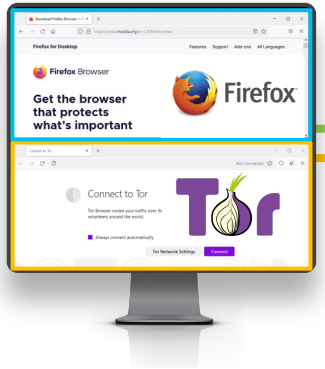
Jedi erő a gyakorlatban – IT biztonsági automatizálási esetek

HBONE Workshop 2023

Ács György, Security Technical Solution Architect
https://github.com/Gyuri1

2023. november 15.

Encrypted Visibility Engine uses TLS Fingerprinting



TLS ClientHello

```
Cipher Suites (18 suites)
    Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
    Cipher Suite: TLS_CHACHA20_POLY1305_SHA256 (0x1303)
    Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca9)
    Cipher Suite: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca8)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
    Cipher Suite: TLS_DHE_RSA_WITH_AES_128_CBC_SHA (0x0033)
    Cipher Suite: TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)
```

TCP/TLS 192.168.2.110/34624->172.16.45.200/443

TCP/TLS 192.168.2.110/21013->203.0.113.154/443

FTD

TLS ClientHello

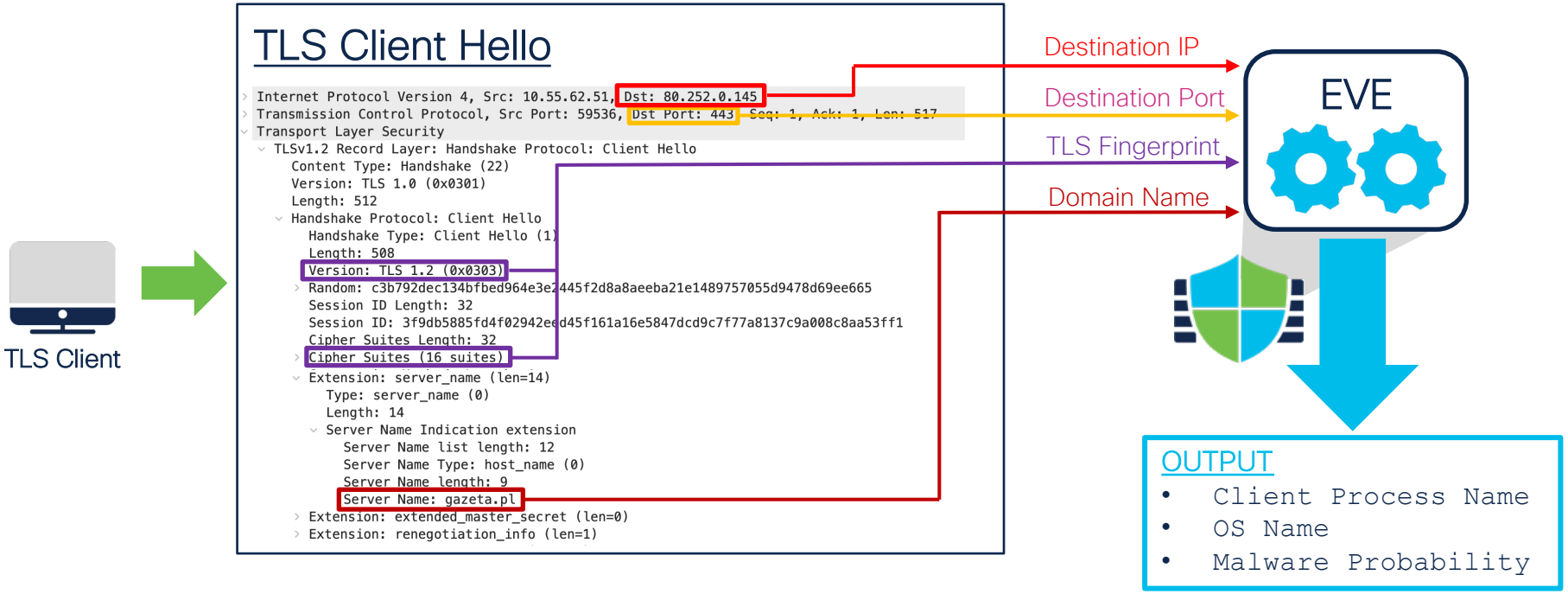
```
Cipher Suites (19 suites)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (0xc024)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (0xc023)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xc027)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
    Cipher Suite: TLS_RSA_WITH_AES_256_GCM_SHA384 (0x009d)
    Cipher Suite: TLS_RSA_WITH_AES_128_GCM_SHA256 (0x009c)
```

Confidence: 99.94%
Process: firefox.exe
Version: 76.0.1
Category: browser
OS: Windows 10 19041.329
Typical FQDN: cisco.com

Generate unique fingerprints for client applications based on TLS, TCP and other clear text fields to use for context enrichment

Confidence: 100%
Process: tor.exe
Version: 9.0.2
Category: anonymizer
OS: Windows 10 19041.329
Typical FQDN: nsksdlkoup.me

Fingerprinting Analysis at the Firewall



Policy Editor | Secure Firewall Manag

Not secure | https://fmc7.cisco.com/ddd/#FirewallPolicyEditor;uuid=005056A9-C859-0ed3-0000-008589934595;type=PG.FIREWALL.N...

Firewall Management Center
Policies / Access Control / Policy Editor

Overview | Analysis | Policies | Devices | Objects | Integration | Deploy | admin | SECURE

EVE Demo ACP

Enter Description

Try New UI Layout | Analyze Hit Counts | Save | Cancel

Inheritance Settings | Policy Assignments (1)

Rules | Security Intelligence | HTTP Responses | Logging | Advanced

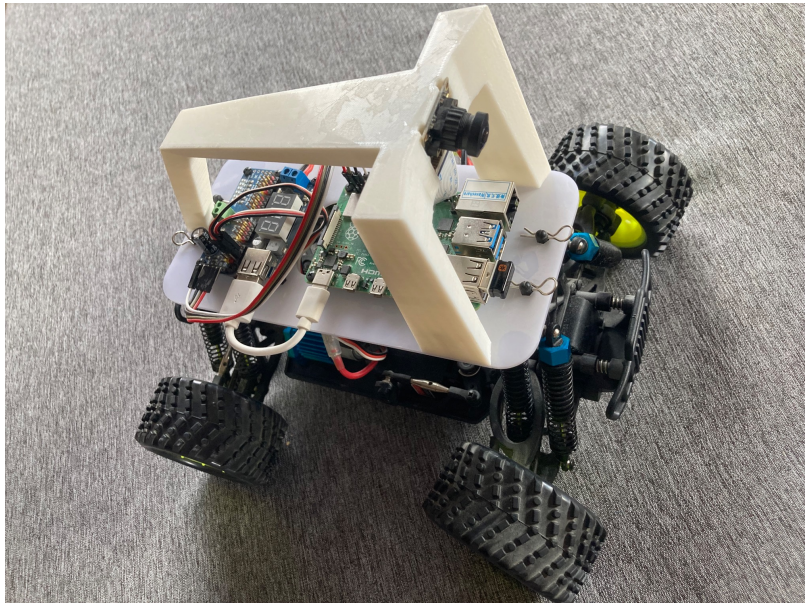
Prefilter Policy: Default Prefilter Policy | SSL Policy: Test SSL | Identity Policy: None

Filter by Device | Search Rules | Show Rule Conflicts | + Add Category | + Add Rule

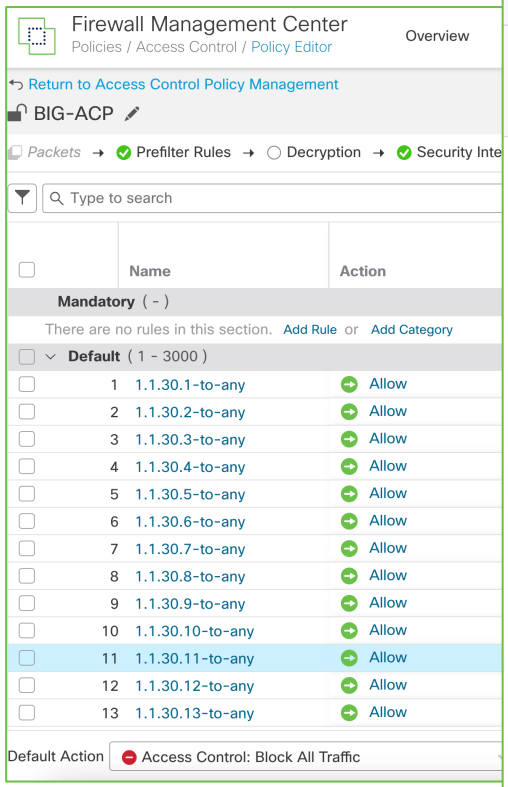
#	Name	Source Zones	Dest Zones	Source Networks	Dest Networks	Users	Source Ports	Dest Ports	Source Dynamic Attributes	Destinati... Dynamic Attributes	Action
Mandatory - Top-Level (-)											
Mandatory - EVE Demo ACP (1-6)											
1	Block QUIC	Inside-400	Outside-DMZ	Any	Any	Any	Any	Any	Any	Any	Block w
2	Block DNS over HTTF	Inside-400	Outside-DMZ	Any	Any	Any	Any	Any	Any	Any	Block w
3	Wget outbound	Inside-400	Outside-DMZ	Any	Any	Any	Any	Any	Any	Any	Allow
4	Firefox Outbound (Disabled)	Inside-400	Outside-DMZ	Any	Any	Any	Any	Any	Any	Any	Allow
5	Block some apps (Disabled)	Inside-400	Outside-DMZ	Any	Any	Any	Any	Any	Any	Any	Block w
6	Allow Outbound	Inside-400	Outside-DMZ	Any	Any	Any	Any	Any	Any	Any	Allow
Default - EVE Demo ACP (-)											

