

Preventing Kubernetes Misconfiguration: Static Analysis and Beyond

Matt Johnson
Developer Advocate Lead

bridgecrew

AGENDA



Misconfiguration challenges



Write policy as code



Automate in our CI Pipeline



Helm chart analysis

Runtime analysis of k8 cluster



Matt Johnson

 @Metahertz
 metahertz

As an engineer
I want to move fast

I DO NOT
want to break things

The thing I have
love/hate relationship with is...



BC-3111

Do not admit root containers

 Edit	 Comment	 Assign	Started work	Finished work	Workflow ▾	Admin ▾		
Type:	Bug	Status:	TO DO	(View workflow)	People			
Priority:	Highest	Resolution:	Unresolved		Assignee:	Barak Schoster		
Components:	SecEng				Reporter:	Or Evron		
Labels:	PRODUCTION				Votes:	0 Vote for this issue		
Sprint:					Watchers:	1 Start watching this issue		

BC-3112

B
D

Do not admit containers wishing to share the host network namespace



Edit



Comment



Assign

Started work

Finished work

Workflow ▾

Admin ▾



...

Type:	<input checked="" type="checkbox"/> Bug	Status:	TO DO (View workflow)
Priority:	⬆ Medium	Resolution:	Unresolved
Components:	SecEng		
Labels:	None		Reporter:
Sprint:			Votes:

Description

Bridgecrew has been admitting containers that want to share the host's network namespace.

This is a security risk because it allows the container to access the host's network stack.

We need to prevent this from happening by default.

People

Assignee:

 Nimrod Kor
[Assign to me](#)

Reporter:

 Or Evron

Votes:

 0 [Vote for this issue](#)

Watchers:

 1 [Start watching this issue](#)

BC-3112

D Do not admit containers wishing to share the host network namespace

BC-3113

A Ensure no security groups allow ingress from 0.0.0.0:0 to port 22

Type

Edit

Comment

Assign

Started work

Finished work

Workflow ▾

Admin ▾



Priority

Bug

Status:

TO DO

(View workflow)

People

Assignee:



Nimrod Kor

[Assign to me](#)

Label

Medium

Resolution:

Unresolved

Reporter:



Or Evron

Sprint

SecEng

Labels:

None

Votes:



[0 Vote for this issue](#)

Watchers:



[1 Start watching this issue](#)

Description



And this is where
our story begins...



Picard Tips @PicardTips · Aug 5

▼

Picard leadership tip: Resources are always limited. Time, expertise, material, energy. Don't prioritize foolishness when you have been tasked to solve real problems.

