goto;

GOTO
CHICAGO 2023

#GOTOchgo

goto;

Catching Commits to Secure Infrastructure as Code

May 23, 2023

OWASP Secure Coding Practices

- Input Validation
- Output Encoding
- Authentication and Password Management
- Session Management
- Access Control
- Cryptographic Practices
- Error Handling and Logging
- Data Protection
- Communication Security
- System Configuration
- Database Security
- File Management
- Memory Management
- General Coding Practices

https://owasp.org/www-project-secure-coding-practices-quick-reference-guide/

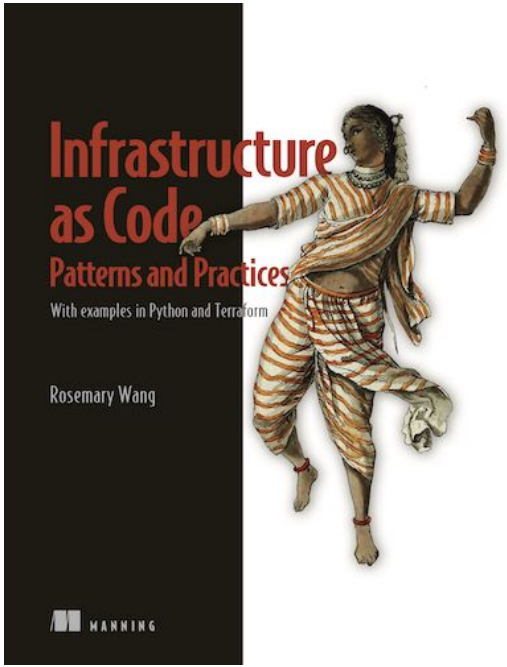
What about infrastructure as code?

Rosemary Wang

Developer Advocate, HashiCorp

joatmon08.github.io

@joatmon08



How do you write **secure** infrastructure as code?