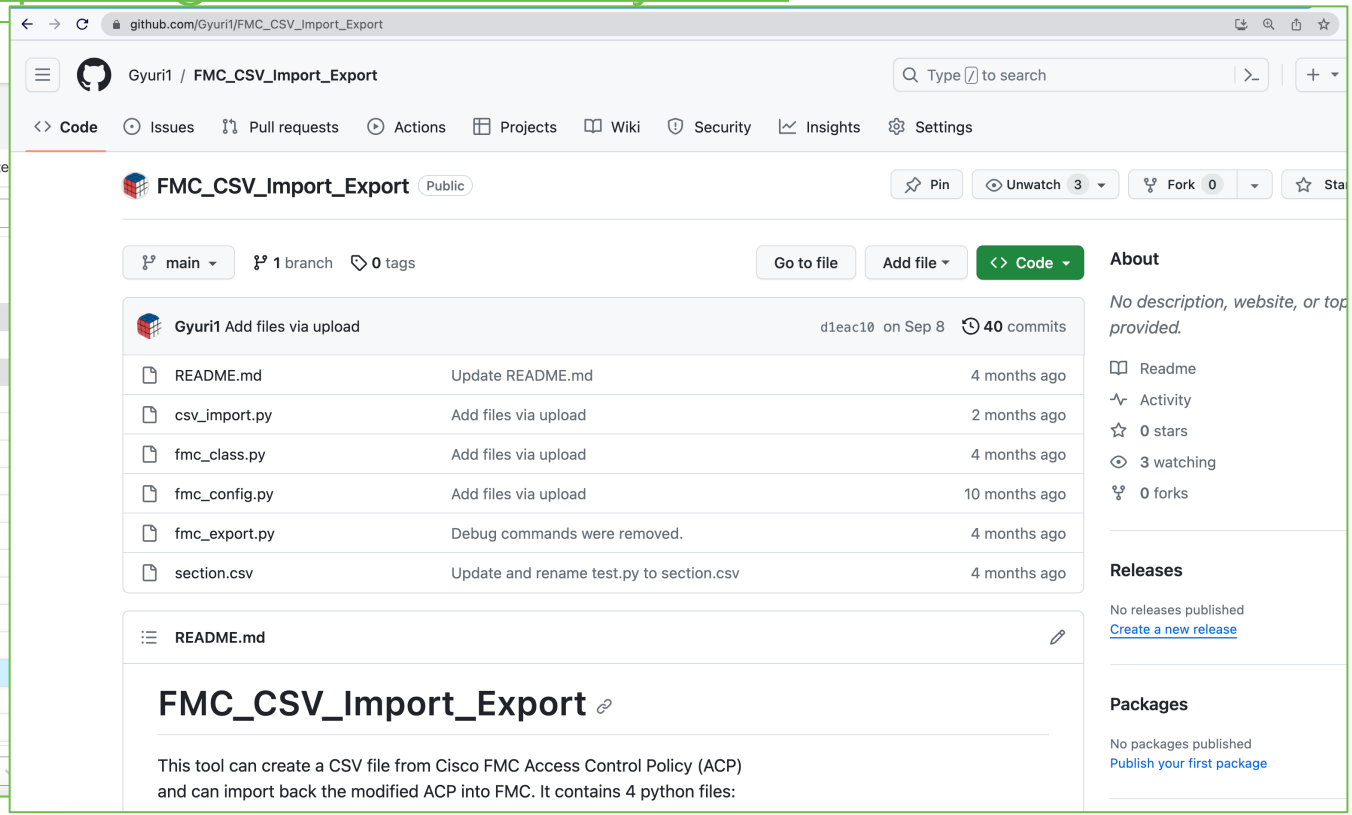
There are no rules in this section. Add Rule or Add Category

Displaying 1 - 6 of 6 rules | Page 1 of 1 | Rules per page: 100



Motiváció – 1: Nagy számú policy konfiguráció kezelése, https://github.com/Gyuri1

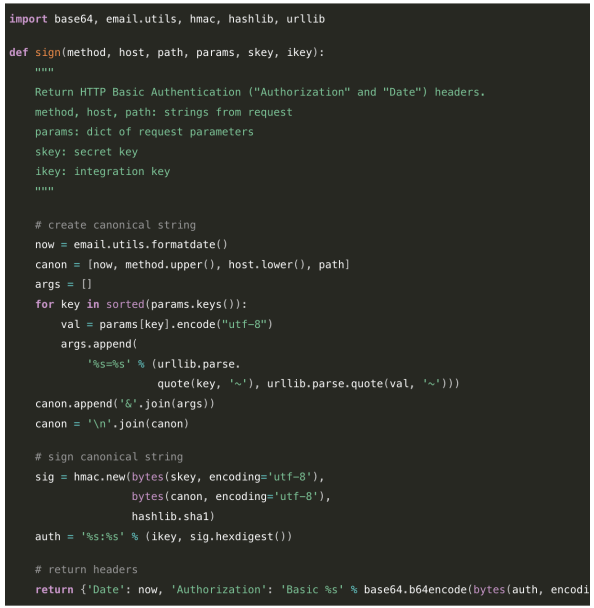




Motiváció – 2: A duo_client “SDK” története

- https://duo.com/docs/adminapi

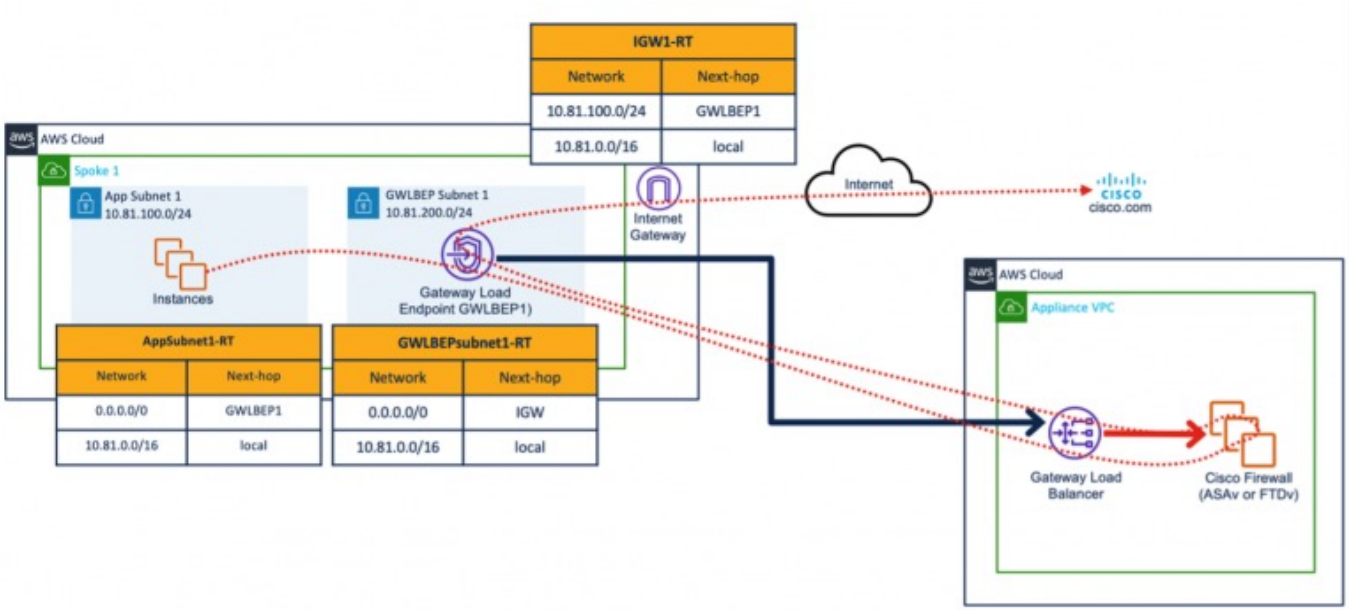
https://github.com/duosecurity/duo_client_python/blob/master/examples/create_user_and_phone.py



```python
import base64, email.utils, hmac, hashlib, urllib

def sign(method, host, path, params, skey, ikey):
    """
    Return HTTP Basic Authentication ("Authorization" and "Date") headers.
    method, host, path: strings from request
    params: dict of request parameters
    skey: secret key
    ikey: integration key
    """

    # create canonical string
    now = email.utils.formatdate()
    canon = [now, method.upper(), host.lower(), path]
    args = []
    for key in sorted(params.keys()):
        val = params[key].encode("utf-8")
        args.append(
            '%s=%s' % (urllib.parse.
                       quote(key, '~'), urllib.parse.quote(val, '~')))
    canon.append('&'.join(args))
    canon = '\n'.join(canon)

    # sign canonical string
    sig = hmac.new(bytes(skey, encoding='utf-8'),
                   bytes(canon, encoding='utf-8'),
                   hashlib.sha1)
    auth = '%s:%s' % (ikey, sig.hexdigest())

    # return headers
    return {'Date': now, 'Authorization': 'Basic %s' % base64.b64encode(bytes(auth, encodi
```

