LATACORA

All the Data, All the Time:

A New Strategy for Security Tooling

Xavier Garceau-Aranda ⌨️☁️🛡️
Principal Security Engineer @ Latacora

Hello, ~~World~~ **DevOps**Con!

LATACORA

aws

okta

▲Vercel

the BROWSER COMPANY of NEW YORK

panther

CLOUDFLARE

Google Workspace

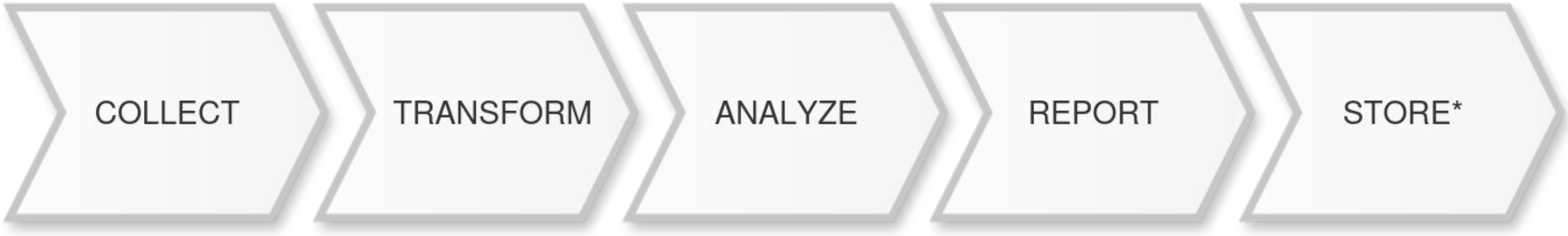
tailscale