6

284

997

↑

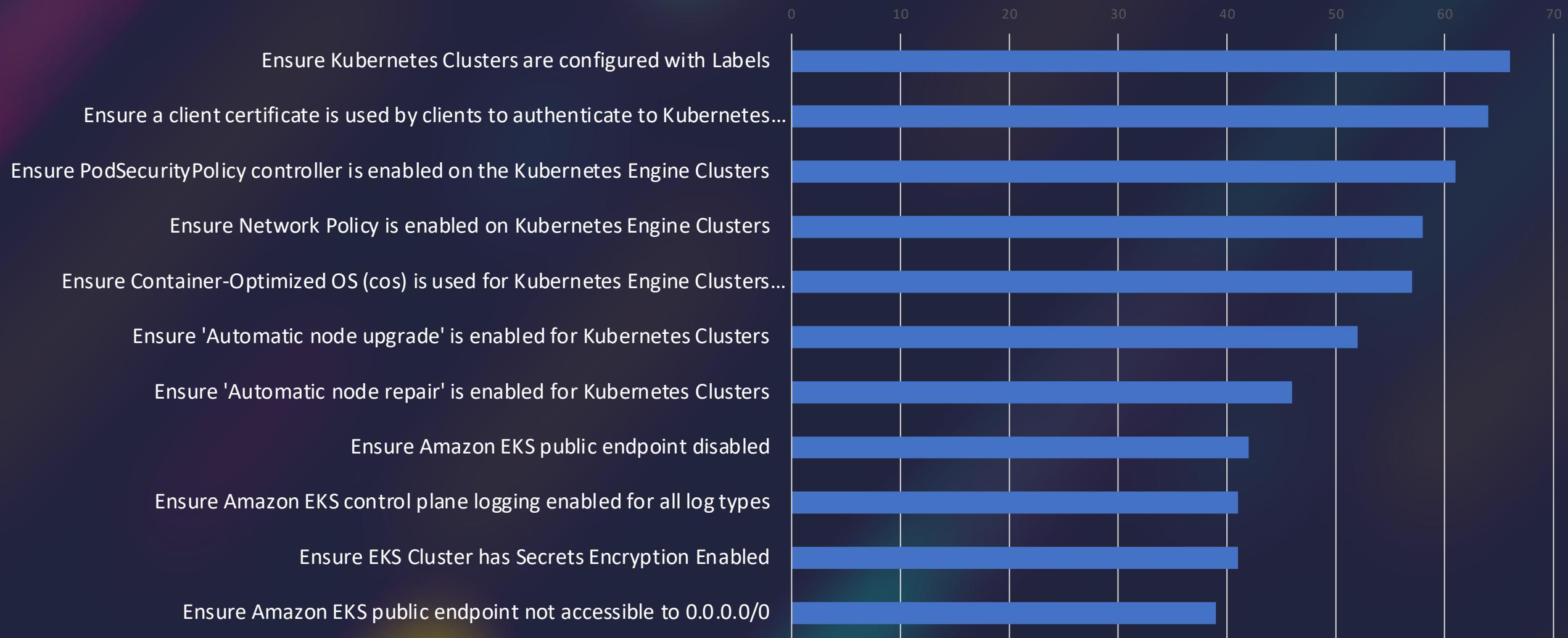
A close-up photograph of a man's face, likely Captain Jean-Luc Picard from Star Trek: The Next Generation. He is wearing a red Starfleet uniform. His right hand is raised to his forehead, with his fingers resting against his temple, partially obscuring his eyes. He has a serious, contemplative expression. The background is dark and out of focus.

So let's open our eyes
and look at some...



...data

Top Failing Kubernetes checks



A team of people in a control room, seen from behind, are monitoring a large screen displaying sensor activity. The screen shows a map at the top left, followed by a list of detected anomalies and their coordinates. The main title is "SENSOR ACTIVITY 060269".

Infrastructure as code (IaC) presents
a new **risk** and a new **opportunity**

```
13  apiVersion: v1
14  kind: Pod
15  metadata:
16    name: pod2
17  spec:
18    securityContext:
19      runAsNonRoot: true
20    containers:
21      - name: main
22        image: alpine
23        command: ["/bin/sleep", "999999"]
24    securityContext:
25      runAsNonRoot: false
26
```

```
apiVersion: v1
kind: Pod
metadata:
  name: pod2
spec:
  containers:
  - name: main
    image: alpine
    command: ["/bin/sleep", "999999"]
    securityContext:
      runAsNonRoot: true
```

A Star Trek character with reddish-brown hair, wearing a gold uniform, is seated at a control panel with multiple glowing blue and green screens. He is looking slightly off-camera with a serious expression.

<https://github.com/bridgecrewio/checkov>

README.md

★ Unstar

1.5k

Fork

141

checkov

by bridgecrew

maintained by [bridgecrew.io](#)