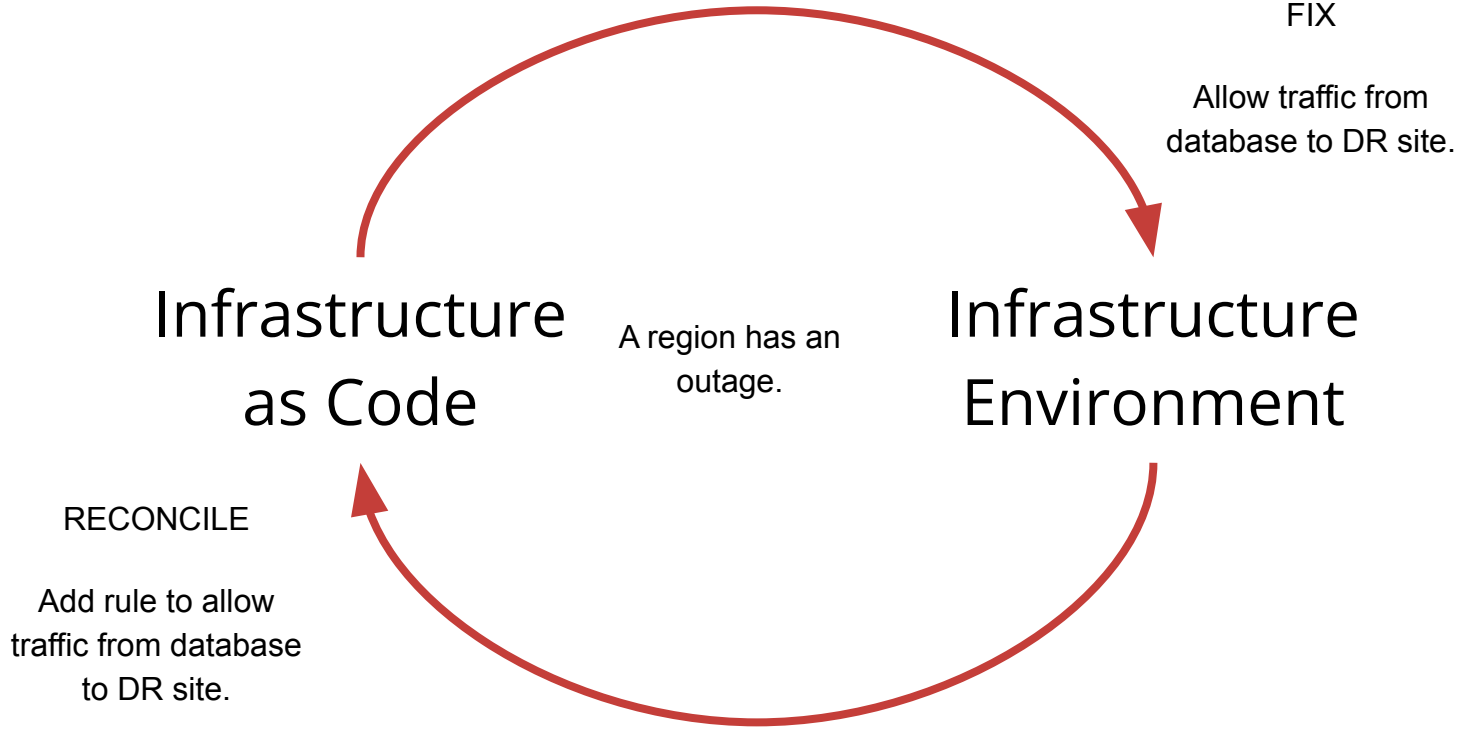
Security Testing

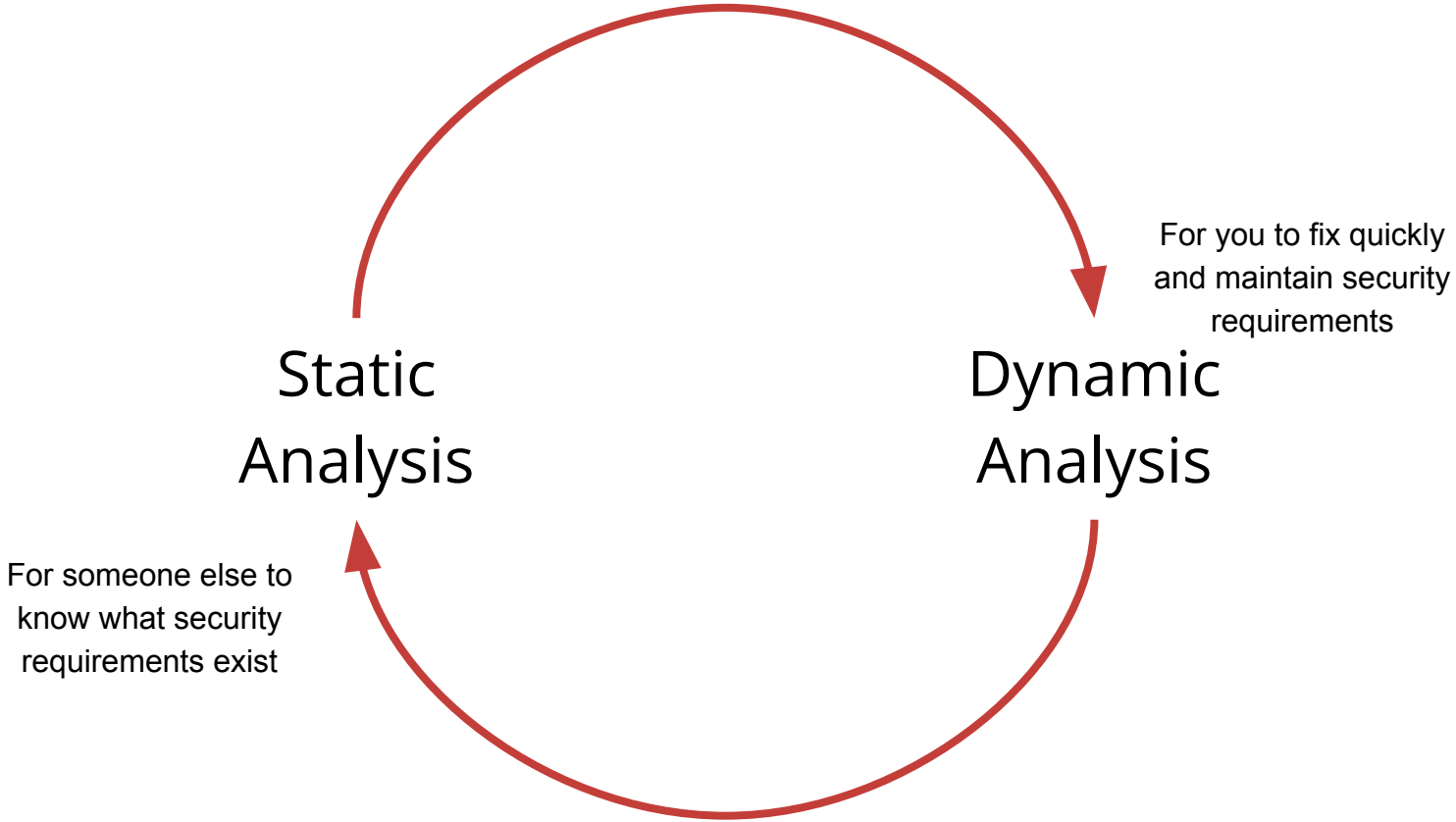
Static Analysis

- “Shift-left” security testing
- Policy as code
- Unit / contract testing for security

