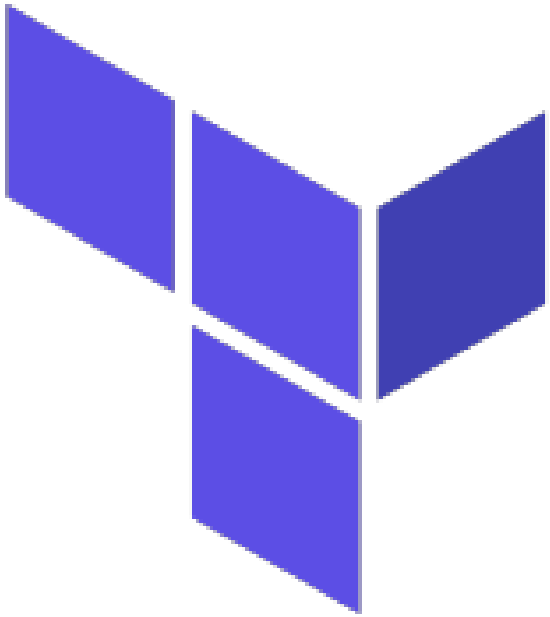
Infrastructure Provisioning with Terraform

MIGUEL ÁNGEL DOMÍNGUEZ COLOMA

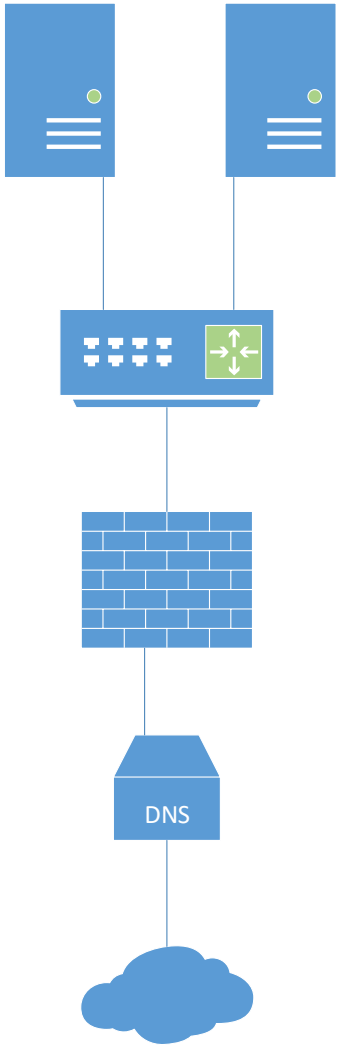
30 minutes to:

- **What is Terraform**



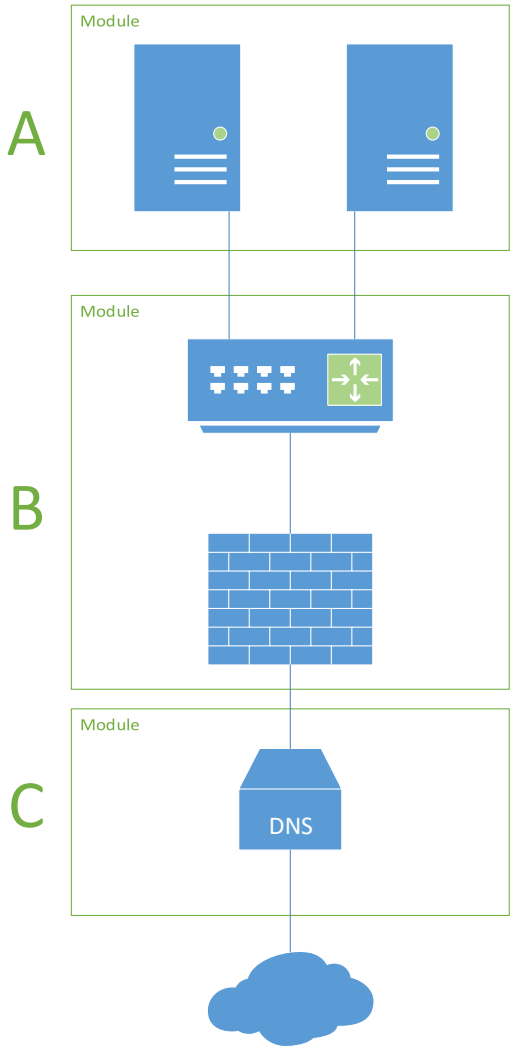
30 minutes to:

- What is Terraform
- **Set up infrastructure**
  - Instances
  - Network
  - Security
  - DNS



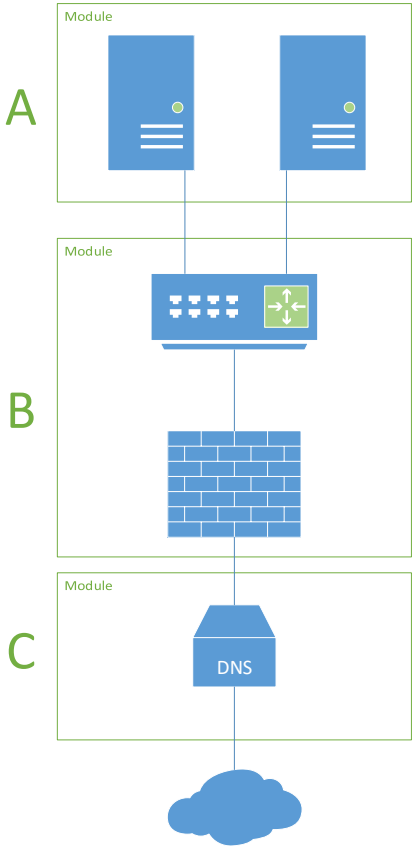
30 minutes to:

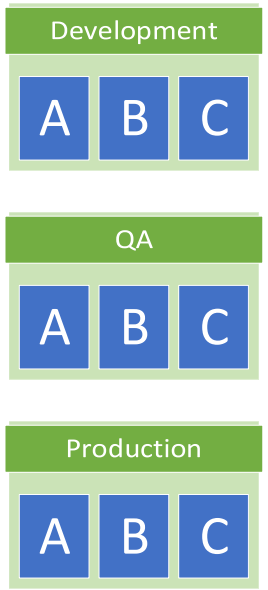
- What is Terraform
- Set up infrastructure
  - Instances
  - Network
  - Security
  - DNS
- **Modules**



30 minutes to:

- What is Terraform
- Set up infrastructure
  - Instances
  - Network
  - Security
  - DNS
- Modules
- **Environments**





Terraform

Terraform

files

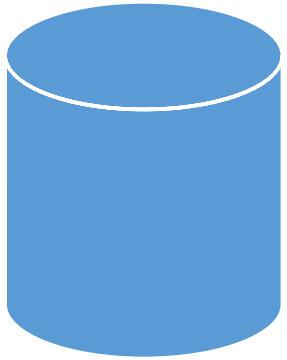
+

terraform

=

API

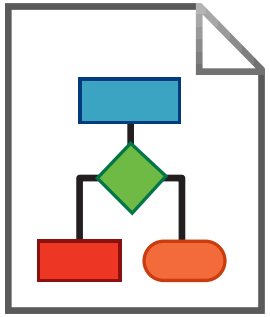
Terraform



state

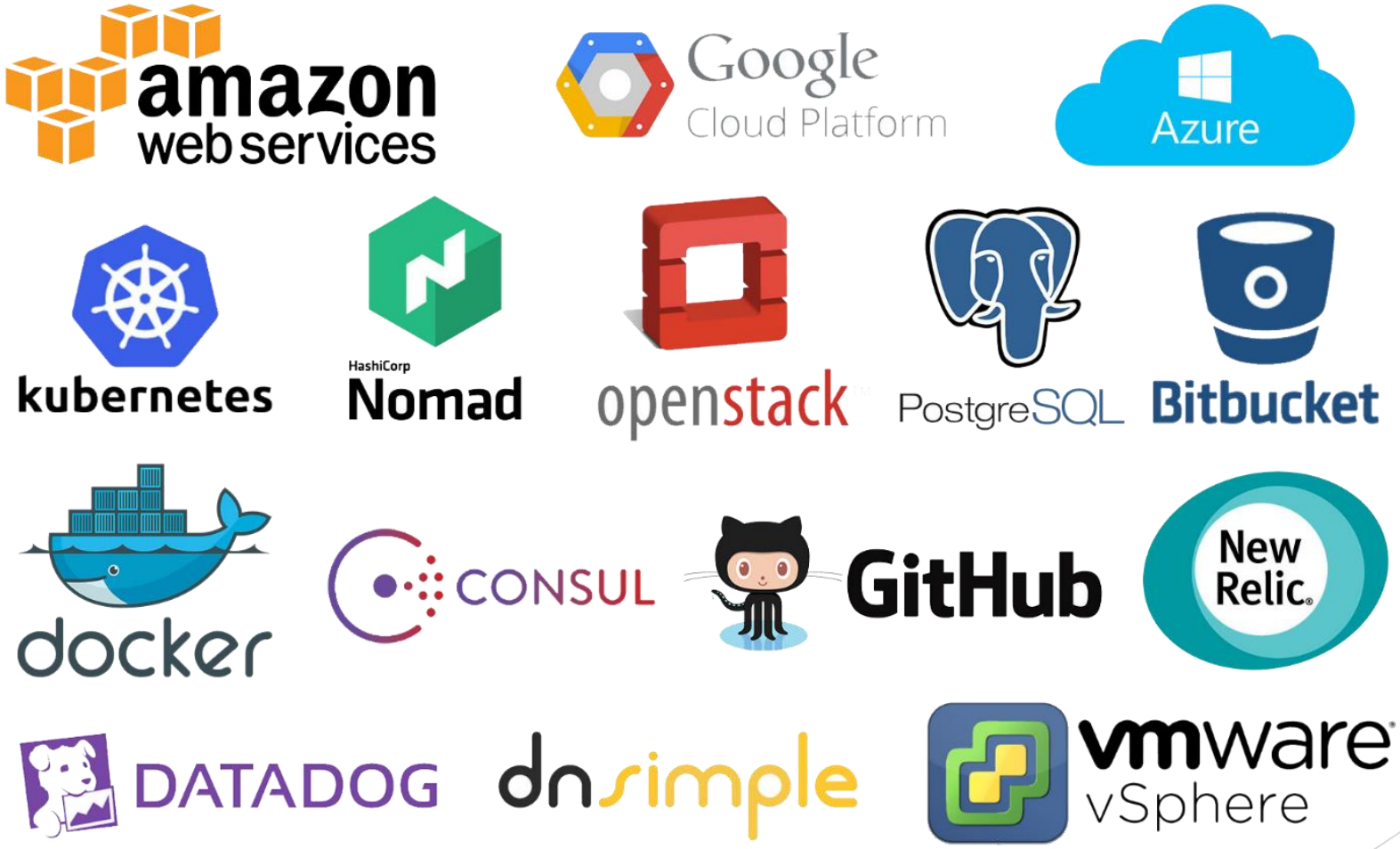
Work collaboratively

Pre-plan changes



Visualization

Terraform


amazon
web services
Google
Cloud Platform
Azure
kubernetes
HashiCorp
Nomad
openstack
PostgreSQL
Bitbucket
docker
CONSUL
GitHub
New
Relic
DATADOG
dnsimple
vmware
vSphere