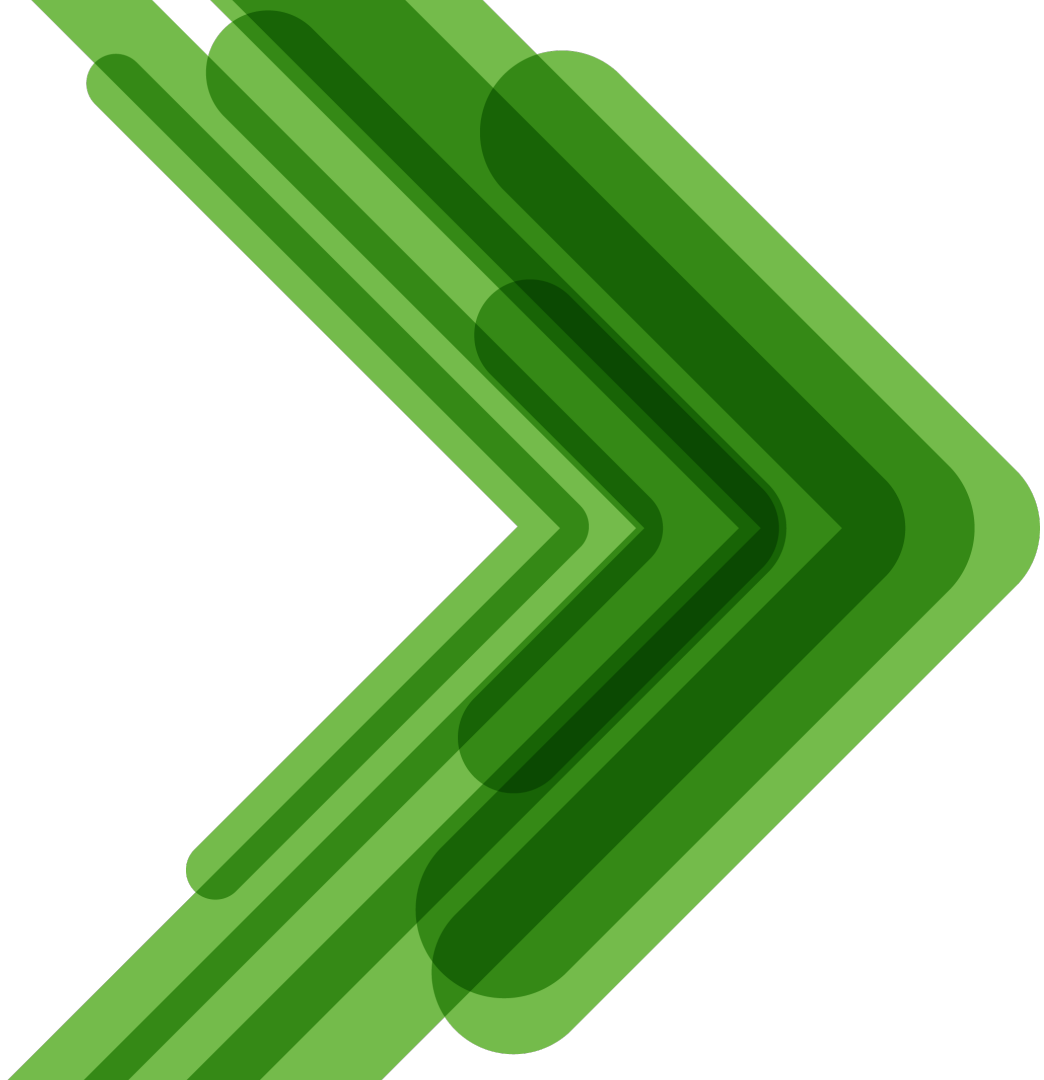
```python
# Create and return a new user object.
user = admin_api.add_user(
    username=USERNAME,
    realname=REALNAME,
)
print('Created user:')
pprint.pprint(user)
```

Motiváció – 3: felhős rendszerek telepítése és törlése(!)



- https://blogs.cisco.com/security/building-a-scalable-security-architecture-on-aws-with-cisco-secure-firewall-and-aws-transit-gateway

Demo



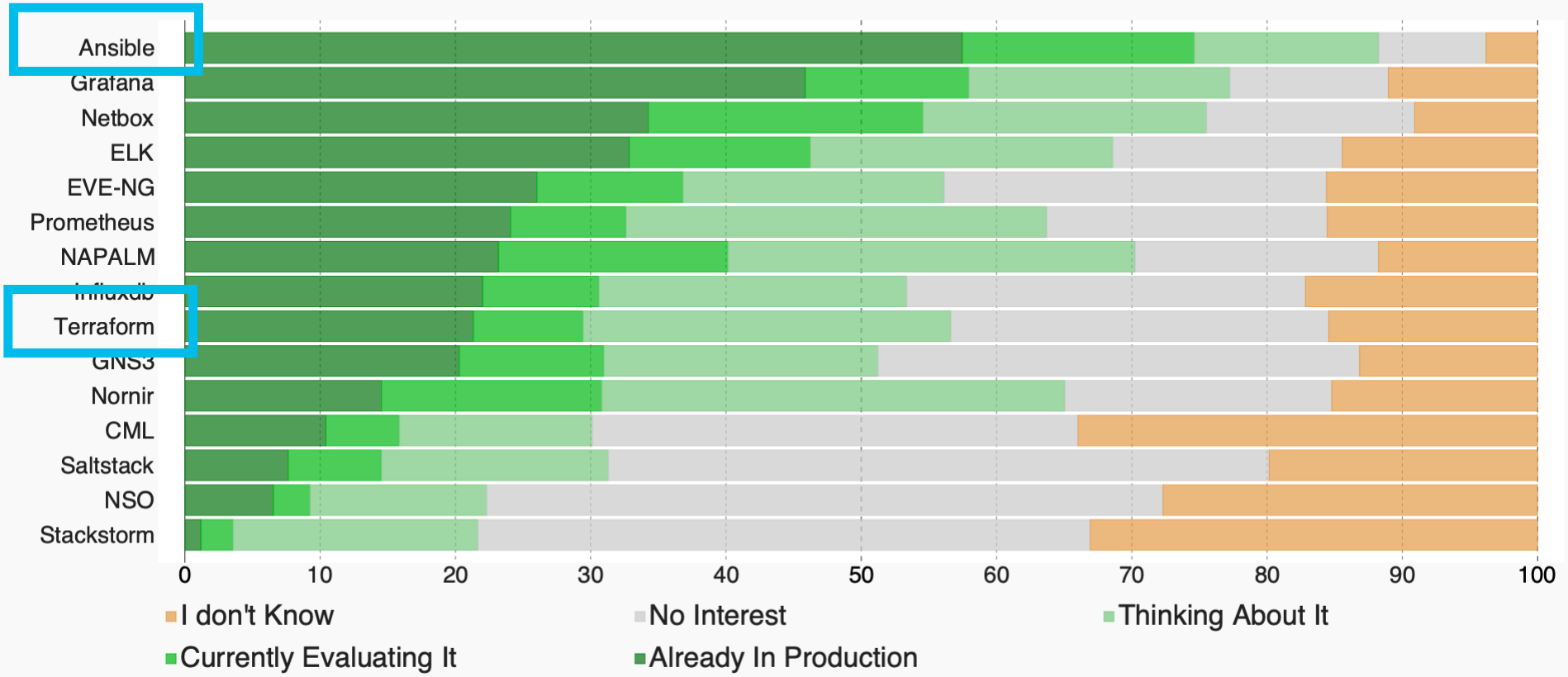
```
Rule name: 1.1.1.94-to-any
Rule name: 1.1.1.95-to-any
Rule name: 1.1.1.96-to-any
Rule name: 1.1.1.97-to-any
Rule name: 1.1.1.98-to-any
Rule name: 1.1.1.99-to-any
Rule name: 1.1.1.100-to-any
--------------------------------------------------------------------------------
CSV Rule file created : BULK-ACP.csv
Policy file created: BULK-ACP_policy.json
Object file created: BULK-ACP_objects.csv
>
>
> ls -las
total 464
  0 drwxr-xr-x  11 gacs  staff     352 Nov 12 23:31 .
  0 drwxr-xr-x  17 gacs  staff     544 Nov 12 23:28 ..
272 -rw-r--r--   1 gacs  staff  116355 Nov 12 23:31 BULK-ACP.csv
  8 -rw-r--r--   1 gacs  staff      76 Nov 12 23:31 BULK-ACP_objects.csv
  8 -rw-r--r--   1 gacs  staff    1786 Nov 12 23:31 BULK-ACP_policy.json
  8 -rw-rw-r--@  1 gacs  staff    1972 Sep  8 18:07 README.md
 64 -rw-rw-r--@  1 gacs  staff   29086 Sep  8 18:07 csv_import.py
 40 -rw-rw-r--@  1 gacs  staff   17079 Sep  8 18:07 fmc_class.py
  8 -rw-rw-r--@  1 gacs  staff      61 Nov 12 23:31 fmc_config.py
 48 -rw-rw-r--@  1 gacs  staff   23532 Sep  8 18:07 fmc_export.py
  8 -rw-rw-r--@  1 gacs  staff     858 Sep  8 18:07 section.csv
>
```