SCOUTSUITE

Teleport

Notion

Current state of security tooling



COLLECT

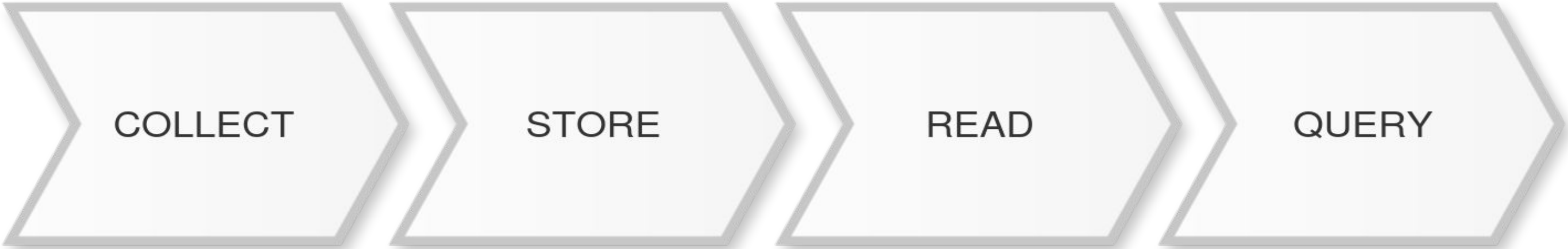
TRANSFORM

ANALYZE

REPORT

STORE*

Our approach



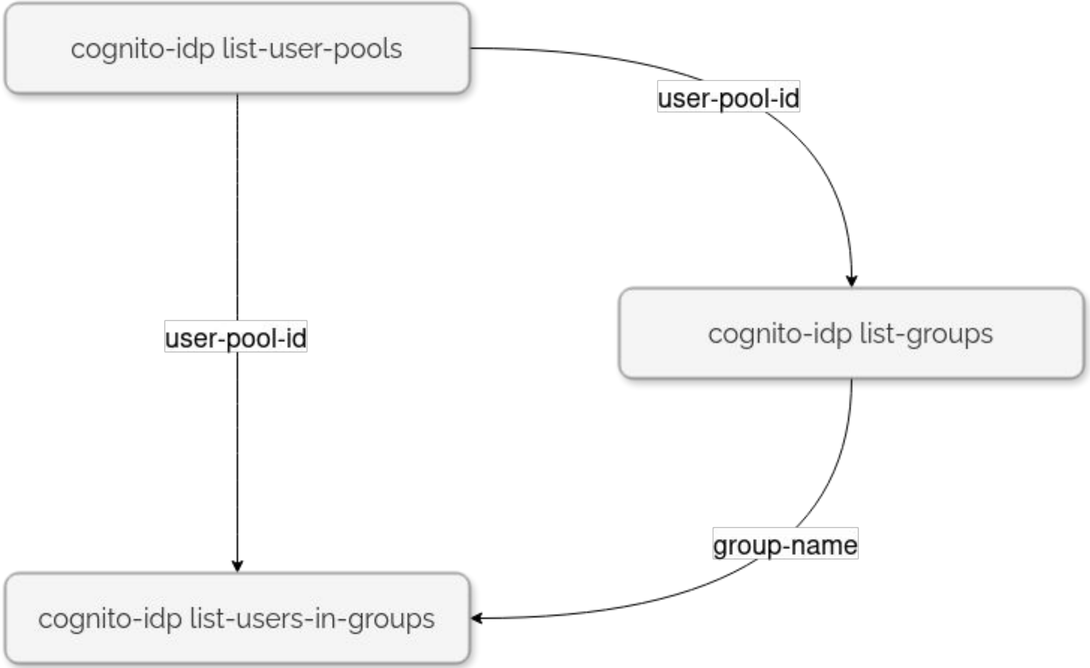
COLLECT

STORE

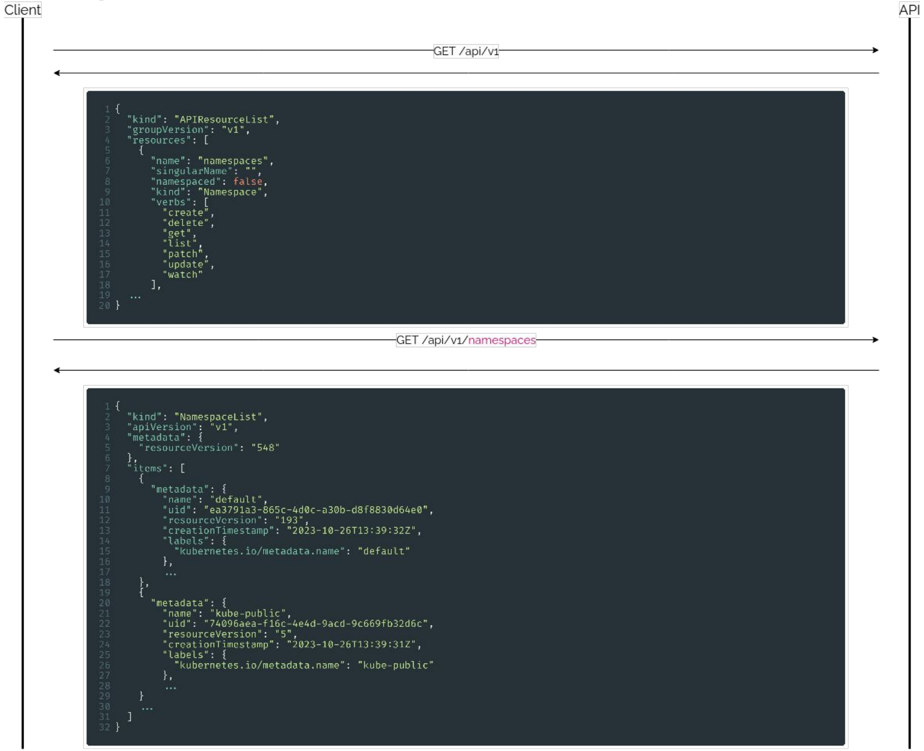
READ

QUERY

First step: retrieving "all the things"