Terraform

variables

providers

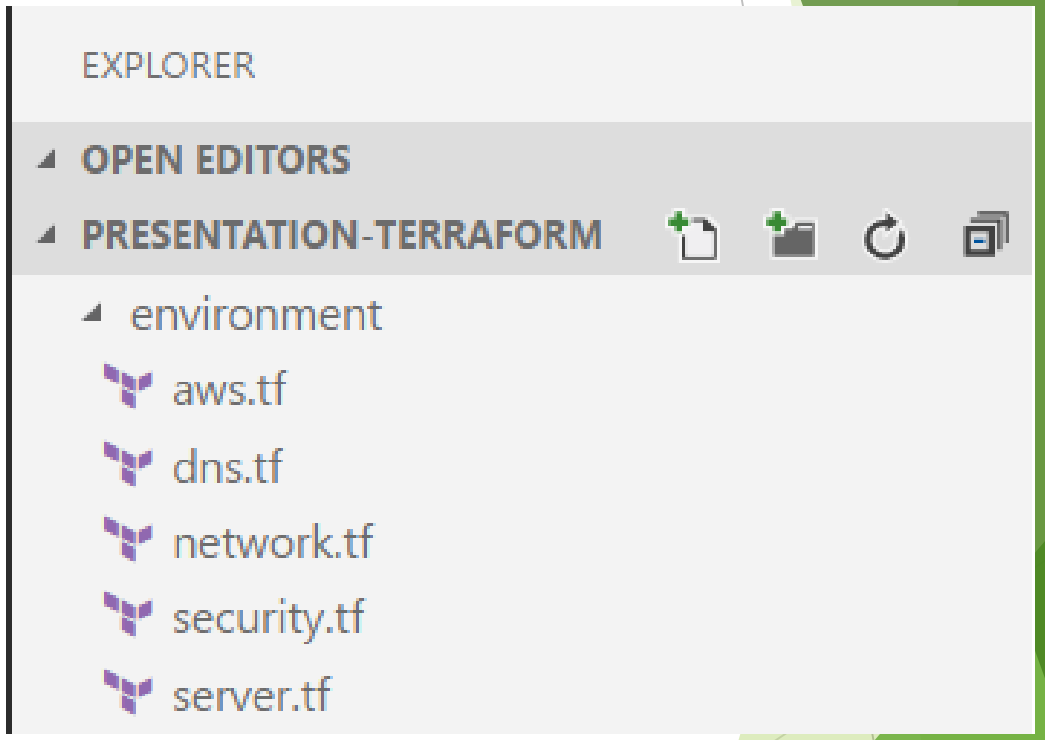
resources

Background

Background

What do we need

`terraform` binary



Instances

AWS Provider

- /aws.tf

```
variable "aws_access_key" {}
variable "aws_secret_key" {}
variable "aws_default_region" {
 default = "eu-west-1"
}
```

```
provider "aws" {
 access_key = "${var.aws_access_key}"
 secret_key = "${var.aws_secret_key}"
 region     = "${var.aws_default_region}"
}
```

Creating instances

- /server.tf

```
variable "ami"           { default = "ami-785db401" }
variable "instance_type" { default = "t2.micro" }
```

```
resource "aws_instance" "my_product" {
 ami           = "${var.ami}"
 instance_type = "${var.instance_type}"

 tags {
  Name = "WebServer"
 }
}
```

Applying changes

- /

```
terraform init
```

```
TF_VAR_aws_access_key=XXXXX \
TF_VAR_aws_secret_key=YYYYY \
terraform apply
```

aws Services Resource Groups

EC2 Dashboard
Events
Tags
Reports
Limits

INSTANCES
Instances
Spot Requests
Reserved Instances
Scheduled Instances
Dedicated Hosts

IMAGES
AMIs
Bundle Tasks

ELASTIC BLOCK STORE
Volumes
Snapshots

NETWORK & SECURITY
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces

LOAD BALANCING
Load Balancers
Target Groups

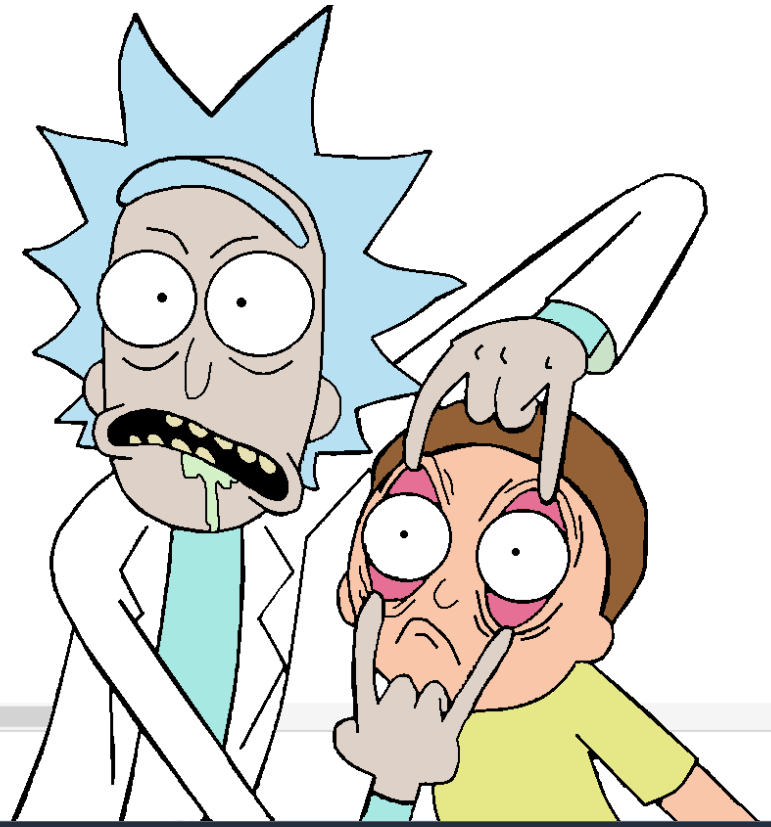
AUTO SCALING
Launch Configurations

Launch Instance | Connect | Actions

search : WebServer | Add filter

1 to 1 of 1

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP
WebServer	i-03547ffa9ba71ff11	t2.micro	eu-west-1a	running	2/2 checks ...	None		54.246.158.127



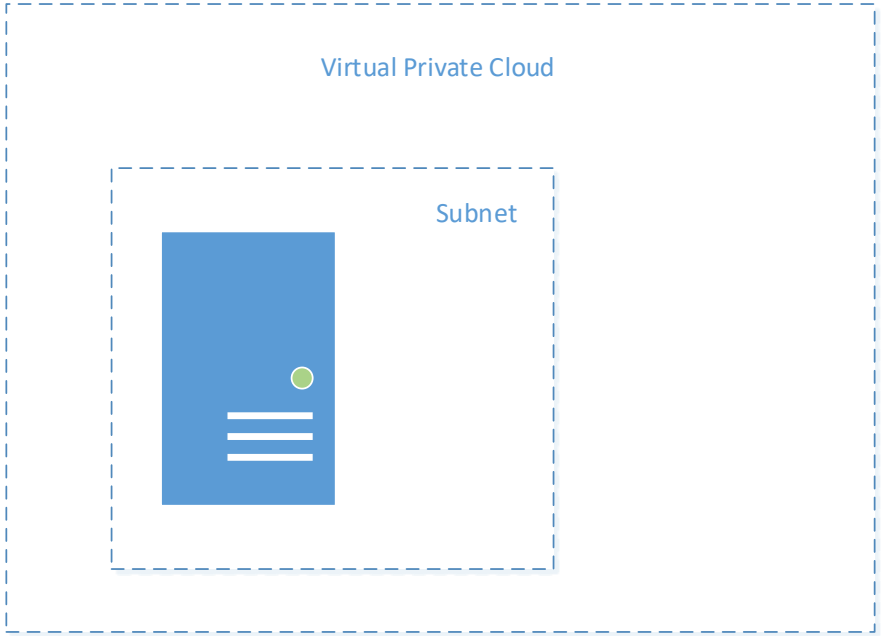
Instance: i-03547ffa9ba71ff11 (WebServer) Public IP: 54.246.158.127