Ansible vs Terraform

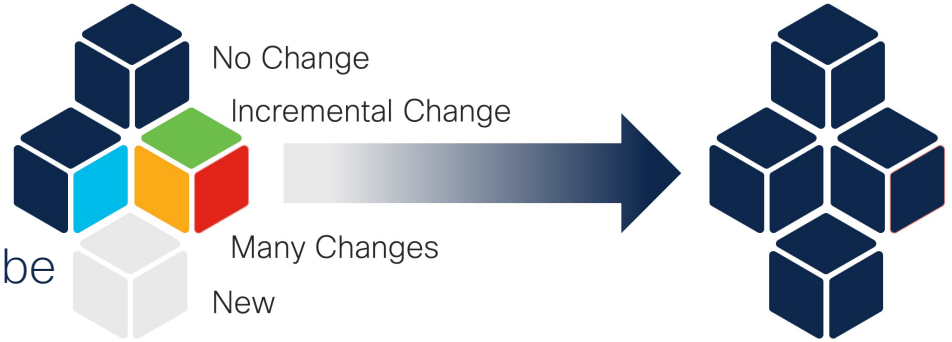
Networks Are Complex – Like Clouds

- Imagine a day in the life of a cloud operator:
  - Thousands to millions of servers
  - Multiple operating systems
  - High deployment velocity
  - Human error
  - Lack of team scale
- Enterprise networks are the same, just smaller
  - Daily repetitive tasks
  - NetDevOps and SecDevOps are replacing NetOps and SecOps

NetDevOps Survey (2020)
In the following list of tool, which one are you interested about or have currently implemented ?



5 Key Terms



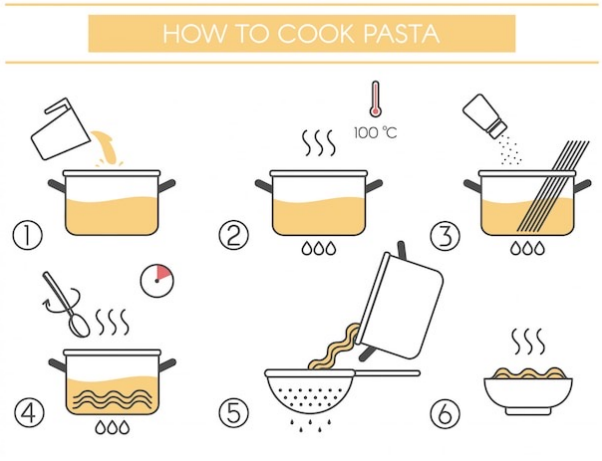
- **Declarative Configuration:** You describe what you want the end state to be
- **Idempotency**: repeated execution yields predictably consistent results
- **State**: a description of the current landscape
- **Stateful Configuration**: current landscape is considered before execution
- **Stateless Configuration**: execute without analyzing the current landscape

Idempotency:
from the Latin:
idem + *potence* (same + power)

Imperative vs Declarative

Imperative

1. Do this
2. Then do that
3. And finally do this also



Declarative

- I want this end-state. You figure it and let me know when it's in place.

JSON (JavaScript Object Notation) vs YAML (yet another markup language)

```json
[
  {
    "hosts": "all",
    "connection": "httpapi",
    "tasks": [
      {
        "name": "Get Domain UUID",
        "cisco.fmcansible.fmc_configuration": {
          "operation": "getAllDomain",
          "register_as": "domain"
        }
      },
      {
        "name": "Create a network object for Cisco FTD 1",
        "cisco.fmcansible.fmc_configuration": {
          "operation": "createMultipleNetworkObject",
          "data": {
            "name": "ansibleNet1",
            "value": "10.10.30.0/24",
            "type": "Network"
          },
          "path_params": {
            "domainUUID": "{{ domain[0].uuid }}"
          }
        }
      }
    ]
  }
]
```

```yaml
- hosts: all
  connection: httpapi
  tasks:
    - name: Get Domain UUID
      cisco.fmcansible.fmc_configuration:
        operation: getAllDomain
        register_as: domain

    - name: Create a network object for Cisco FTD 1
      cisco.fmcansible.fmc_configuration:
        operation: createMultipleNetworkObject
        data:
          name: ansibleNet1
          value: 10.10.30.0/24
          type: Network
        path_params:
          domainUUID: '{{ domain[0].uuid }}'
```

- YAML has COMMENT option

https://onlineyamltools.com/convert-yaml-to-json

Similarities between Terraform and Ansible

- Infrastructure configuration tool
- Run from the command line
- Can be run from Docker containers or in CI/CD pipelines
- Connect to any configuration interface
- Declarative configuration syntax
- Configuration idempotency
- Open Source, commercial offering

Red Hat Ansible

- Entirely stateless execution
- Uses YAML for configuration
- Can be executed programmatically via Python
- Extended by Modules written in Python

```
[edge]
cat8kv ansible_host=18.188.19.239
```

```
---
- name: Cloud Edge Router
  hosts: cat8kv
  gather_facts: False
  tasks:
    - name: PING!
      ping:
```

Hashicorp Terraform

- Entirely stateful execution
- Uses HashiCorp Configuration Language (HCL)
- Programmatic execution via CDKTFs (Cloud Development Kit for Terraform) in multiple languages (including Python)
- Extended by providers written in Golang

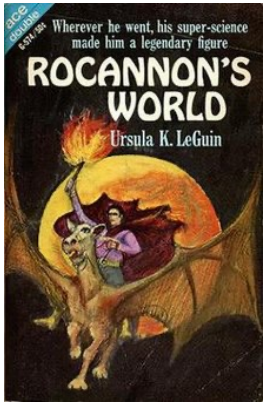
```
##############################################
# Terraform Configuration
##############################################
terraform {
  required_version = "~> 1.2.8"
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 4.31.0"
    }
  }
  backend "s3" {
    bucket  = "hwa-terraform"
    key     = "clus-2023/terraform.tfstate"
    region  = "us-east-2"
    encrypt = true
  }
}
provider "aws" {
  region = var.aws_region
}
```

Now let’s Ansible!

Why choose Ansible?