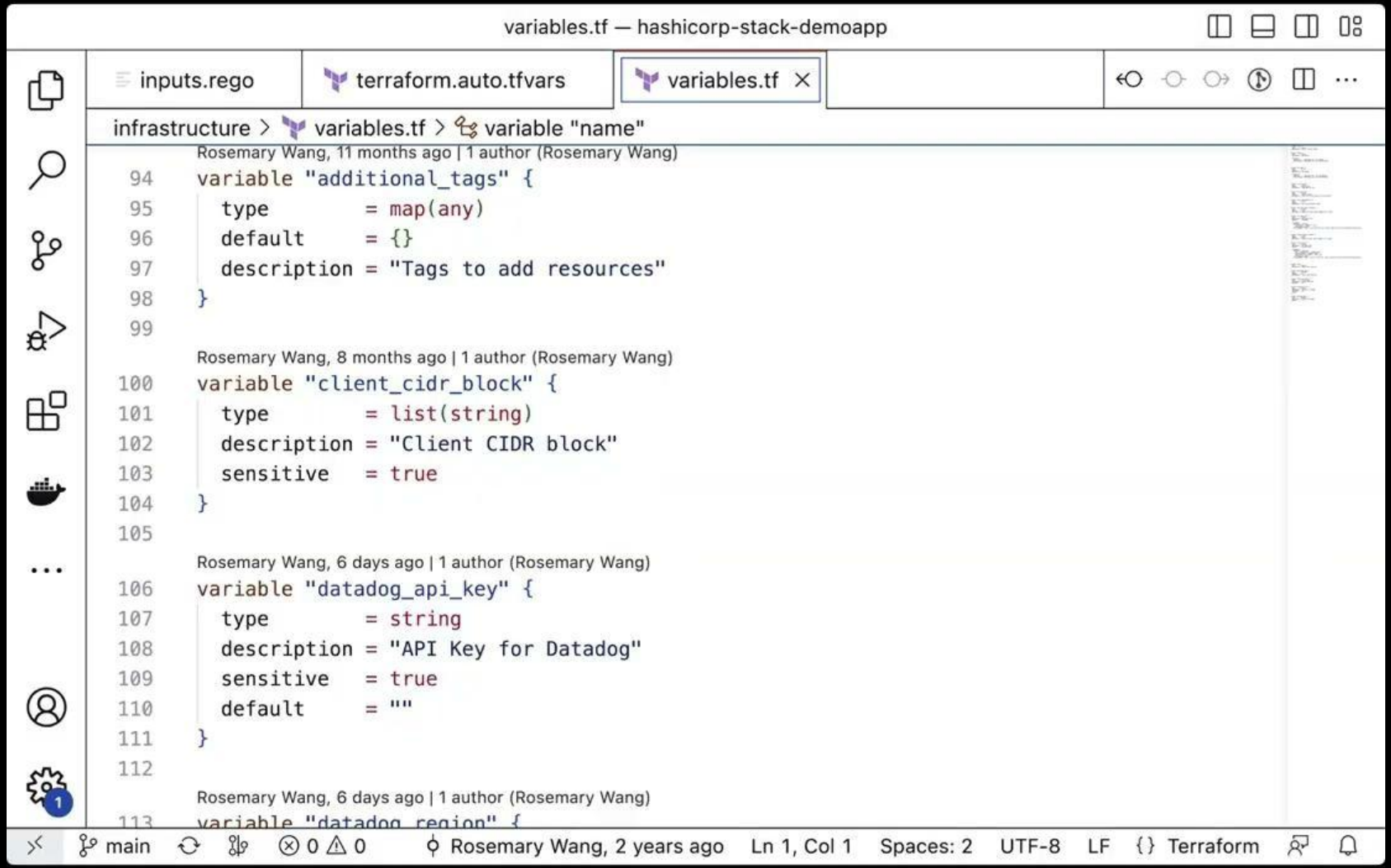
Dynamic Analysis

- Vulnerability scanning
- Continuous validation
- Security monitoring

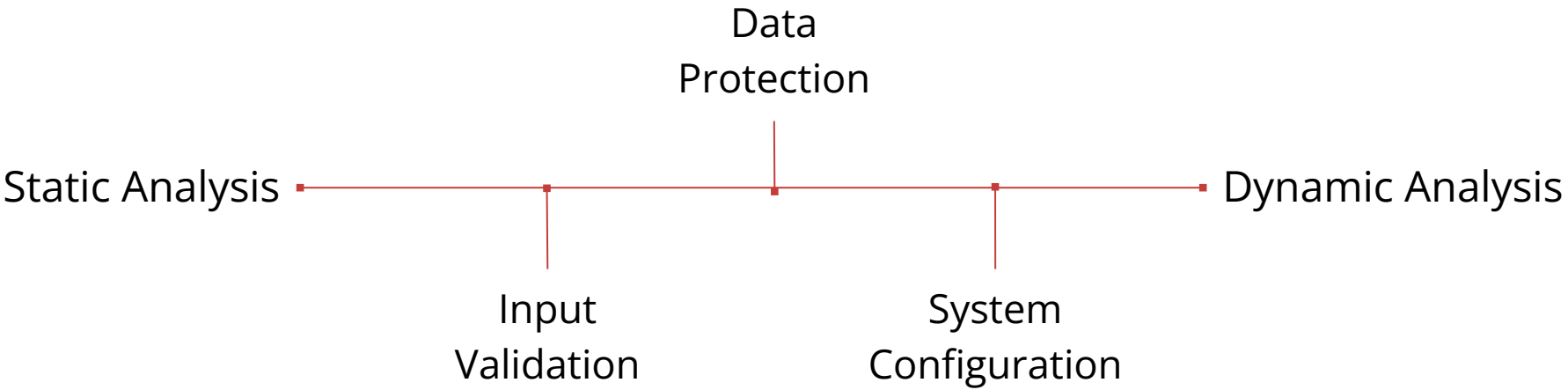

FIX
Allow traffic from database to DR site.
Infrastructure as Code
A region has an outage.
Infrastructure Environment
RECONCILE
Add rule to allow traffic from database to DR site.