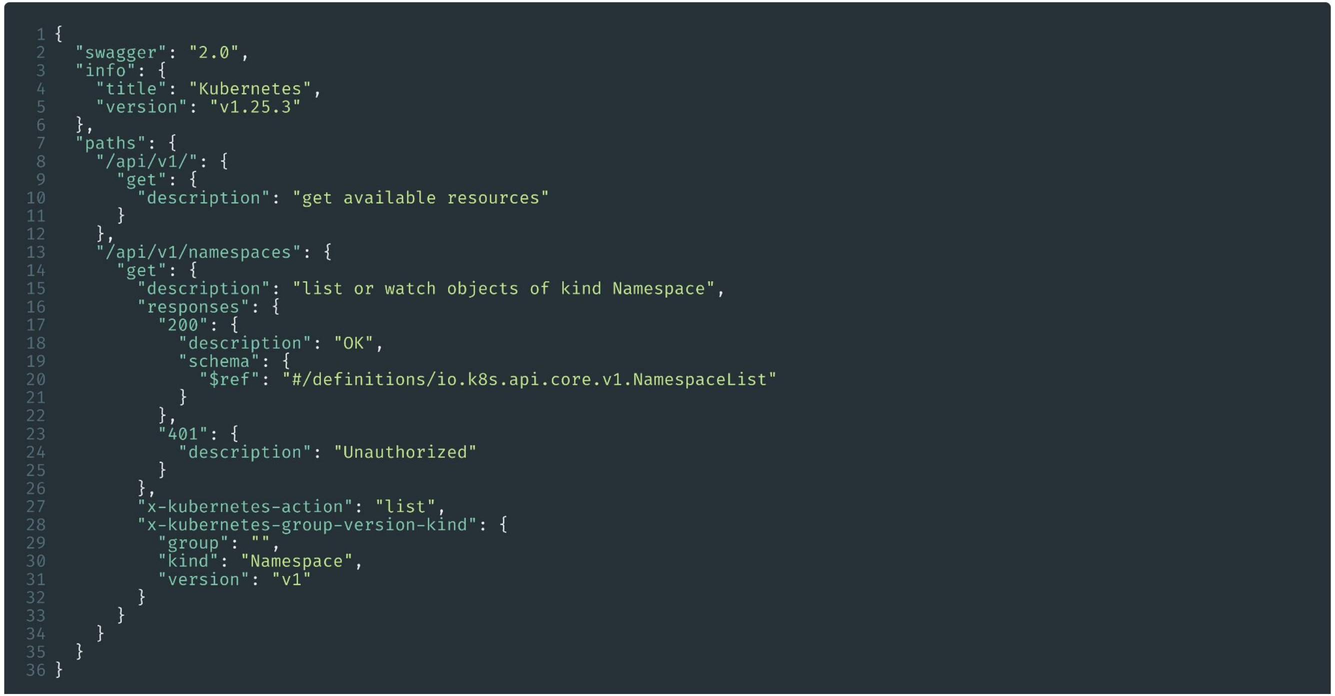
Simple APIs (K8s)



```
{
  "kind": "APIResourceList",
  "groupVersion": "v1",
  "resources": [
    {
      "name": "namespaces",
      "singularName": "",
      "namespaced": false,
      "kind": "Namespace",
      "verbs": [
        "create",
        "delete",
        "get",
        "list",
        "patch",
        "update",
        "watch"
      ],
  ...
}
```