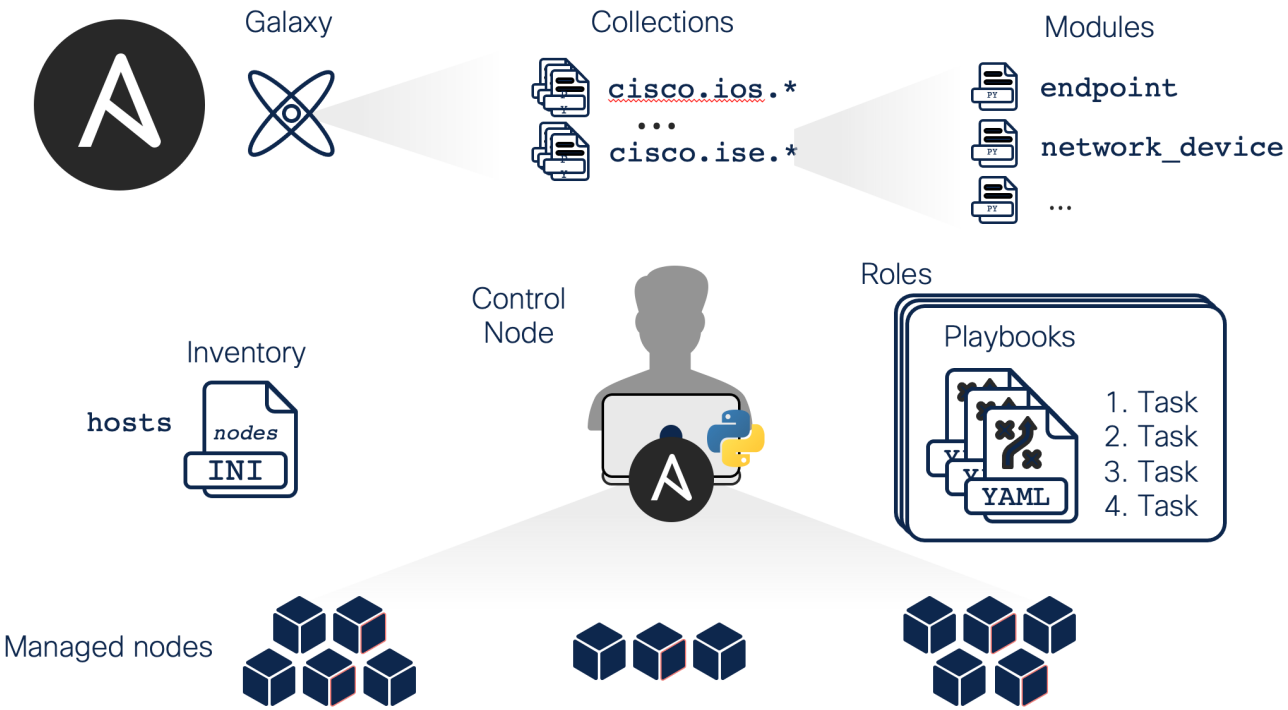
- Quick Learning Curve
- Agentless, Modular
- Supports Every Device from Every Vendor
- Open Source Community and Vendor Commercial Support
- Common Toolset with server / DevOps / infra teams

Ansible originated in a sci-fi novel as a contraction of "answerable" for a device allow its users to receive answers to their messages in a reasonable amount of time, even over interstellar distances. The word "ansible" for a **faster-than-light communicator**.

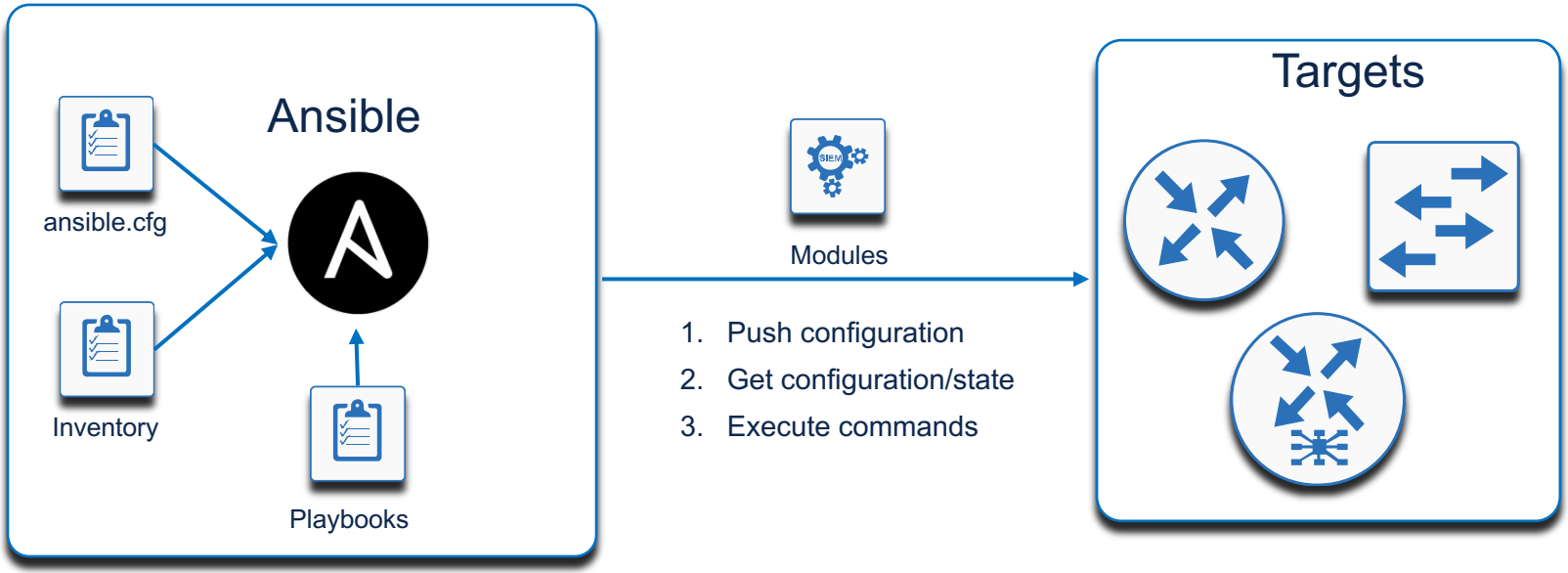


Simple	Flexible	Agentless
• Human-readable • Declarative configs • Ordered tasks • No coding required • Start small and scale	• Config management • Workstations • Servers / containers • Applications • Networks	• SSH (Linux, maxOS) • REST (Cisco ISE, Secure Firewall, …) • WinRM (Windows) • Others as needed • Efficient • Secure

Ansible Framework



Ansible in a nutshell



What Can I Automate Out of the Box?

- Cisco Devices
  - IOS / IOS XE / IOS XR / NX-OS
  - Meraki
  - ACI
  - ASA, FTD
- Other networking vendors
- NETCONF
- Generic network modules
- NSO
- Make your own – its open source

- RHEL
- Ubuntu
- Openstack
- Containers
- Files
- Package Managers

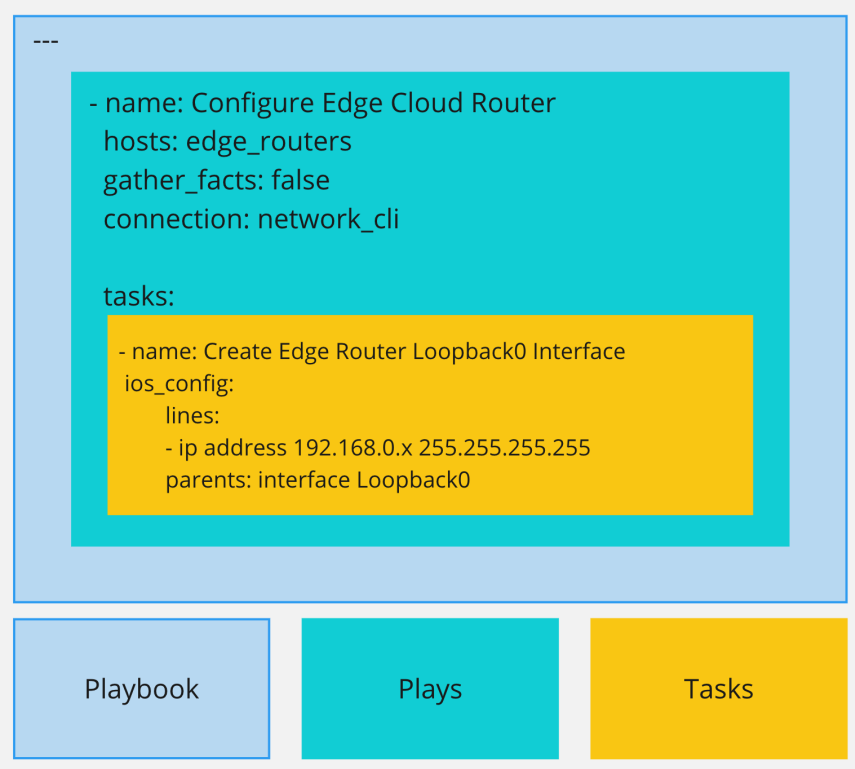
YAML

- Structure to define:
- dictionary (unordered set of key value pairs, lists)
- list of items
- key value pair

```
---
- name: Cloud Edge Router
  hosts: cat8kv
  gather_facts: False
  tasks:
    - name: PING!
      ping:
```

Anatomy of an Ansible Playbook

- Playbook
  - Contains other playbooks, roles, and/or tasks and is ran when you want to ”do something”.
- Plays
  - Maps, at minimum, a set of selected hosts and the tasks which should run on these hosts
- Tasks
  - The smallest unit of action that you wish to automate using an Ansible playbook.