Feedback English (US)

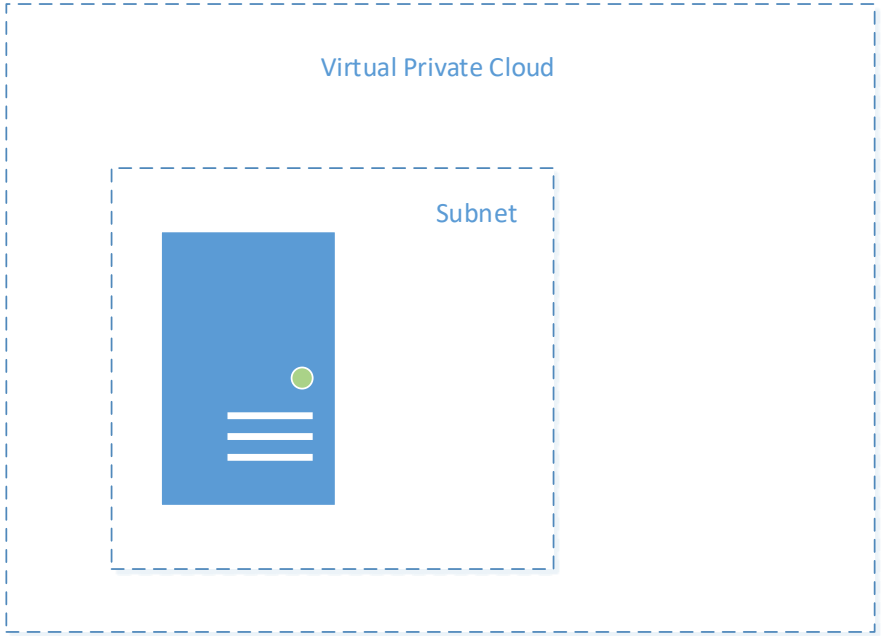
 Privacy Policy Terms of Use

Networks

Creating networks

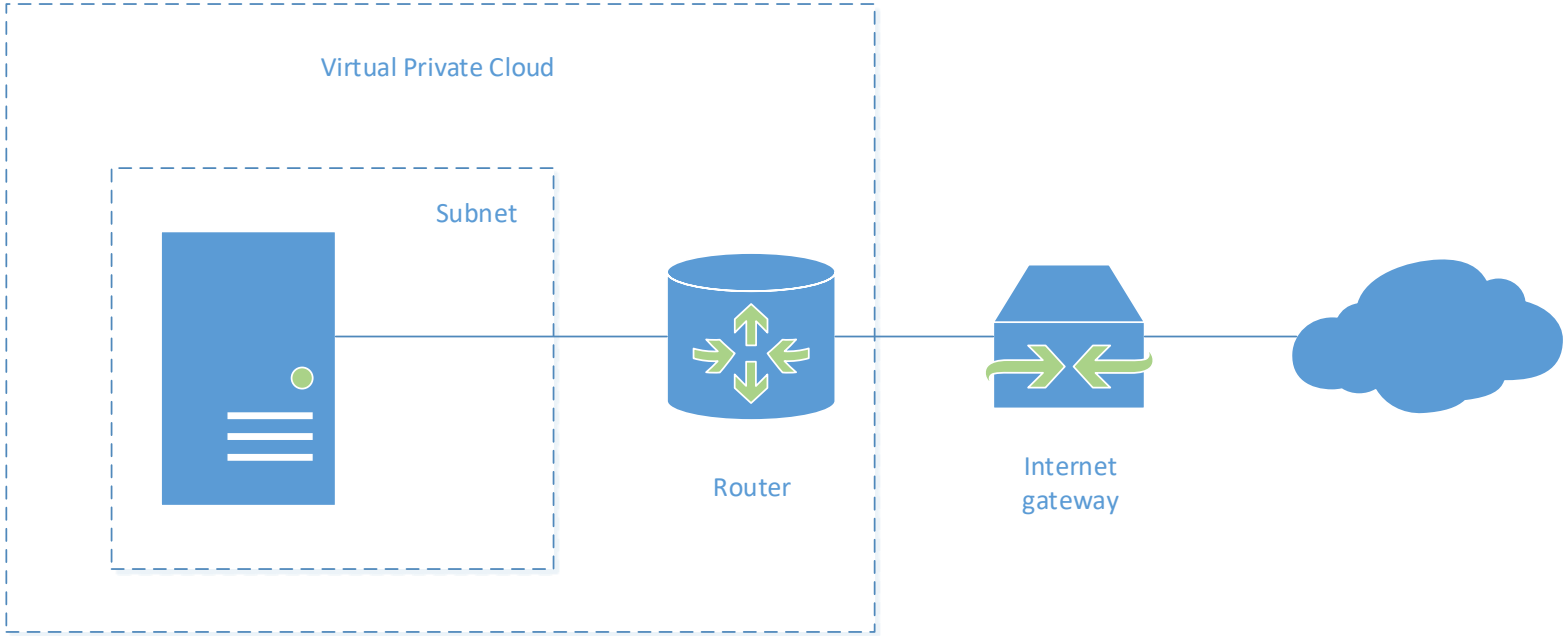


Creating networks



Internet gateway

Creating networks



VPC

- /network.tf

```
resource "aws_vpc" "my_product" {
 cidr_block = "10.0.0.0/16"

 tags {
  Name = "MyProduct.VPC"
 }
}
```

Gateway

- /network.tf

```
resource "aws_internet_gateway" "my_product" {
 vpc_id = "${aws_vpc.my_product.id}"

 tags {
  Name = "MyProduct.GW"
 }
}
```

