├── outputs.tf
│   │   └── var.auto.tfvars
│   └── prod
│       ├── main.tf
│       ├── variables.tf
│       ├── outputs.tf
│       └── var.auto.tfvars
└── modules
    └── networking
        ├── main.tf
        └── variables.tf
```


OpenTofu

Before

Early Evaluation

Flexible Management Across Different Environments

OpenTofu

Before

Early Evaluation

Flexible Management Across Different Environments

Ingredients

```
├── main.tf
├── variables.tf
├── outputs.tf
└── configs
    ├── backend-dev.tf
    └── backend-prod.tf
```


OpenTofu

Before

Early Evaluation

Flexible Management Across Different Environments

Ingredients

```
├── main.tf
├── variables.tf
├── outputs.tf
└── configs
    ├── backend-dev.tf
    └── backend-prod.tf
```

configs/backend-dev.tf

```
backend "s3" {
  bucket = "fe-dev-state"
  key    = "terraform.tfstate"
  region = "us-east-1"
}
```

configs/backend-prod.tf

```
backend "s3" {
  bucket = "fe-prod-state"
  key    = "terraform.tfstate"
  region = "us-east-1"
}
```

OpenTofu

Before

Early Evaluation

Flexible Management Across Different Environments

Ingredients

```
├── main.tf
├── variables.tf
├── outputs.tf
└── configs
    ├── backend-dev.tf
    └── backend-prod.tf
```

configs/backend-dev.tf

```
backend "s3" {
  bucket = "fe-dev-state"
  key    = "terraform.tfstate"
  region = "us-east-1"
}
```

configs/backend-prod.tf

```
backend "s3" {
  bucket = "fe-prod-state"
  key    = "terraform.tfstate"
  region = "us-east-1"
}
```

Instructions

```
// init and apply for dev
tofu init
  -backend-config=backend-dev.tf
```

```
// init and apply for prod
tofu init
  -backend-config=backend-prod.tf
```

OpenTofu

Before

Early Evaluation

Flexible Management Across Different Environments

Ingredients

```
├── main.tf
├── variables.tf
├── outputs.tf
└── configs
    ├── backend-dev.tf
    └── backend-prod.tf
```

configs/backend-dev.tf

```
backend "s3" {
  bucket = "fe-dev-state"
  key    = "terraform.tfstate"
  region = "us-east-1"
}
```

configs/backend-prod.tf

```
backend "s3" {
  bucket = "fe-prod-state"
  key    = "terraform.tfstate"
  region = "us-east-1"
}
```

Instructions

```
// init and apply for dev
tofu init
  -backend-config=backend-dev.tf
tofu apply -var="env=dev"

// init and apply for prod
tofu init
  -backend-config=backend-prod.tf
tofu apply -var="env=prod"
```

OpenTofu

Before

Early Evaluation

Flexible Management Across Different Environments

Ingredients

```
├── main.tf
├── variables.tf
├── outputs.tf
└── configs
    ├── backend-dev.tf
    └── backend-prod.tf
```

configs/backend-dev.tf

```
backend "s3" {
  bucket = "fe-dev-state"
  key    = "terraform.tfstate"
  region = "us-east-1"
}
```

configs/backend-prod.tf

```
backend "s3" {
  bucket = "fe-prod-state"
  key    = "terraform.tfstate"
  region = "us-east-1"
}
```

Instructions

```
// init and apply for dev
tofu init
  -backend-config=backend-dev.tf
tofu apply -var="env=dev"

// init and apply for prod
tofu init
  -backend-config=backend-prod.tf
tofu apply -var="env=prod"
```

OpenTofu

Before

Early Evaluation

Flexible Management Across Different Environments

Ingredients

```
├── main.tf
├── variables.tf
├── outputs.tf
└── configs
    ├── backend-dev.tf
    └── backend-prod.tf
```

configs/backend-dev.tf

```
backend "s3" {
  bucket = var.bucket_name
  key    = "terraform.tfstate"
  region = "us-east-1"
}
```

configs/backend-prod.tf

```
backend "s3" {
  bucket = var.bucket_name
  key    = "terraform.tfstate"
  region = "us-east-1"
}
```

Instructions

```
// init and apply for dev
tofu init
  -backend-config=backend-dev.tf
tofu apply -var="env=dev"

// init and apply for prod
tofu init
  -backend-config=backend-prod.tf
tofu apply -var="env=prod"
```

OpenTofu

After

Early Evaluation

Flexible Management Across Different Environments

OpenTofu

After

Early Evaluation

Flexible Management Across Different Environments

Ingredients

```
├── main.tf
├── variables.tf
├── outputs.tf
└── backend.tf
```


OpenTofu

After

Early Evaluation

Flexible Management Across Different Environments

Ingredients