Static Analysis
Dynamic Analysis
For you to fix quickly and maintain security requirements
For someone else to know what security requirements exist

Capture **security knowledge** as tests.


variables.tf — hashicorp-stack-demoapp
inputs.rego
terraform.auto.tfvars
variables.tf
infrastructure > variables.tf > variable "name"
Rosemary Wang, 11 months ago | 1 author (Rosemary Wang)
94 variable "additional_tags" {
95 type = map(any)
96 default = {}
97 description = "Tags to add resources"
98 }
99
Rosemary Wang, 8 months ago | 1 author (Rosemary Wang)
100 variable "client_cidr_block" {
101 type = list(string)
102 description = "Client CIDR block"
103 sensitive = true
104 }
105
Rosemary Wang, 6 days ago | 1 author (Rosemary Wang)
106 variable "datadog_api_key" {
107 type = string
108 description = "API Key for Datadog"
109 sensitive = true
110 default = ""
111 }
112
Rosemary Wang, 6 days ago | 1 author (Rosemary Wang)
main
0 0
Rosemary Wang, 2 years ago
Ln 1, Col 1
Spaces: 2
UTF-8
LF
Terraform

What commits to catch?



Input Validation

Input Validation

Verify...

- Expected types
- Expected values
- Proper character sets
- Potentially insecure values

Static analysis

Input Validation

Examples:

- Password should have at least 1 symbol and 1 uppercase character.
- Region should be in United States.
- Private CA certificate key should be marked as sensitive.
- Names should include standard environment.

Input Validation

Variable Validation

Better for...

- Team
- Provider
- Modules

Unit Test

Better for...

- Multiple providers
- Organizational policy
- Dependencies

```
variable "region" {
 type        = string
 description = "AWS Region"

 validation {
   condition     = can(regex("^us-", var.region))
   error_message = "AWS Region must be in United
                    States"
 }
}
```

Variable Validation

```
resource "random_string" "boundary" {
 ## omitted
 lifecycle {
   precondition {
     condition     = random_string.boundary.length > 3
     error_message = "HCP Boundary requires username to be
                      at least 3 characters in length"
   }
 }
}
```

Variable Validation, Inline with PreCondition

```
deny[msg] {
    r := tfplan.variables
    r.aws_secret_access_key
    msg := "do not define AWS secret access key as part of
           variables, use AWS_SECRET_ACCESS_KEY environment
           variable instead"
}
```

Data Protection

Data Protection

- Implement access controls to state
  - Data
  - State
- Encrypt in transit and at rest.
- Sanitize sensitive values in logs or outputs
- Ensure least privilege access to providers

Static analysis

Dynamic analysis

Data Protection

Examples:

- Database should be encrypted.
- Password should not be printed in output.
- Virtual machine resource should have attached IAM role.
- Infrastructure state should be limited to owners of workspace.

Data Protection

Static Analysis

Better for...

- Enforcement of secure practices
- Testing valid dependencies

Dynamic Analysis

Better for...

- Least-privilege API access
- Continuous enforcement
- Auditing data access

```
deny[msg] {
    r := resource_changes[_]
    r.type == "aws_db_instance"
    r.change.after.storage_encrypted == ""
    msg := sprintf("%v should have encrypted storage",
           [r.address])
}
```

```
deny[msg] {
    outputs := planned_values.outputs
    plaintext_password_outputs := [key |
        outputs[key]
        contains(key, "password")
        not outputs[key].sensitive
    ]
    count(plaintext_password_outputs) != 0
    msg := sprintf("%v should be marked as sensitive
           outputs", [plaintext_password_outputs])
}
```

```
resource "aws_db_instance" "products" {
 ## omitted
 storage_encrypted       = true

 lifecycle {
   postcondition {
     condition     = self.storage_encrypted
     error_message = "encrypt AWS RDS database storage"
   }
 }
}
```

System Configuration

System Configuration

- Check versions or images
- Ensure least privilege network access
- Separate development and production
- Analyze vulnerabilities and access
- Assess drift
- Remove idle / unused resources

Need dynamic analysis

System Configuration

Examples:

- Separate state for development and production.
- Verify network policies and secure versions.
- Tags should include environment.
- Image should include secure base.
- Scan running infrastructure for new vulnerabilities.

System Configuration

Static Analysis

Better for...

- Enforcement of secure practices
- Complex logic for validation
- Testing valid dependencies

Dynamic Analysis

Better for...