Steps

- 1. Install Ansible
  - https://medium.com/javarevisited/how-to-install-ansible-on-mac-2baf00d42466
  - https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html
- 2. Install Ansible module
  - `ansible-galaxy collection install` **`cisco.fmcansible`**
  - https://github.com/CiscoDevNet/FMCAnsible
- 3. Create the inventory file
  - The default location for inventory is /etc/ansible/hosts, but you can specify a different path by adding the -i <path> argument to the ansible-playbook command.

Steps – cont.

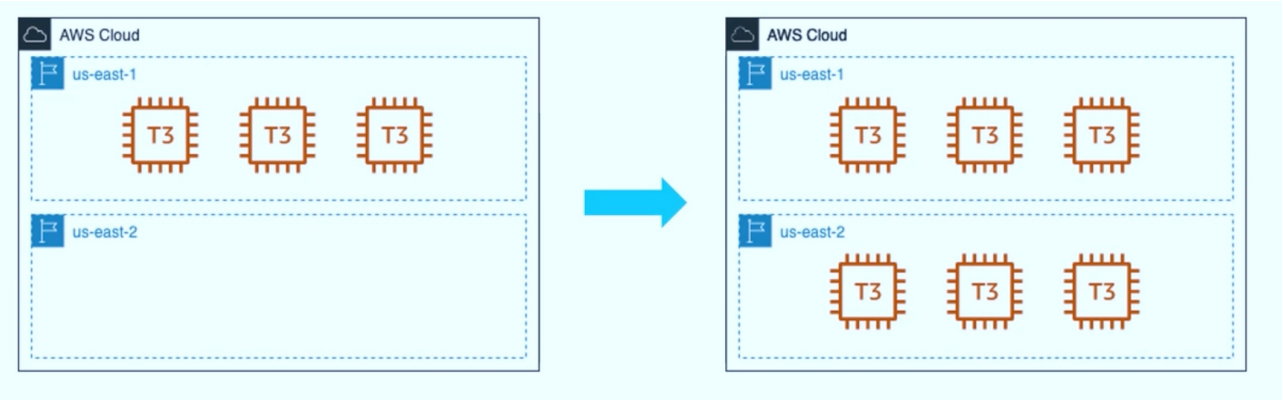
- 4. create a playbook `network.yml`
- 5. run it: `ansible-playbook -i hosts.txt network.yml -vvv`

network.yml

```yaml
- hosts: all
  connection: httpapi
  tasks:
    - name: Get Domain UUID
      cisco.fmcansible.fmc_configuration:
        operation: getAllDomain
        register_as: domain

    - name: Create a network object for Cisco FTD 1
      cisco.fmcansible.fmc_configuration:
        operation: createMultipleNetworkObject
        data:
          name: ansibleNet1
          value: 10.10.30.0/24
          type: Network
        path_params:
          domainUUID: '{{ domain[0].uuid }}'
```

Ansible – Stateless Change Management

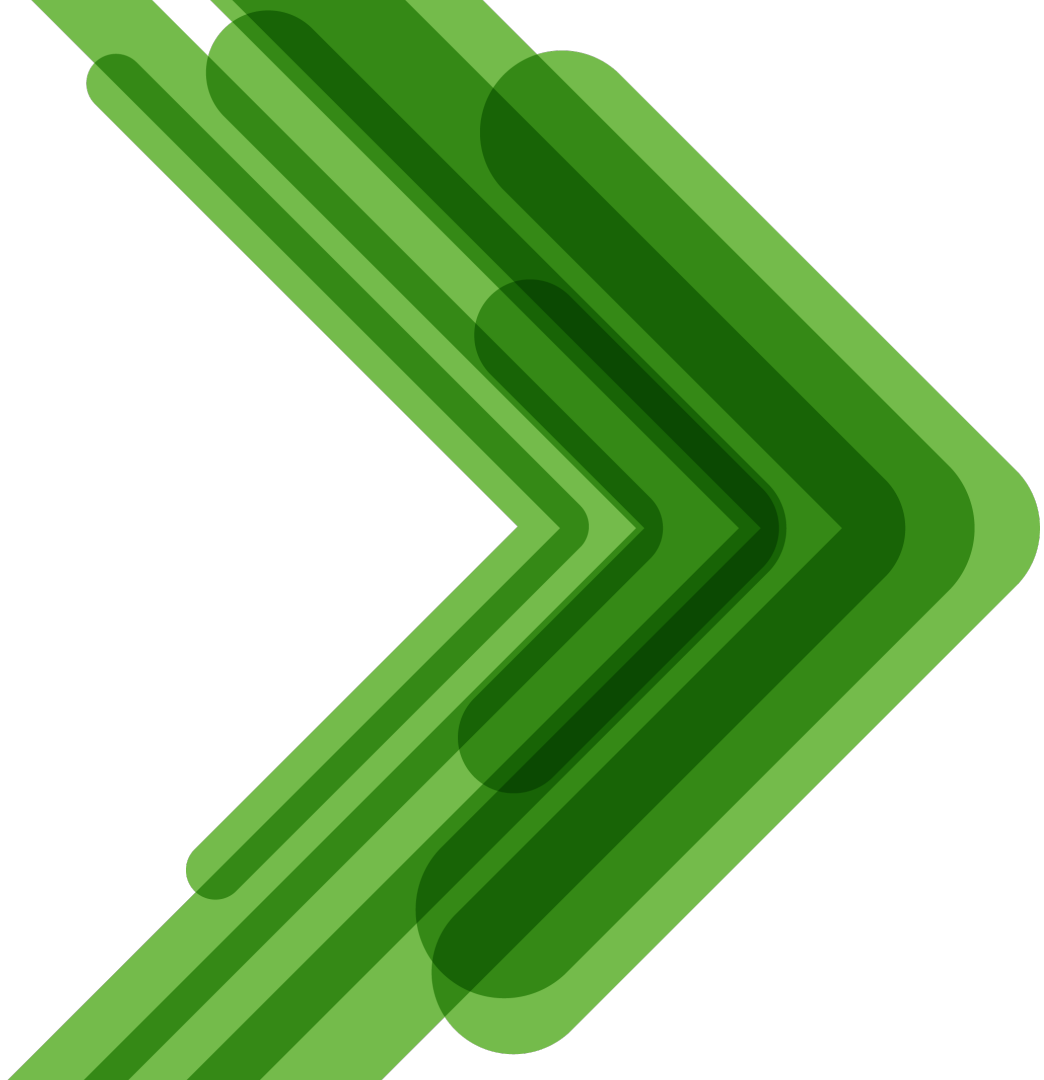
- Initial configurations provision 3 instances
- New instructions provision 3 NEW instances with NO knowledge of the original 3



Some Up-Sides of Statelessness

- Only impact the resources you explicitly want to impact
- Don't have to worry about conflicts
  - Whatever was in the last run is what happened
- Adoptation is easy (if required modules are avaialble)
  - If not, write one

Demo



- AAA Server
- Access List
- Address Pools
- Application Filters
- AS Path
- BFD Template
- Cipher Suite List
- Community List
- DHCP IPv6 Pool
- Distinguished Name
- DNS Server Group
- External Attributes
- File List
- FlexConfig
- Geolocation
- Interface
- Key Chain
- Network
- PKI
- Policy List
- Port
- Prefix List

Network

Add Network | Filter

Show Unused Objects

A network object represents one or more IP addresses. Network objects are used in various places, including access control policies, network variables, intrusion rules, identity rules, network discovery rules, event searches, reports, and so on.