Route

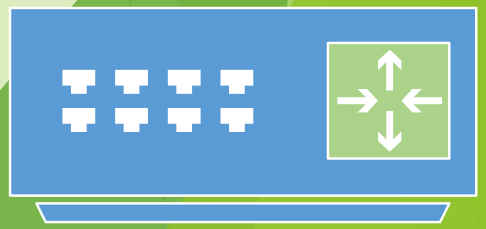
- /network.tf

```
resource "aws_route" "my_product" {
 route_table_id         = "${aws_vpc.my_product.main_route_table_id}"
 destination_cidr_block = "0.0.0.0/0"

 gateway_id = "${aws_internet_gateway.my_product.id}"
}
```

Subnet

- /network.tf

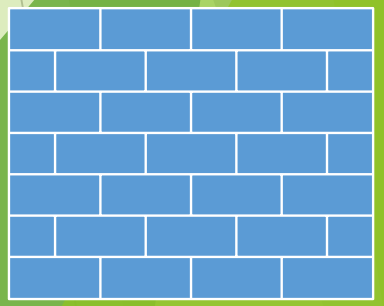
```
resource "aws_subnet" "my_product_a" {
 availability_zone = "eu-west-1a"

 vpc_id = "${aws_vpc.my_product.id}"
 cidr_block = "10.0.1.0/24"

 tags {
  Name = "MyProduct.Subnet.A"
 }
}
```



Security

- /security.tf

```
resource "aws_security_group" "my_product_from_office" {
 name = "my_product_from_office"
 description = "Allow traffic from the offices"
 vpc_id = "${aws_vpc.my_product.id}"

 ingress {
  protocol = "-1"
  from_port = 0
  to_port = 0

  cidr_blocks = ["12.34.56.78/32", "87.65.43.21/32"]
 }

 egress {
  protocol = "-1"
  from_port = 0
  to_port = 0
  cidr_blocks = ["0.0.0.0/0"]
 }
}
```



Attach changes to the instance

- /server.tf

```
resource "aws_instance" "web_server" {
 ami           = "${var.ami}"
 instance_type = "${var.instance_type}"
 subnet_id              = "${aws_subnet.my_product_a.id}"
 vpc_security_group_ids = [
  "${aws_security_group.my_product_from_office.id}"
 ]
 tags {
  Name = "WebServer"
 }
}
```

Applying changes

- /

```
terraform init
```

```
TF_VAR_aws_access_key=XXXXX \
TF_VAR_aws_secret_key=YYYYY \
terraform apply
```

aws Services Resource Groups

VPC Dashboard

Filter by VPC:

Select a VPC

Virtual Private Cloud

- Your VPCs
- Subnets
- Route Tables
- Internet Gateways
- Egress Only Internet Gateways
- DHCP Options Sets
- Elastic IPs
- Endpoints
- NAT Gateways
- Peering Connections

Security

- Network ACLs
- Security Groups

VPN Connections

- Customer Gateways
- Virtual Private Gateways
- VPN Connections

Create VPC | Actions

MyProduct.VPC

1 to 1 of 1 VPC

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP options set	Route table	Network ACL	Tenancy	Default VPC
MyProduct.VPC	vpc-05d7f262	available	10.0.0.0/16		dopt-8c6203e8	rtb-0ffe6f69	acl-a4669cc2	Default	No

vpc-05d7f262 | MyProduct.VPC

Summary | CIDR Blocks | Flow Logs | Tags

VPC ID:	vpc-05d7f262 | MyProduct.VPC	**Network ACL:**	acl-a4669cc2
State:	available	**Tenancy:**	Default
IPv4 CIDR:	10.0.0.0/16	**DNS resolution:**	yes
IPv6 CIDR:		**DNS hostnames:**	no
DHCP options set:	dopt-8c6203e8	**ClassicLink DNS Support:**	no
Route table:	rtb-0ffe6f69		

amazon web services™

Feedback English (US)

Privacy Policy Terms of Use

aws Services Resource Groups

VPC Dashboard

Filter by VPC:

Select a VPC

Virtual Private Cloud

- Your VPCs
- Subnets
- Route Tables
- Internet Gateways
- Egress Only Internet Gateways
- DHCP Options Sets
- Elastic IPs
- Endpoints
- NAT Gateways
- Peering Connections

Security

- Network ACLs
- Security Groups

VPN Connections

- Customer Gateways
- Virtual Private Gateways
- VPN Connections

Create Internet Gateway | Delete | Attach to VPC | Detach from VPC

MyProduct

1 to 1 of 1 Internet Gateway

Name	ID	State	VPC
MyProduct.GW	igw-8145a6e6	attached	vpc-05d7f262 | MyProduct.VPC

igw-8145a6e6 | MyProduct.GW

Summary | Tags

ID: igw-8145a6e6 | MyProduct.GW

State: attached

Attached VPC ID: vpc-05d7f262 | MyProduct.VPC

Attachment state: available

aws Services Resource Groups

VPC Dashboard

Filter by VPC:

Select a VPC

Virtual Private Cloud

- Your VPCs
- Subnets
- Route Tables
- Internet Gateways
- Egress Only Internet Gateways
- DHCP Options Sets
- Elastic IPs
- Endpoints
- NAT Gateways
- Peering Connections

Security

- Network ACLs
- Security Groups

VPN Connections

- Customer Gateways
- Virtual Private Gateways
- VPN Connections

Create Route Table | Delete Route Table | Set As Main Table

MyProduct

1 to 1 of 1 Route Table

Name	Route Table ID	Explicitly Associat	Main	VPC
	rtb-0ffe6f69	0 Subnets	Yes	vpc-05d7f262 | MyProduct.VPC

rtb-0ffe6f69

Summary | Routes | Subnet Associations | Route Propagation | Tags

Edit

View: All rules

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	igw-8145a6e6	Active	No

aws Services Resource Groups

VPC Dashboard

Filter by VPC:

Select a VPC

Virtual Private Cloud

- Your VPCs
- Subnets
- Route Tables
- Internet Gateways
- Egress Only Internet Gateways
- DHCP Options Sets
- Elastic IPs
- Endpoints
- NAT Gateways
- Peering Connections