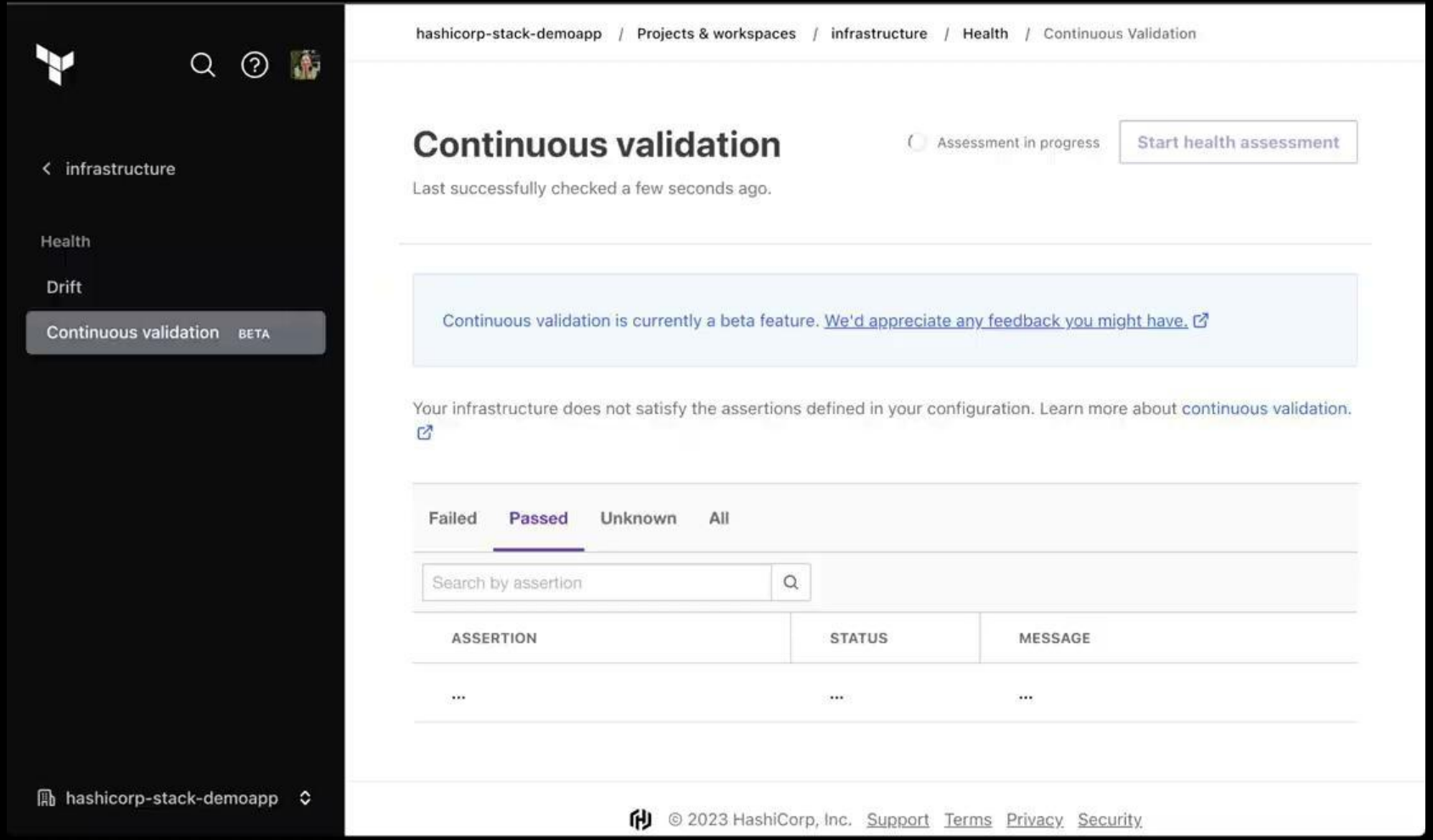
- Network auditing
- Continuous validation
- Drift detection

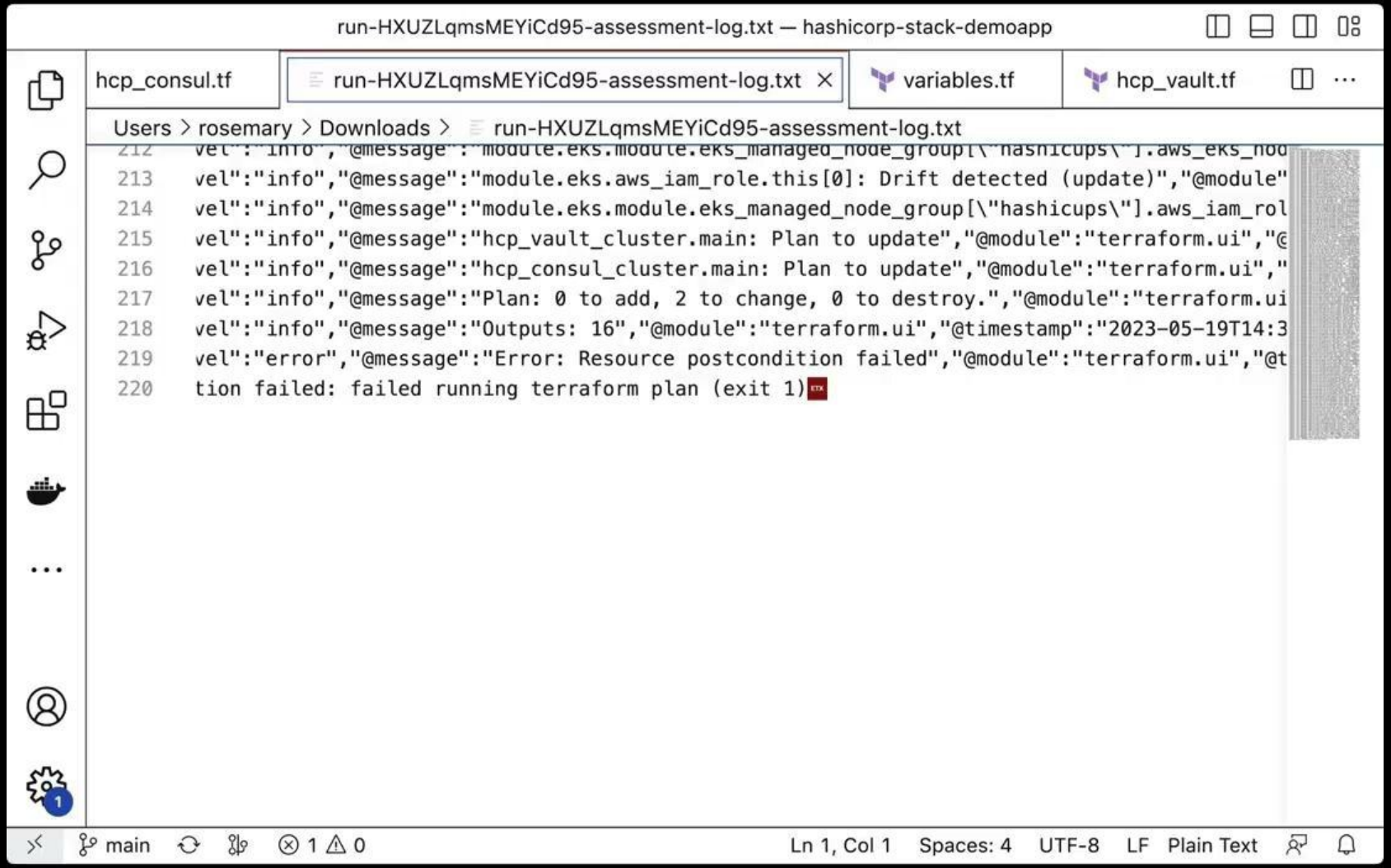
```
deny[msg] {
     ## omitted logic to parse VPC
     private_subnets := {r.values.id |
          r := vpc_modules[_].resources[_]
          r.type == "aws_subnet"
          not r.values.map_public_ip_on_launch
     }

     public_subnets := cluster_subnet_ids - private_subnets

     count(public_subnets) != 0
     msg := sprintf("EKS cluster %v should be in private
               subnets (%v are public subnets)",
               [cluster_id, public_subnets])
}
```