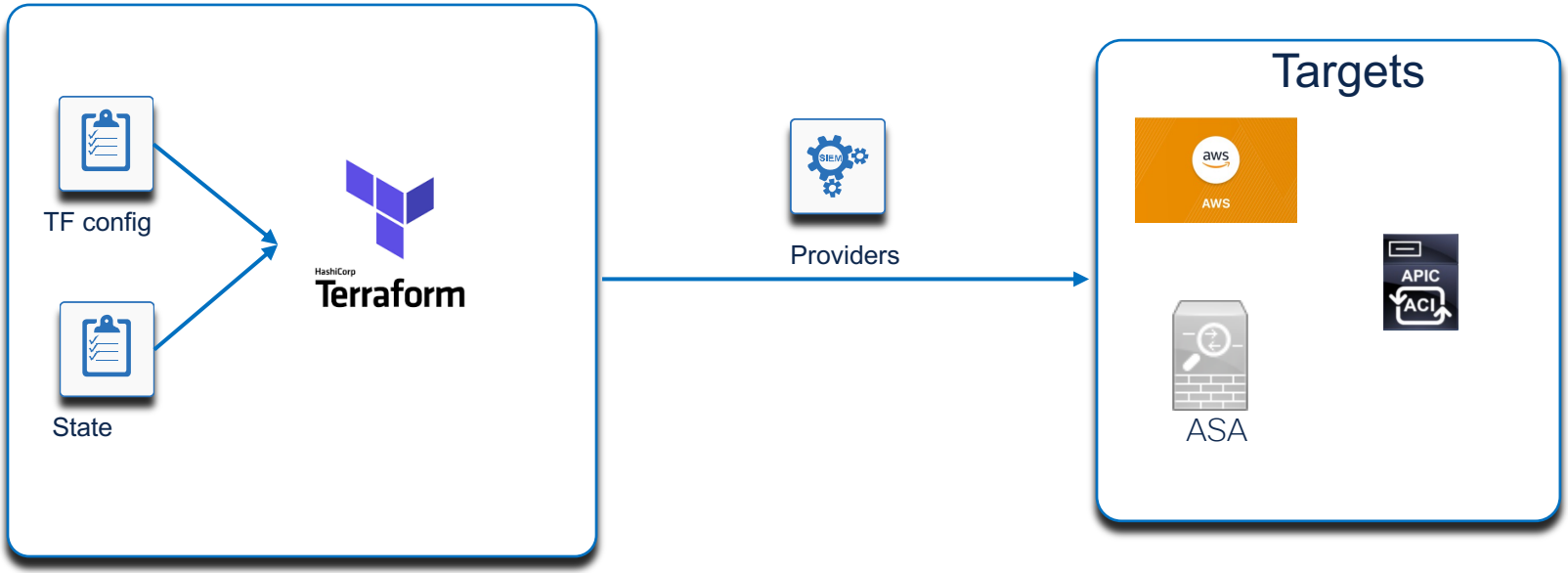
Name	Value	Type	Override
10.236.131.0_24	10.236.131.0/24	Network	
10.236.131.251	10.236.131.251	Host	
10.236.25.0_24	10.236.25.0/24	Network	
10.236.255.0_24	10.236.255.0/24	Network	
10.236.26.0_24	10.236.26.0/24	Network	
10.236.27.0_24	10.236.27.0/24	Network	
10.236.4.252	10.236.4.252	Host	
10.236.9.0_24	10.236.9.0/24	Network	
Africa	10.1.2.1	Host	
all-lans	10.0.0.0/8	Network	
any	0.0.0.0/0 ::/0	Group	

Now let’s Terraform!

Terraform – what is it?

- Developed by HashiCorp
- Initial release in July 2014
- Designed to be a full Infrastructure as Code (IaC) management tool for Cloud Infrastructure Provisioning
- Completely written in Go, creating a single binary file
- Fully declarative leveraging HashiCorp Configuration Language (HCL)

Terraform in a nutshell

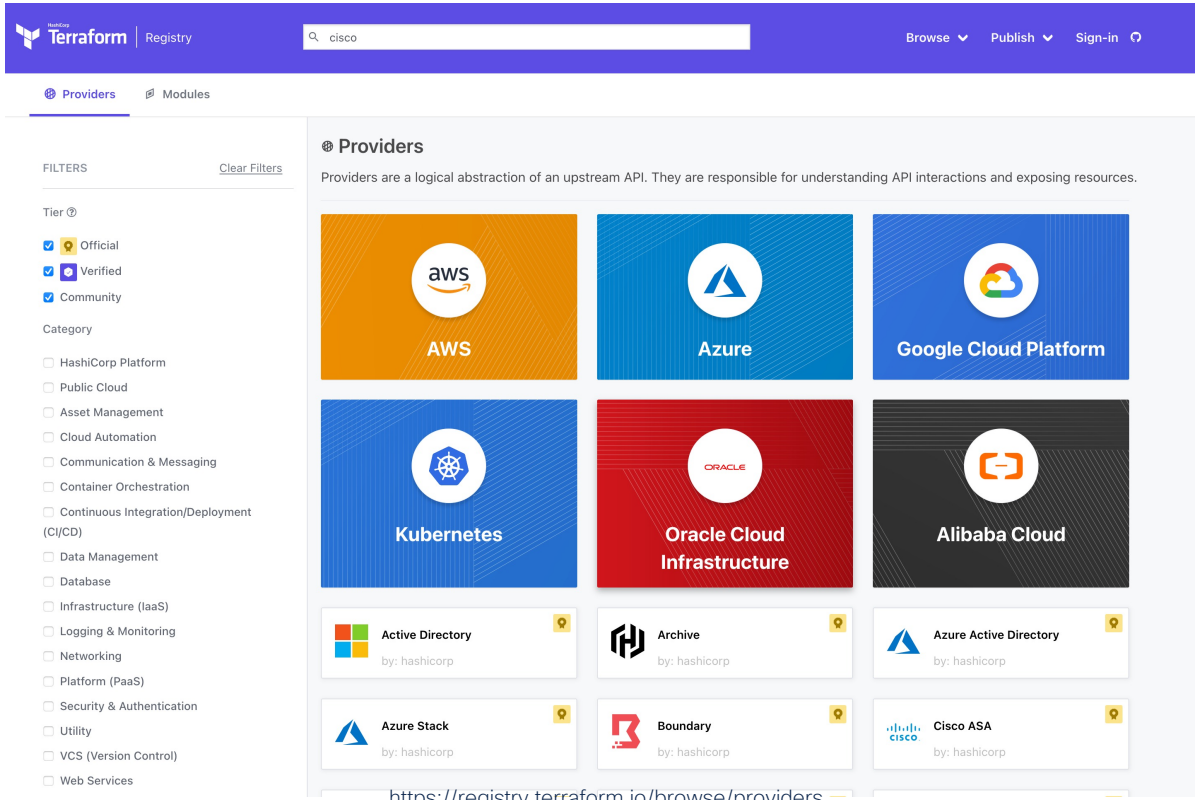


Terraform Providers

- Providers allow TF to communicate with desired resources
- Downloaded and installed automatically
- Signed for security

```
terraform {
  required_providers {
    aci = {
      source = "CiscoDevNet/aci"
    }
  }
}
```

Terraform Provider Registry



https://registry.terraform.io/browse/providers

Steps

- 1. Install Terraform
  - https://developer.hashicorp.com/terraform/tutorials/aws-get-started/install-cli
- 2. Add the necessary provider(s) to your terraform plugins directory
  - https://github.com/CiscoDevNet/terraform-provider-fmc/releases/tag/v1.4.6
- 3. Create the inventory file
  - The default inventory file is `terraform.tfvars`

Steps – cont.

- 4. create or download a playbook:
  - `main.tf`
- 5. run it:
  - `terraform init`
  - `terraform plan`
  - `terraform apply`
  - `(terraform destroy)`
- 6. check the state file :
  - `terraform.tfstate`

```
terraform {
  required_providers {
    fmc = {
      source = "CiscoDevNet/fmc"
      # version = "0.1.1"
    }
  }
}

provider  "fmc" {
  fmc_username=var.fmc_username
  fmc_password=var.fmc_password
  fmc_host=var.fmc_host
  fmc_insecure_skip_verify=var.fmc_insecure_skip_verify
}


locals {
  rg =csvdecode(file("./objs.csv"))
}


###
# CREATE OBJECTS
###

resource "fmc_host_objects" "newHostObj1" {
  for_each = { for rg in local.rg: rg.Name => rg}
  name= each.value["Name"]
  value = each.value["Value"]
  description = "1st test object from terraform"
}

output "fmc_host_objects_tumb_220" {
  value = fmc_host_objects.newHostObj1
}
```