Security

- Network ACLs
- Security Groups

VPN Connections

- Customer Gateways
- Virtual Private Gateways
- VPN Connections

Create Subnet | Subnet Actions

MyProduct

1 to 2 of 2 Subnets

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone	Route Table	Networ
MyProduct.Subnet.A	subnet-354fdf6e	available	vpc-05d7f262 | MyProduct.VPC	10.0.1.0/24	249		eu-west-1a	rtb-0ffe6f69	acl-a46
MyProduct.Subnet.B	subnet-df3760b8	available	vpc-05d7f262 | MyProduct.VPC	10.0.2.0/24	250		eu-west-1b	rtb-0ffe6f69	acl-a46

subnet-354fdf6e | MyProduct.Subnet.A

Summary | Route Table | Network ACL | Flow Logs | Tags

Edit

Route Table: rtb-0ffe6f69

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-8145a6e6

aws Services Resource Groups

VPC Dashboard

Filter by VPC:

Select a VPC

Virtual Private Cloud

- Your VPCs
- Subnets
- Route Tables
- Internet Gateways
- Egress Only Internet Gateways
- DHCP Options Sets
- Elastic IPs
- Endpoints
- NAT Gateways
- Peering Connections

Security

- Network ACLs
- Security Groups

VPN Connections

- Customer Gateways
- Virtual Private Gateways
- VPN Connections

Create Security Group | Security Group Actions

Filter All security groups | my_product_from_office | 1 to 1 of 1 Security Group

Name tag	Group ID	Group Name	VPC	Description
	sg-42ad6f39	my_product_from_offi...	vpc-05d7f262 | MyProduct.V...	Allow traffic from the offices

sg-42ad6f39

Summary | Inbound Rules | Outbound Rules | Tags

Edit

Type	Protocol	Port Range	Source	Description
ALL Traffic	ALL	ALL	83.209.8.88/32	

aws Services Resource Groups

- EC2 Dashboard
- Events
- Tags
- Reports
- Limits

INSTANCES
- Instances
- Spot Requests
- Reserved Instances
- Scheduled Instances
- Dedicated Hosts

IMAGES
- AMIs
- Bundle Tasks

ELASTIC BLOCK STORE
- Volumes
- Snapshots

NETWORK & SECURITY
- Security Groups
- Elastic IPs
- Placement Groups
- Key Pairs
- Network Interfaces

LOAD BALANCING
- Load Balancers
- Target Groups

AUTO SCALING
- Launch Configurations

Launch Instance | Connect | Actions

search : WebServer | Add filter | 1 to 1 of 1

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP
WebServer	i-03547ffa9ba71ff11	t2.micro	eu-west-1a	running	2/2 checks ...	None		54.246.158.127

Instance: i-03547ffa9ba71ff11 (WebServer) **Public IP: 54.246.158.127**

Description | Status Checks | Monitoring | Tags

Instance ID	i-03547ffa9ba71ff11
Instance state	running
Instance type	t2.micro
Elastic IPs	
Availability zone	eu-west-1a
Security groups	my_product_from_office. view inbound rules
Scheduled events	No scheduled events
AMI ID	ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20170721 (ami-785db401)
Platform	-
IAM role	-
Key pair name	-
Owner	245629883278
Launch time	October 29, 2017 at 6:10:37 PM UTC+1 (less than one hour)
Termination protection	False
Lifecycle	normal
Monitoring	basic
Alarm status	None

Public DNS (IPv4)	-
IPv4 Public IP	54.246.158.127
IPv6 IPs	-
Private DNS	ip-10-0-1-109.eu-west-1.compute.internal
Private IPs	10.0.1.109
Secondary private IPs	
VPC ID	vpc-05d7f262
Subnet ID	subnet-354fdf6e
Network interfaces	eth0
Source/dest. check	True
EBS-optimized	False
Root device type	ebs
Root device	/dev/sda1
Block devices	/dev/sda1
Elastic GPU	-
Elastic GPU type	-

DNS

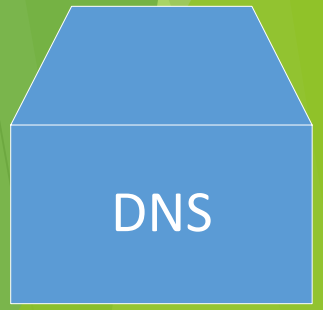
DNS

- /dns.tf

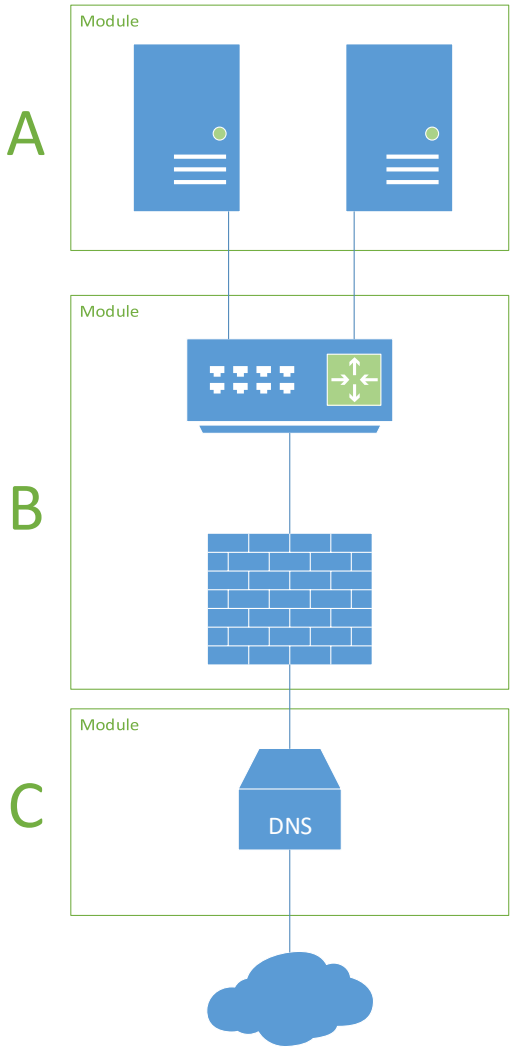
```
variable "environment_name" {}
```

```
resource "aws_route53_zone" "my_product" {
 name = "${var.environment_name}.my-product.com"
}
```

```
resource "aws_route53_record" "my_product_a" {
 zone_id = "${aws_route53_zone.my_product.id}"
 name = "${var.environment_name}-my-product-a"
 type = "A"
 ttl = "300"
 records = ["${aws_instance.my_product.public_ip}"]
}
```