More Complex Static Analysis with OPA


hashicorp-stack-demoapp / Projects & workspaces / infrastructure / Health / Continuous Validation
infrastructure
Health
Drift
Continuous validation BETA
Continuous validation
Assessment in progress
Start health assessment
Last successfully checked a few seconds ago.
Continuous validation is currently a beta feature. We'd appreciate any feedback you might have.
Your infrastructure does not satisfy the assertions defined in your configuration. Learn more about continuous validation.
Failed
Passed
Unknown
All
Search by assertion
ASSERTION
STATUS
MESSAGE
...
...
...
hashicorp-stack-demoapp
© 2023 HashiCorp, Inc. Support Terms Privacy Security

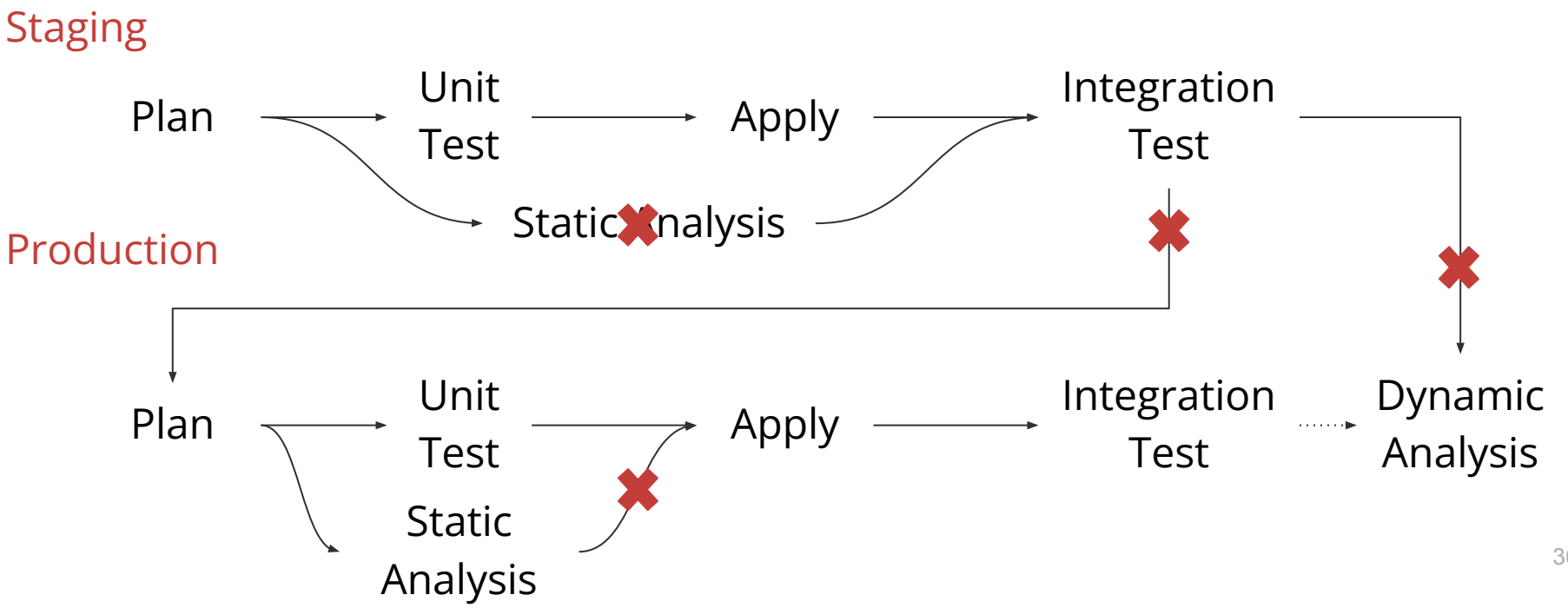

run-HXUZLqmsMEYiCd95-assessment-log.txt — hashicorp-stack-demoapp
hcp_consul.tf
run-HXUZLqmsMEYiCd95-assessment-log.txt
variables.tf
hcp_vault.tf
Users > rosemary > Downloads > run-HXUZLqmsMEYiCd95-assessment-log.txt