 build passing

 security passing

coverage 86%

docs passing

pypi v1.0.589

python v3.7

tf >=0.12.0

downloads 846k

slack 169

- Released publicly in December 2019
- Apache 2.0 license
- 50+ contributors
- >800K downloads
- >1400 stars
- Written in Python



```
tronxd@Naors-MacBook-Pro ~ % pipenv run checkov -d /Users/tronxd/PycharmProjects/checkov/tests/terraform/runner/resources/example -o cli  
version: 1.0.64  
Passed checks: 1, Failed checks: 5, Skipped checks: 1, Parsing errors: 1  
Check: "Ensure all data stored in the S3 bucket have versioning enabled"  
PASSED for resource: aws_s3_bucket.foo-bucket  
File: /example.tf:60-73  
  
Check: "Ensure Azure Instance does not use basic authentication(Use SSH Key Instead)"  
FAILED for resource: azurerm_virtual_machine.main  
File: /example.tf:1-39  
1 | resource "azurerm_virtual_machine" "main" {  
2 |   name           = "${var.prefix}-vm"  
3 |   location        = "${azurerm_resource_group.main.location}"  
4 |   resource_group_name = "${azurerm_resource_group.main.name}"  
5 |   network_interface_ids = [  
6 |     "${azurerm_network_interface.main.id}"  
7 |   ]  
8 |   vm_size         = "Standard_DS1_v2"  
9 |  
10 |  # Uncomment this line to delete the OS disk automatically when deleting the VM  
10 |  # delete_os_disk_on_termination = true  
11 |  
12 |  
13 |  # Uncomment this line to delete the data disks automatically when deleting the VM  
13 |  # delete_data_disks_on_termination = true  
14 |  
15 |  
16 |  storage_image_reference {  
17 |    publisher = "Canonical"  
18 |    offer     = "UbuntuServer"  
19 |    sku       = "16.04-LTS"  
20 |    version   = "latest"  
21 |  }  
22 |  storage_os_disk {  
23 |    name           = "myosdisk1"  
24 |    caching        = "ReadWrite"  
25 |    create_option   = "FromImage"  
26 |    managed_disk_type = "Standard_LRS"  
27 |  }  
28 |  os_profile {  
29 |    computer_name  = "hostname"  
30 |    admin_username = "testadmin"  
31 |    admin_password = "Password1234!"  
32 |  }  
33 |  os_profile_linux_config {  
34 |    disable_password_authentication = false  
35 |  }  
36 |  tags = {  
37 |    environment = "staging"  
38 |  }  
39 | }  
  
Check: "Ensure Azure managed disk have encryption enabled"  
FAILED for resource: azurerm_managed_disk.source  
File: /example.tf:41-50  
41 | resource "azurerm_managed_disk" "source" {  
42 |   encryption_settings {  
43 |     enabled = false  
44 |   }  
45 |   create_option      = ""
```

Checkov statically analyzes
for known best practices
implemented in IaC
manifests like k8s YAML