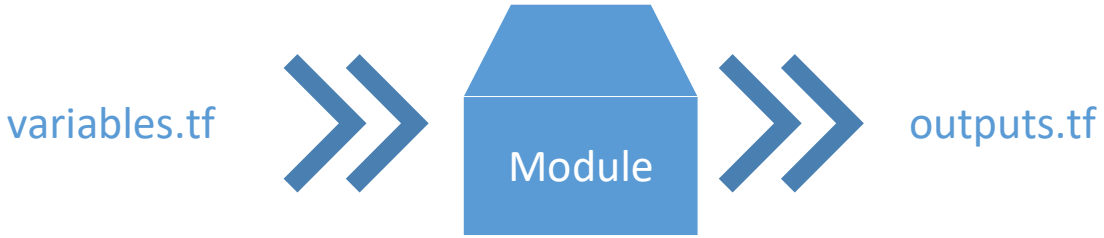


Modules



Modules

- Folder with name
- Files
  - main.tf
  - variables.tf
  - outputs.tf



Modules

- Folder with name
- Files
  - main.tf
  - variables.tf
  - outputs.tf

- /modules/network

- /modules/network/network.tf
- /modules/network/security.tf

- /modules/network/variables.tf
- /modules/network/outputs.tf

Instance module

- /modules/network/variables.tf

```
variable "aws_default_region" {}
variable "name" {}

# Network
variable "vpc_cidr" {}
variable "subnet_cidr" {}

# Security
variable "office_cidr" {}
```

Instance module

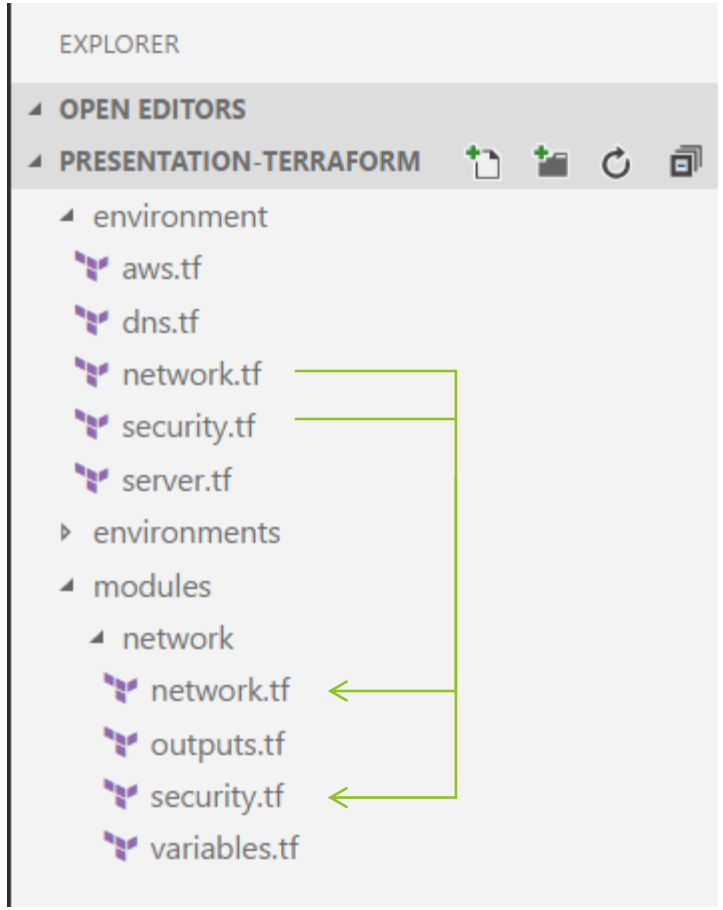
- /modules/network/outputs.tf

```
output "subnet_id" { value = "${aws_subnet.my_product_a.id}" }
```

```
output "security_group_id" {
  value = "${aws_security_group.my_product_from_office.id}"
}
```

Instance module

- /instance/network/*



Module usage

Network module

- /instances/dev/main.tf

```
module "my_product_network" {

  source              = "../../modules/network"

  aws_default_region = "${var.aws_default_region}"
  name                = "MyProduct"

  vpc_cidr            = "10.0.0.0/16"
  subnet_cidr         = "10.0.1.0/24"

  office_cidr         = "12.34.56.78/32"

}
```

Network module

- /instances/dev/main.tf

```
module "my_product_network" {

  source              = "git::ssh://example.com/modules.git?ref=v1.0.0"

  aws_default_region = "${var.aws_default_region}"
  name                = "MyProduct"

  vpc_cidr            = "10.0.0.0/16"
  subnet_cidr         = "10.0.1.0/24"

  office_cidr         = "12.34.56.78/32"

}
```