```
├── main.tf
├── variables.tf
├── outputs.tf
└── backend.tf
```

backend.tf

```
variable "env" {}

locals {
 env_to_bucket = {
    dev  = "fe-dev-state"
    prod = "fe-prod-state"
  }
}

terraform {
  backend "s3" {
    bucket =
      local.env_to_bucket[var.env]
    key    = "terraform.tfstate"
    region = "us-east-1"
  }
}
```


OpenTofu

After

Early Evaluation

Flexible Management Across Different Environments

Ingredients

```
├── main.tf
├── variables.tf
├── outputs.tf
└── backend.tf
```

backend.tf

```
variable "env" {}

locals {
 env_to_bucket = {
    dev  = "fe-dev-state"
    prod = "fe-prod-state"
  }
}

terraform {
  backend "s3" {
    bucket =
      local.env_to_bucket[var.env]
    key    = "terraform.tfstate"
    region = "us-east-1"
  }
}
```


OpenTofu

After

Early Evaluation

Flexible Management Across Different Environments

Ingredients

```
├── main.tf
├── variables.tf
├── outputs.tf
└── backend.tf
```

backend.tf

```
variable "env" {}

locals {
 env_to_bucket = {
    dev  = "fe-dev-state"
    prod = "fe-prod-state"
  }
}

terraform {
  backend "s3" {
    bucket =
      local.env_to_bucket[var.env]
    key    = "terraform.tfstate"
    region = "us-east-1"
  }
}
```


OpenTofu

After

Early Evaluation

Flexible Management Across Different Environments

Ingredients

```
├── main.tf
├── variables.tf
├── outputs.tf
└── backend.tf
```

backend.tf

```
variable "env" {}

locals {
 env_to_bucket = {
    dev  = "fe-dev-state"
    prod = "fe-prod-state"
  }
}

terraform {
  backend "s3" {
    bucket =
      local.env_to_bucket[var.env]
    key    = "terraform.tfstate"
    region = "us-east-1"
  }
}
```

Instructions

```
// init and apply for dev
tofu init -var="env=dev"
tofu apply -var="env=dev"

// init and apply for prod
tofu init -var="env=prod"
tofu apply -var="env=prod"
```


OpenTofu

Before

Exclude Flag

Simplifying Resource Management

OpenTofu

Before

Exclude Flag

Simplifying Resource Management

```
tofu apply
```

main.tf

```
resource "local_file" "first" {
  filename = "first.txt"
}

resource "local_file" "second" {
  filename = "second.txt"
}

resource "local_file" "third" {
  filename = "third.txt"
}
```


OpenTofu

Before

Exclude Flag

Simplifying Resource Management

```
tofu apply -target local_file.first
```

main.tf

```
resource "local_file" "first" {
  filename = "first.txt"
}

resource "local_file" "second" {
  filename = "second.txt"
}

resource "local_file" "third" {
  filename = "third.txt"
}
```

Only deploy this resource

OpenTofu

Before

Exclude Flag

Simplifying Resource Management

feature request: inverse targeting / exclude #2253

Open

shubhambhartiya opened on Jun 6, 2015

Is there anything that can be done such that db_instance - RDS formed by the terraform files can be saved if we destroy the whole state?

👍 1472 😄 89 🎉 98 😕 42 ❤️ 121 🚀 82 👀 85

OpenTofu

After

Exclude Flag

Simplifying Resource Management

OpenTofu

After

Exclude Flag

Simplifying Resource Management