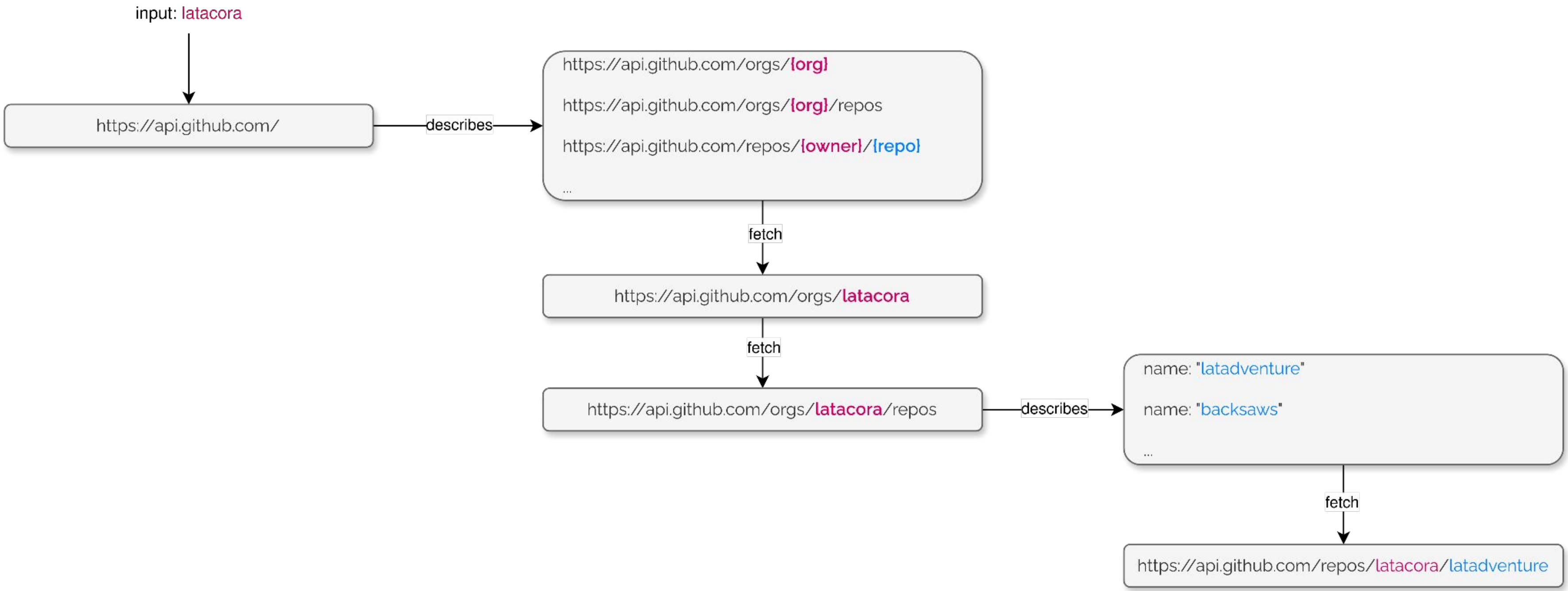
```
{
  "kind": "NamespaceList",
  "apiVersion": "v1",
  "metadata": {
    "resourceVersion": "548"
  },
  "items": [
    {
      "metadata": {
        "name": "default",
        "uid": "ea3791a3-865c-4d0c-a30b-d8f8830d64e0",
        "resourceVersion": "193",
        "creationTimestamp": "2023-10-26T13:39:32Z",
        "labels": {
          "kubernetes.io/metadata.name": "default"
        },
        ...
    },
    {
      "metadata": {
        "name": "kube-public",
        "uid": "74096aea-f16c-4e4d-9acd-9c669fb32d6c",
        "resourceVersion": "5",
        "creationTimestamp": "2023-10-26T13:39:31Z",
        "labels": {
          "kubernetes.io/metadata.name": "kube-public"
        },
        ...
    }
    ...
  ]
}
```

Formal APIs (K8s)



```
{
  "swagger": "2.0",
  "info": {
    "title": "Kubernetes",
    "version": "v1.25.3"
  },
  "paths": {
    "/api/v1/": {
      "get": {
        "description": "get available resources"
      }
    },
    "/api/v1/namespaces": {
      "get": {
        "description": "list or watch objects of kind Namespace",
        "responses": {
          "200": {
            "description": "OK",
            "schema": {
              "$ref": "#/definitions/io.k8s.api.core.v1.NamespaceList"
            }
          },
          "401": {
            "description": "Unauthorized"
          }
        },
        "x-kubernetes-action": "list",
        "x-kubernetes-group-version-kind": {
          "group": "",
          "kind": "Namespace",
          "version": "v1"
        }
      }
    }
  }
}
```

Formal APIs (GitHub)



input: latacora

https://api.github.com/

describes

https://api.github.com/orgs/{org}

https://api.github.com/orgs/{org}/repos

https://api.github.com/repos/{owner}/{repo}

...

fetch

https://api.github.com/orgs/latacora

fetch

https://api.github.com/orgs/latacora/repos

describes

name: "latadventure"