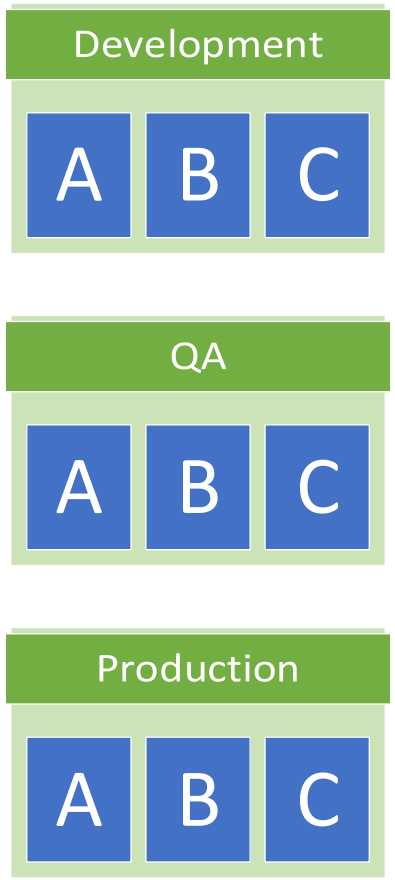
Network module

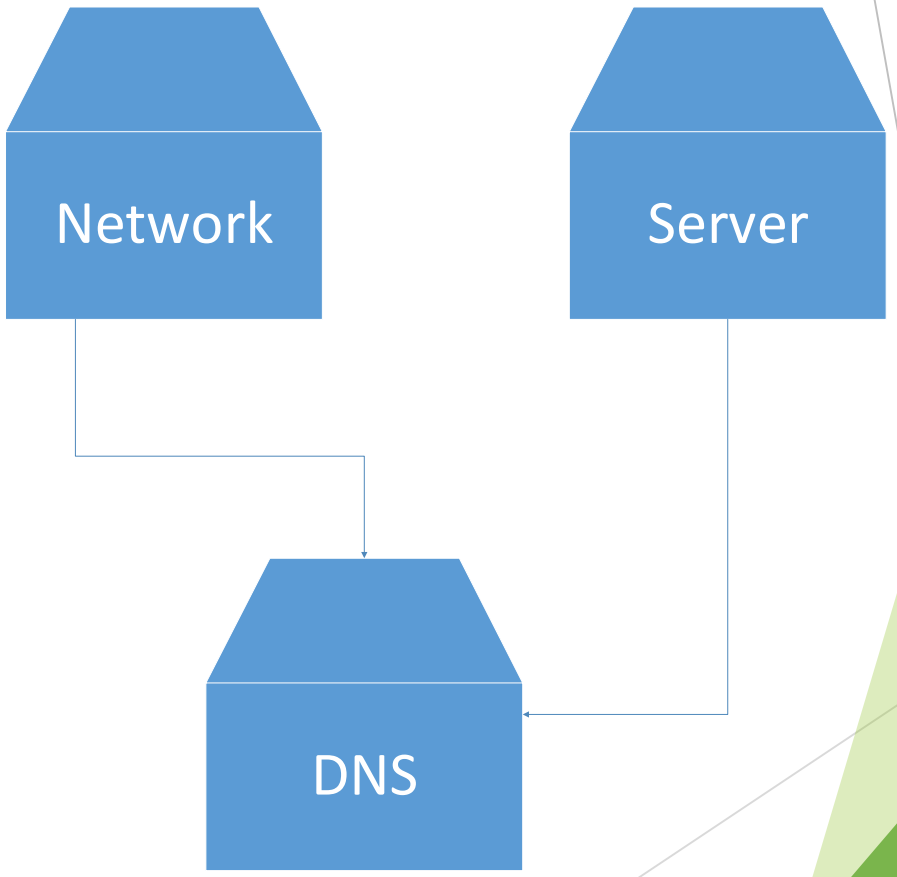
- /instances/dev/main.tf

```
resource "aws_instance" "my_product" {
 # ...

 subnet_id = "${module.my_product_network.subnet_id}“

 vpc_security_group_ids = [
  "${module.my_product_network.security_group_id}"
 ]

 # ...
}
```

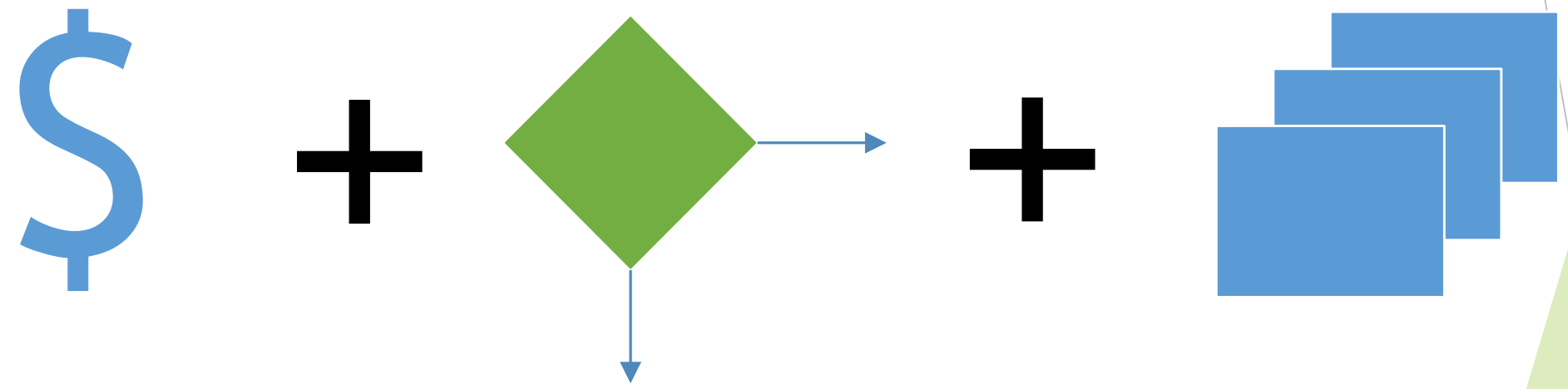
Environments

Environments, Folder Alternative





Environments, Count Alternative



variable

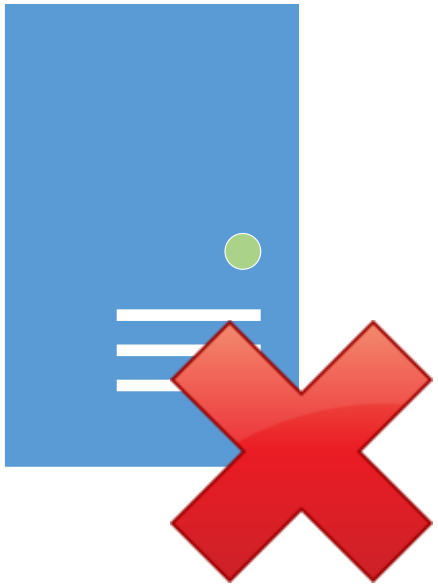
conditions

count

Environments, Count Alternative

- /instances/server.tf

```
resource "aws_instance" "my_product" {
 count = "0"
 # ...
}
```



Environments, Count Alternative

- /instances/server.tf

```
resource "aws_instance" "my_product" {
 count = "1"
 # ...
}
```

Environments, Count Alternative

- /instances/server.tf

```
resource "aws_instance" "my_product" {
 count = "${ var.environment == "dev" ? 0 : 1 }"
 # ...
}
```

Conclusions

- Recreate infrastructure from code
- Add changes to the infra and have them version controlled
  - Pull Request
  - Collaboration
  - TAGS or stable branches

Conclusions

- Hundred of providers
  - Not only infrastructure related
- Do not need to learn different APIs
- Interconnection between them

Environments

- Set up quickly environments
- Destroy them with a single command
  - Not explained, but ask me!
- Disposable environments
- Create an execution plan without apply your changes

Conclusions

Load Balance

Databases

Instances

Network

Conclusions

Infra Team

Miguel Á. Domínguez Coloma

Infrastructure Engineer	dooer
Owner/Consultant	eridem ab
Blog	http://eridem.net
Email	m@eridem.net
LinkedIn	https://linkedin.com/in/eridem