```
tofu apply -exclude local_file.third
```

main.tf

```
resource "local_file" "first" {
  filename = "first.txt"
}

resource "local_file" "second" {
  filename = "second.txt"
}

resource "local_file" "third" {
  filename = "third.txt"
}
```

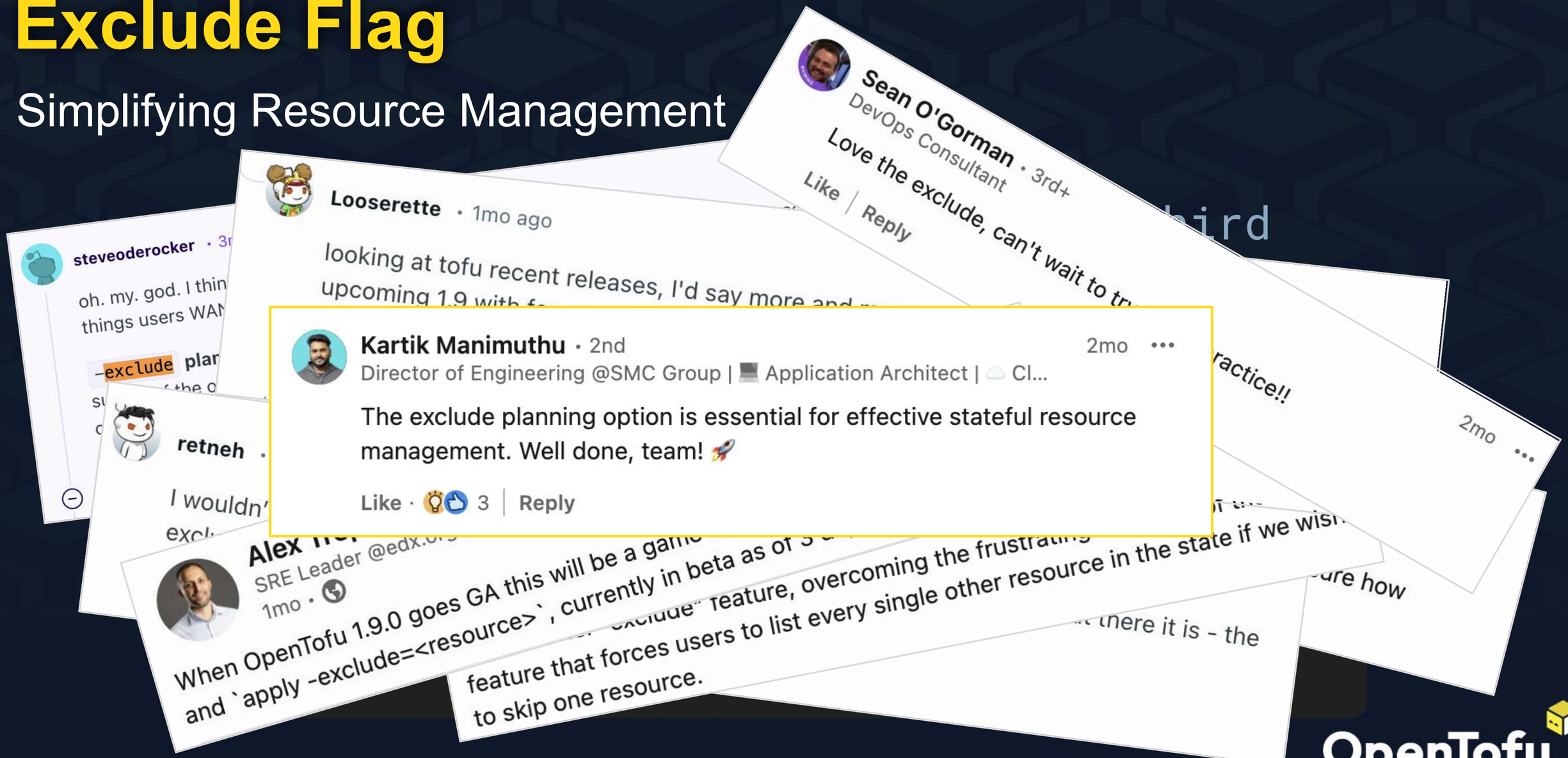
Deploy only these resources

OpenTofu

After

Exclude Flag

Simplifying Resource Management



OpenTofu

Before

Provider Iteration

Customize Provider Settings for Global Deployments

OpenTofu

Before

Provider Iteration

Customize Provider Settings for Global Deployments

main.tf

```
provider "aws" {                                  ## Two provider blocks
  alias   = "us"
  region  = "us-east-1"
}

provider "aws" {
  alias   = "eu"
  region  = "eu-west-1"
}

resource "aws_rds_cluster" "us_cluster" { ## Two resource blocks
  provider = aws.us
}

resource "aws_rds_cluster" "eu_cluster" {
  provider = aws.eu
}
```


OpenTofu

After

Provider Iteration

Customize Provider Settings for Global Deployments

OpenTofu

After

Provider Iteration

Customize Provider Settings for Global Deployments

main.tf

```
variable "regions" {
  description = "A list of regions to deploy"
  type        = set(string)
}

provider "aws" {                          ## A single provider block
  alias    = "by_region"
  for_each = var.regions
  region   = each.value
}

resource "aws_rds_cluster" "rds" {   ## A single resource block
  for_each = var.regions
  provider = aws.by_region[each.key]
}
```


OpenTofu

After

Provider Iteration

Customize Provider Settings for Global Deployments

main.tf