vel":"info","@message":"module.eks.aws_iam_role.this[0]: Drift detected (update)","@module"
vel":"info","@message":"module.eks.module.eks_managed_node_group[\"hashicups\"].aws_iam_rol
vel":"info","@message":"hcp_vault_cluster.main: Plan to update","@module":"terraform.ui","@
vel":"info","@message":"hcp_consul_cluster.main: Plan to update","@module":"terraform.ui","
vel":"info","@message":"Plan: 0 to add, 2 to change, 0 to destroy.","@module":"terraform.ui
vel":"info","@message":"Outputs: 16","@module":"terraform.ui","@timestamp":"2023-05-19T14:3
vel":"error","@message":"Error: Resource postcondition failed","@module":"terraform.ui","@t
tion failed: failed running terraform plan (exit 1)
main
1 0
Ln 1, Col 1
Spaces: 4
UTF-8
LF
Plain Text

There’s more!

How do I share tests?

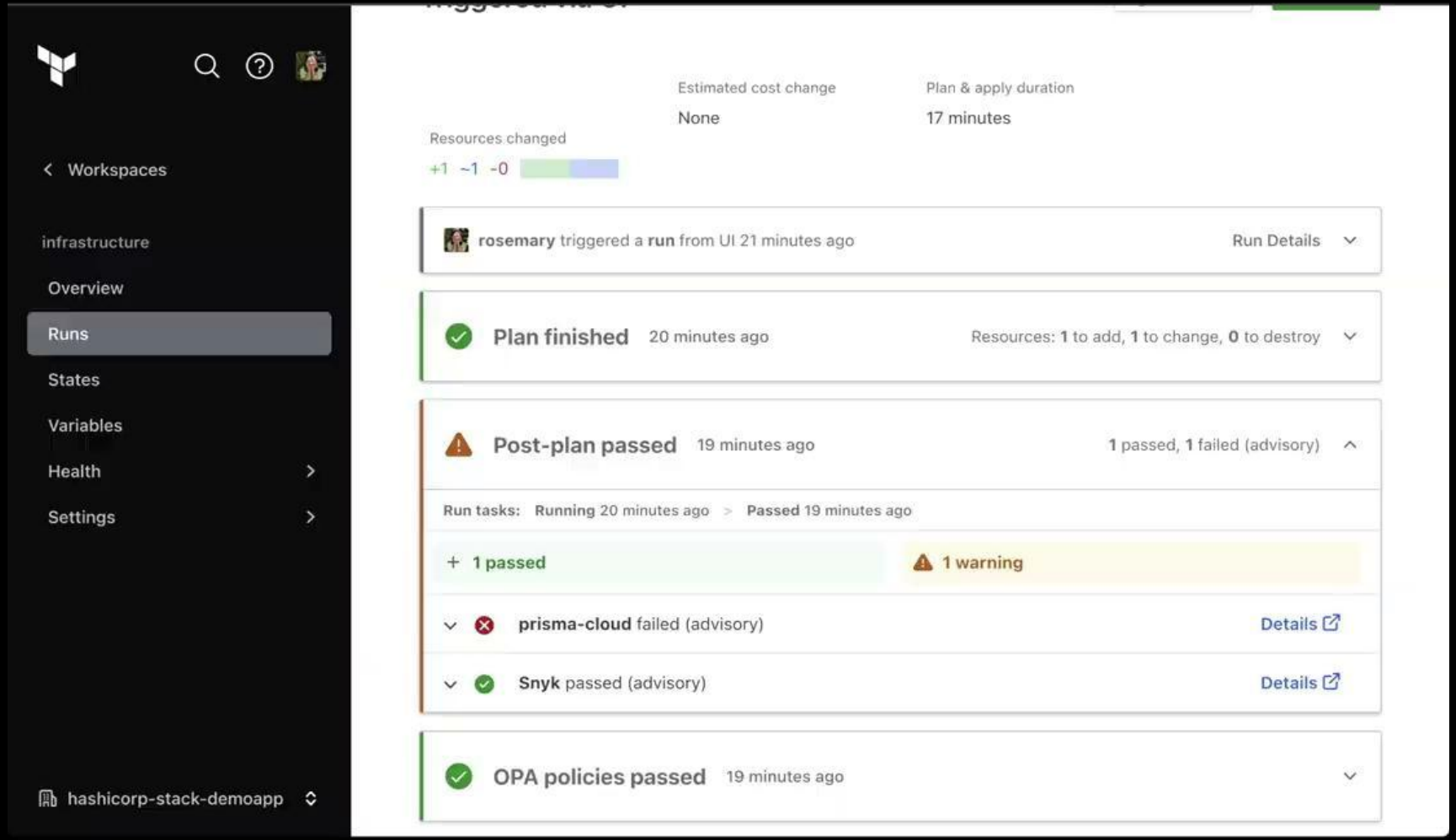
- Store tests in central location.
- Tag / name / number tests for relevance.
- Document secure configurations in tests.