Policy as code

- Version controlled
- Peer reviewed
- Can utilize inheritance and have code reuse (python)
- Part of SDLC
- Continuous integration

```
class CPULimits(BaseK8Check):

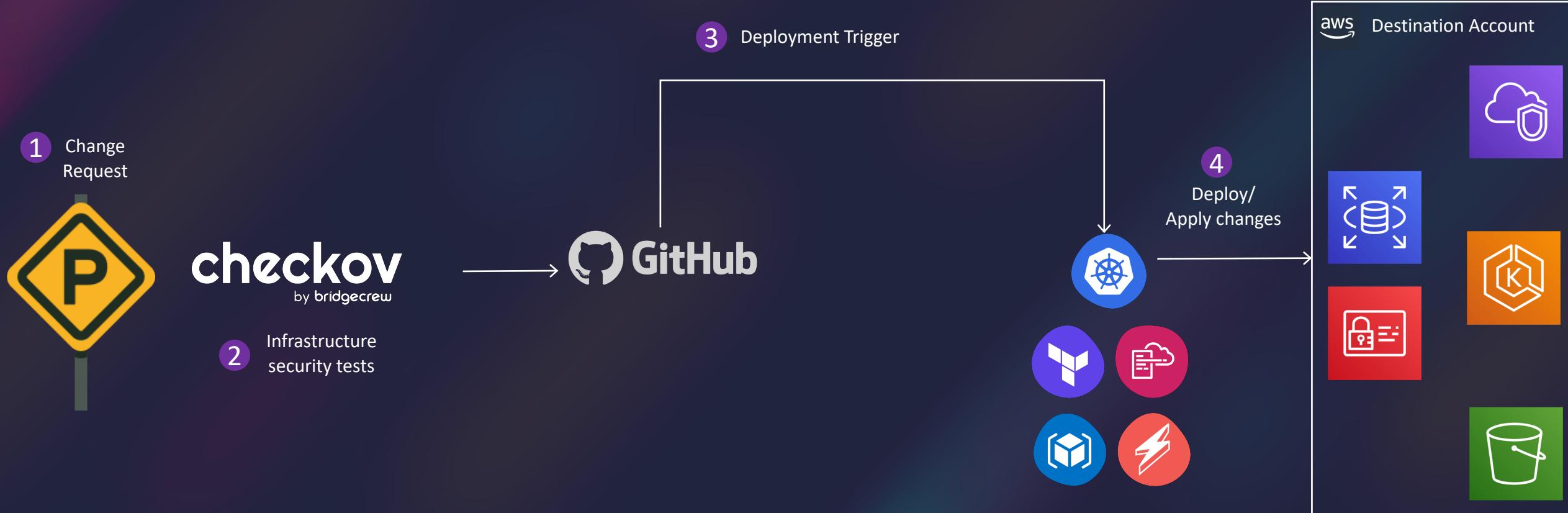
    def __init__(self):
        name = "CPU limits should be set"
        id = "CKV_K8S_11"
        # Location: container .resources.limits.cpu
        supported_kind = ['containers', 'initContainers']
        categories = [CheckCategories.KUBERNETES]
        super().__init__(name=name, id=id, categories=categories, supported_entities=supported_kind)

    def get_resource_id(self, conf):
        return f'{conf["parent"]} - {conf["name"]}'

    def scan_spec_conf(self, conf):
        if "resources" in conf:
            if "limits" in conf["resources"]:
                if "cpu" not in conf["resources"]["limits"]:
                    return CheckResult.FAILED
            else:
                return CheckResult.FAILED
        else:
            return CheckResult.FAILED
    return CheckResult.PASSED
```

A group of Star Trek crew members in the bridge of the USS Enterprise. In the foreground, from left to right, are Mr. Scott (blue shirt), Captain Kirk (yellow shirt), Mr. Spock (blue shirt), and Mr. Chekov (blue shirt). Behind them are Mr. Sulu (red shirt), Mr. McCoy (orange shirt), and Lt. Uhura (red shirt).

Brace for live demo



A group of Star Trek characters in their iconic uniforms are gathered in the bridge of the USS Enterprise. In the foreground, Captain Kirk (William Shatner) sits in his chair, flanked by Mr. Spock (Leonard Nimoy) on his left and Dr. McCoy (DeForest Kelley) on his right. Behind them stand the First Officer (Zefram Cochrane), the Science Officer (Hikaru Sulu), and the Medical Officer (Chekov). The background features the ship's control panels and windows showing space and a distant planet.

Brace for live demo



checkov
by bridgecrew





A group of Star Trek characters in their respective uniforms are gathered in the bridge of the USS Enterprise. In the foreground, Captain Kirk (William Shatner) sits in his chair, flanked by Mr. Spock (Leonard Nimoy) on his left and Mr. Scott (James Doohan) on his right. Behind them stand the First Officer (Zefram Cochrane), Dr. McCoy (DeForest Kelley), and Lt. Uhura (Nichelle Nichols). The background features the helm and various control panels. A large, semi-transparent rectangular box covers the upper portion of the image, containing the text "Another one!".

Another one!

Runtime analysis of K8s cluster

Integrate Checkov with Kubernetes

Background

Checkov is built to scan static code and is typically used at build time. However, resources running in a Kubernetes cluster can be described in the same way as at build time. This allows Checkov to run in a cluster with read-only access and report on the same violations.