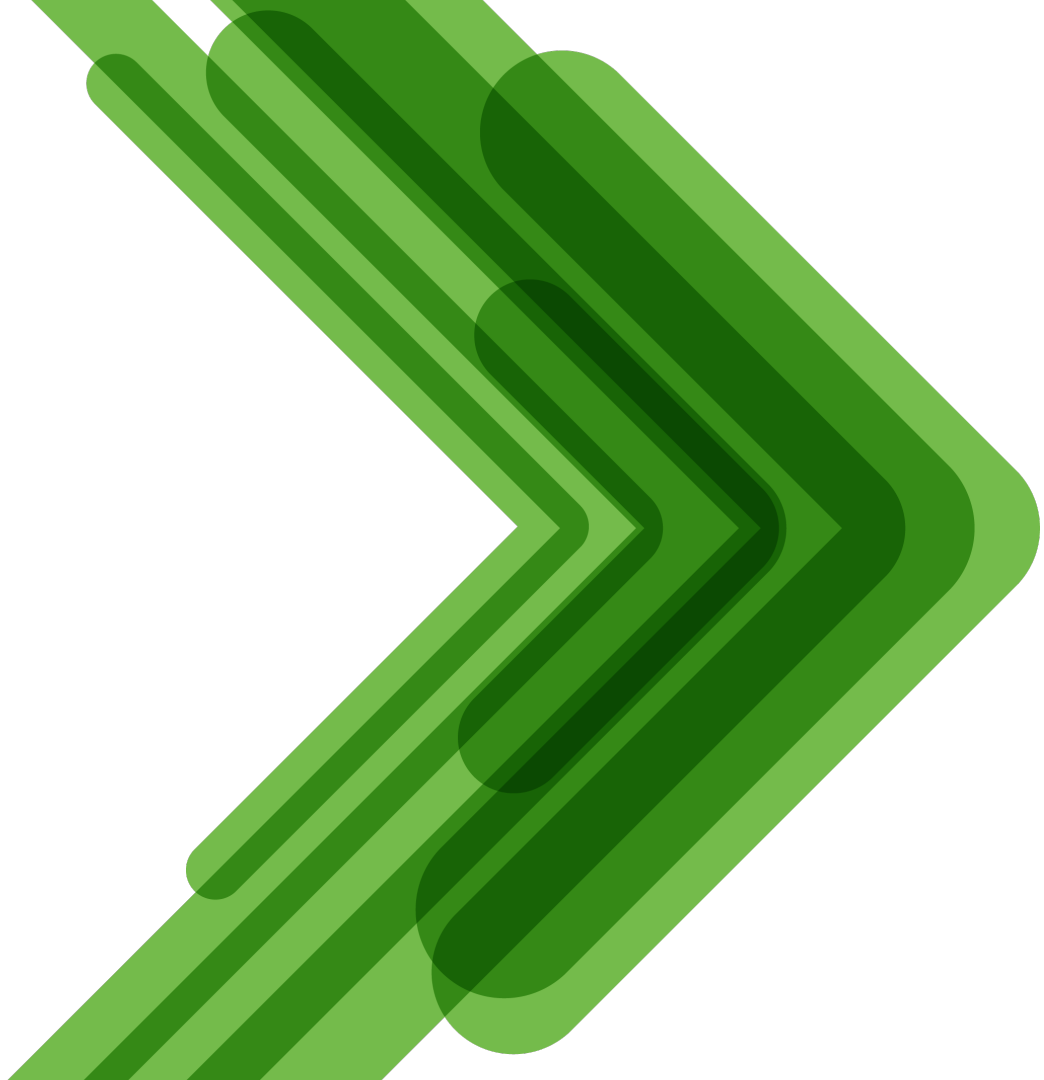
Terraform State File

- JSON file with ALL state data
- Includes default data not specified in configuration files
- Includes **secret data, saved as RAW text**
- Can be stored locally or REMOTELY (Secure S3 bucket – AWS)

terraform.tfstate


```
{
  "version": 4,
  "terraform_version": "1.5.4",
  "serial": 9,
  "lineage": "36f31ade-ecd1-f9b4-b62e-fcf476df3988",
  "outputs": {
    "fmc_host_objects_tumb_220": {
      "value": {
        "terra_test_host_1": {
          "description": "1st test object from terraform",
          "id": "005056AE-FC0E-0ed3-0000-257698489158",
          "name": "terra_test_host_1",
          "type": "Host",
          "value": "10.1.1.1"
        },
        "terra_test_host_2": {
          "description": "1st test object from terraform",
          "id": "005056AE-FC0E-0ed3-0000-257698489140",
          "name": "terra_test_host_2",
          "type": "Host",
          "value": "10.1.1.2"
        }
      },
```

Demo




main.tf
objs.csv
fmc_password=var.fmc_password
fmc_host=var.fmc_host
fmc_insecure_skip_verify=var.fmc_insecure_skip_verify
}
locals {
rg =csvdecode(file("./objs.csv"))
}
###
CREATE OBJECTS
###
resource "fmc_host_objects" "newHostObj1" {
for_each = { for rg in local.rg: rg.Name => rg}
name= each.value["Name"]
value = each.value["Value"]
description = "1st test object from terraform"
}
output "fmc_host_objects_tumb_220" {
value = fmc_host_objects.newHostObj1
}
Line 32, Column 2
Spaces: 2
Plain Text

Terraform vs Opentf (OpenToFu)

- **HashiCorp**, the company behind Terraform, made a pivotal decision last month to move away from its longstanding open-source licensing, opting instead for the **Business Source License (BSL) 1.1.** This license change was not limited to Terraform, but extended through other popular open source projects owned by HashiCorp such as **Vault** and **Nomad**. This transformation has sparked intense debates about the future of Terraform's open-source spirit.

  https://logz.io/blog/terraform-is-no-longer-open-source-is-opentofu-opentf-the-successor/

ChatGPT – Not yet

```
Warning: Value for undeclared variable

The root module does not declare a variable named "fmc_insecure_skip_verify" but a value was found in
file "terraform.tfvars". If you meant to use this value, add a "variable" block to the configuration.

To silence these warnings, use TF_VAR_... environment variables to provide certain "global" settings
to all configurations in your organization. To reduce the verbosity of these warnings, use the
-compact-warnings option.
```

```
Error: Unsupported argument

  on chatgpt_show_devices.tf line 26, in data "http" "fmc_auth":
  26:   headers = {

An argument named "headers" is not expected here.
```

```
Error: Invalid resource type

  on chatgpt_show_devices.tf line 37, in resource "http" "get_managed_devices":
  37: resource "http" "get_managed_devices" {

The provider hashicorp/http does not support resource type "http".

Did you intend to use the data source "http"? If so, declare this using a "data" block instead of a
"resource" block.
```

Ansible OR Terraform?

Ansible OR Terraform?

Ansible

- Stateless requirement
- Configuring, updating, resources in-place
- More than one point of configuration management

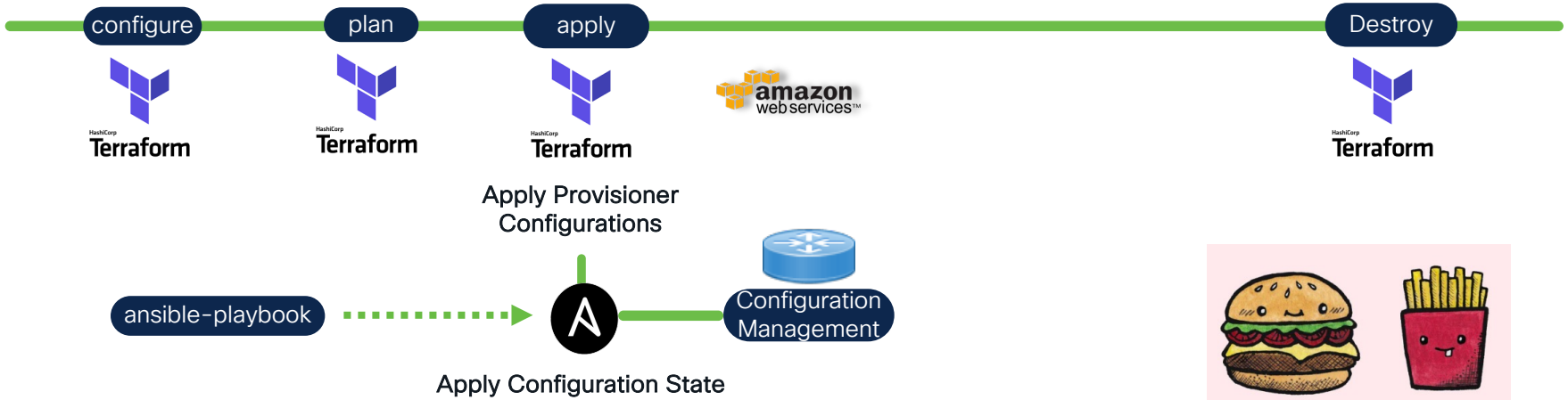
- Stateful requirement
- Provisioning and replacing long standing resources

They play nicely together

Ansible OR Terraform?

Terraform Workflow

Day-0/Day-1 – Terraform with Ansible Provisioner



Terraform WITH Ansible

```
resource "aws_instance" "cat8kv" {
  ami               = "ami-049489b50a99d699e"
  instance_type     = "t3.medium"
  availability_zone = "us-east-1a"
  key_name = "hwa-cat8kv-key"
  count = 3
  tags = {
    environment = "production"
  }

  user_data = "..."

  provisioner "local-exec" {
    command = "ansible-playbook config.yml --extra-vars 'ec2_ip=${aws_instance.cat8kv.public_ip}'"
  }
}
```

The bridge to possible