How to catch commits?



Do I write them all myself?

1. Use pre-written test suites or industry benchmarks.
   - Infrastructure modules
   - Production configuration
2. Create custom policy tests.
   - Divide by...
     - Enforcement level
     - Business unit
     - Resource
     - Type


Workspaces
infrastructure
Overview
Runs
States
Variables
Health
Settings
hashicorp-stack-demoapp
Estimated cost change
None
Plan & apply duration
17 minutes
Resources changed
+1 ~1 -0
rosemary triggered a run from UI 21 minutes ago
Run Details
Plan finished
20 minutes ago
Resources: 1 to add, 1 to change, 0 to destroy
Post-plan passed
19 minutes ago
1 passed, 1 failed (advisory)
Run tasks: Running 20 minutes ago > Passed 19 minutes ago
1 passed
1 warning
prisma-cloud failed (advisory)
Details
Snyk passed (advisory)
Details
OPA policies passed
19 minutes ago

What is important?

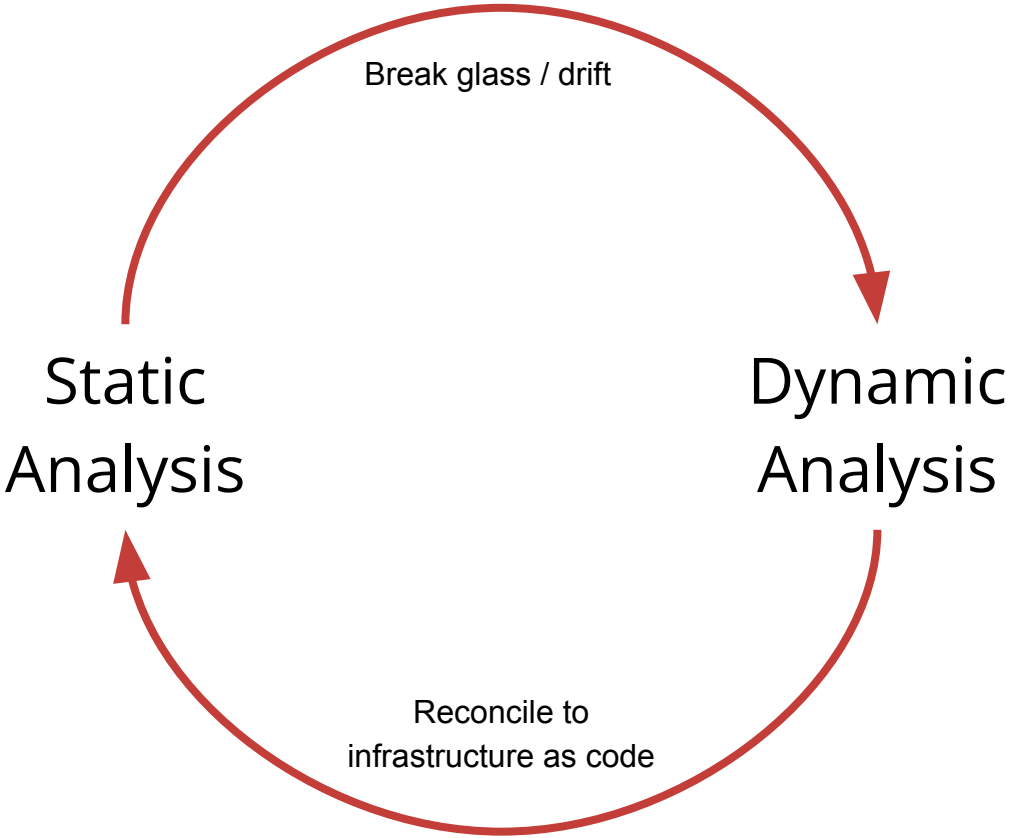
- Prefer secure defaults over tests.
- Choose a severity threshold.
  - Identify mandatory rules.
  - Use advisory as last resort.
  - Try to enforce development environments.
- Evaluate and make exceptions.

Conclusion

How do you write secure IaC?

- Capture practices into tests
- Share tests
- Catch commits using a pipeline
- Write what you must
- Choose what is important

Iterate


Break glass / drift
Static Analysis
Dynamic Analysis
Reconcile to infrastructure as code

More **secure** infrastructure as code over time.

Thank you!

@joatmon08

- github.com/joatmon08/hashicorp-stack-demoapp
- developer.hashicorp.com/terraform/tutorials/cloud-get-started/policy-quickstart
- openpolicyagent.org/docs/latest/terraform/

goto;

Don’t forget to
vote for this session
in the **GOTO Guide app**