```
variable "regions" {
  type        = set(string)
}

provider "aws" {
  alias    = "by_region"
  for_each = var.regions
  region   = each.value
}

module "user_service" {                    ## A single module block
  source   = "./user-service"
  for_each = var.regions
  providers = {
    aws = aws.by_region[each.key]
  }
}
```


OpenTofu

`.tofu` File Extension

A Native Solution for OpenTofu Configurations

OpenTofu

`.tofu` File Extension

A Native Solution for OpenTofu Configurations

- main.tf
- backend.tf
- variables.tf

OpenTofu

`.tofu` File Extension

A Native Solution for OpenTofu Configurations

main.tf

backend.tf → backend.tofu

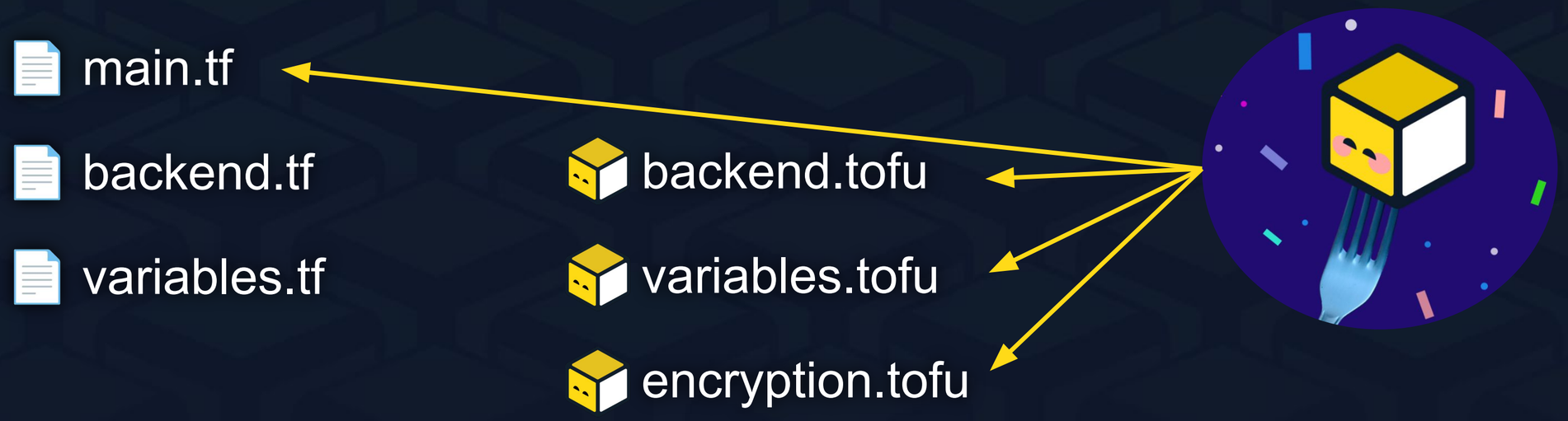
variables.tf → variables.tofu

encryption.tofu

OpenTofu

`.tofu` File Extension

A Native Solution for OpenTofu Configurations



OpenTofu

`.tofu` File Extension

A Native Solution for OpenTofu Configurations

OpenTofu Users

For a smooth migration, explore OpenTofu's exclusive features while retaining the option to easily revert changes.

Module Authors

Enable module authors to create modules that are compatible with both Terraform and OpenTofu.

OpenTofu

OpenTofu’s Compatibility with Terraform

OpenTofu remains almost completely compatible with Terraform:

- Providers
- Modules
- Configurations and State File

OpenTofu

OpenTofu's Compatibility with Terraform

OpenTofu remains almost completely compatible with Terraform:

- Providers
- Modules
- Configurations and State File

Features to Note:

- Removed block
- Provider-defined functions
- `for_each` on import blocks
- Testing feature

OpenTofu

Sneak Peek: What's Coming Next?

OCI Registries

RFCs

DynamoDB Free S3 Backend

OpenTofu

Join the OpenTofu Movement!

We talked about:

- State Encryption
- Early Evaluation
- Exclude Flag
- Provider Iteration
- `.tofu` File Extension
- Compatibility with Terraform

OpenTofu

Join the OpenTofu Movement!

Cast your vote:

Top-Ranking Issues 📊 #1496

Open

cube2222 opened on Apr 16, 2024 · edited by github-actions

Edits

Top Enhancements

1. Support OCI registries for provider and module distribution. #308 - 114 👍
2. Support OpenTofu in the VS Code Language Server #970 - 100 👍
3. Direct support for conditional single-instance resources #1306 - 97 👍
4. tofu plan -light #1703 - 91 👍
5. Unlocking Terraform's S3 Backend: Going DynamoDB-Free #599 - 75 👍

OpenTofu

Join the OpenTofu Movement!

Cast your vote:

Top-Ranking Issues 📊 #1496

Open

cube2222 opened on Apr 16, 2024 · edited by github-actions

Edits

Top Enhancements

1. Support OCI registries for provider and module distribution. #308 - 114 👍
2. Support OpenTofu in the VS Code Language Server #970 - 100 👍
3. Direct support for conditional single-instance resources #1306 - 97 👍
4. tofu plan -light #1703 - 91 👍
5. Unlocking Terraform's S3 Backend: Going DynamoDB-Free #599 - 75 👍

OpenTofu

Thank You!

Join our Slack

OpenTofu