Execution

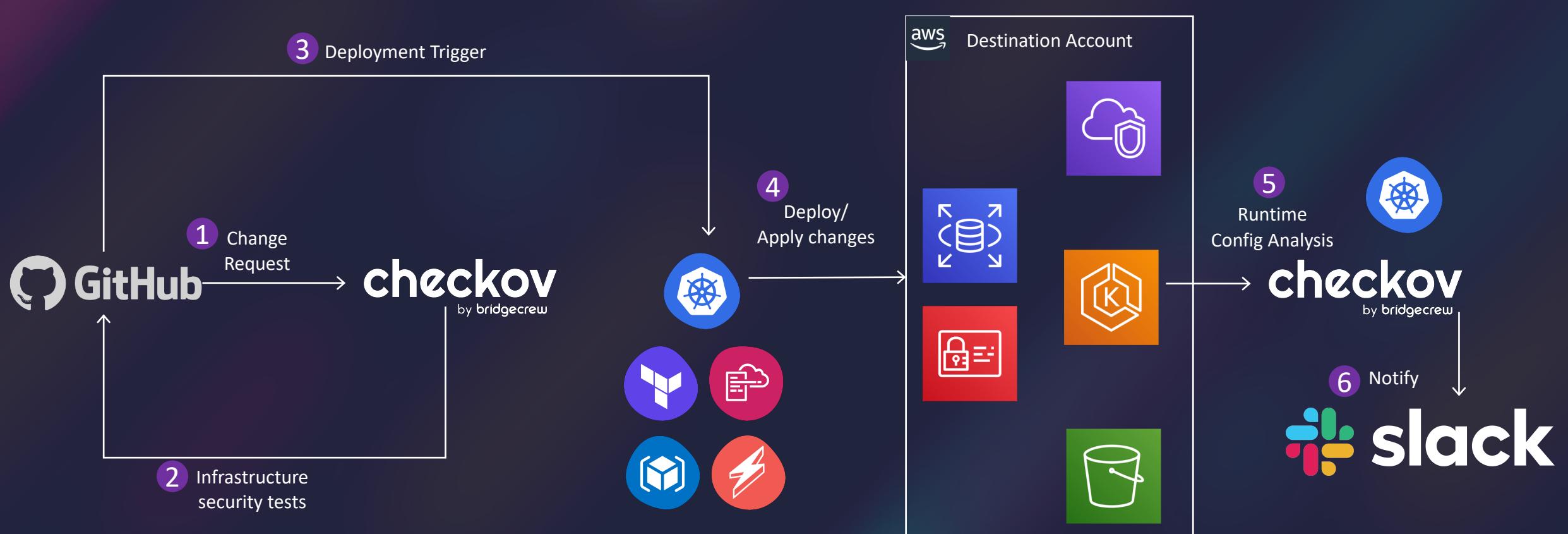
To run Checkov in your cluster you must have Kubernetes CLI access to the cluster.

To execute a job against your cluster, run the following manifest.

```
kubectl apply -f https://raw.githubusercontent.com/bridgecrewio/checkov/master/kubernetes/checkov.yaml
```

Review the output of the job.

```
kubectl get jobs -n checkov  
kubectl logs job/checkov -n checkov
```



Misconfig Analysis



Pre-commit



Continuous
Integration



Running
Cluster

A WORLD WHERE:



Infrastructure is developed and secured in the same place



Issues are automatically prevented from being deployed



Security is a business enabler rather than a hindrance

TAKEAWAYS

Keep your Kubernetes manifests and Helm charts secure

Have a fast feedback loop on configuration changes

Monitor both build-time and runtime

Version control your policies

bridgecrew

Try Checkov and join our
Slack slack.bridgecrew.io

CONTACT ME matt@bridgecrew.io



**WEAR A MASK.
DON'T BE A RED SHIRT.**

imgflip.com