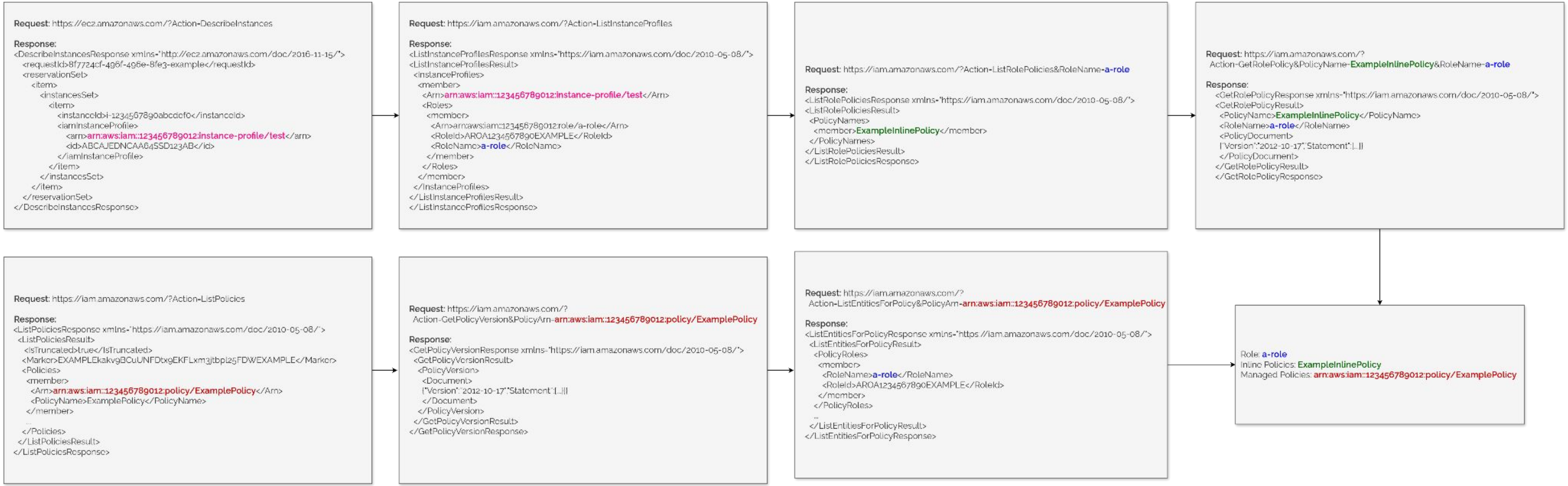
name: "backsaws"

...

fetch

https://api.github.com/repos/latacora/latadventure

Complex APIs (AWS)



Request: https://ec2.amazonaws.com/?Action=DescribeInstances

Response:

```
<DescribeInstancesResponse xmlns="http://ec2.amazonaws.com/doc/2016-11-15/">
  <requestId>8f7724cf-496f-496e-8fe3-example</requestId>
  <reservationSet>
    <item>
      <instancesSet>
        <item>
          <instanceId>i-1234567890abcdef0</instanceId>
          <iamInstanceProfile>
            <arn>arn:aws:iam::123456789012:instance-profile/test</arn>
            <id>ABCAJEDNCAA64SSD123AB</id>
          </iamInstanceProfile>
        </item>
      </instancesSet>
    </item>
  </reservationSet>
</DescribeInstancesResponse>
```

Request: https://iam.amazonaws.com/?Action=ListInstanceProfiles

Response:

```
<ListInstanceProfilesResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
<ListInstanceProfilesResult>
 <InstanceProfiles>
  <member>
   <Arn>arn:aws:iam::123456789012:instance-profile/test</Arn>
   <Roles>
    <member>
     <Arn>arn:aws:iam::123456789012:role/a-role</Arn>
     <RoleId>AROA1234567890EXAMPLE</RoleId>
     <RoleName>a-role</RoleName>
    </member>
   </Roles>
  </member>
 </InstanceProfiles>
</ListInstanceProfilesResult>
</ListInstanceProfilesResponse>
```

Request: https://iam.amazonaws.com/?Action=ListRolePolicies&RoleName=a-role

Response:

```
<ListRolePoliciesResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
<ListRolePoliciesResult>
 <PolicyNames>
  <member>ExampleInlinePolicy</member>
 </PolicyNames>
</ListRolePoliciesResult>
</ListRolePoliciesResponse>
```

Request: https://iam.amazonaws.com/?
Action=GetRolePolicy&PolicyName=ExampleInlinePolicy&RoleName=a-role

Response:

```
  <GetRolePolicyResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <GetRolePolicyResult>
   <PolicyName>ExampleInlinePolicy</PolicyName>
   <RoleName>a-role</RoleName>
   <PolicyDocument>
   {"Version":"2012-10-17","Statement":[...]}
   </PolicyDocument>
  </GetRolePolicyResult>
  </GetRolePolicyResponse>
```

Request: https://iam.amazonaws.com/?Action=ListPolicies

Response:

```
<ListPoliciesResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
 <ListPoliciesResult>
  <IsTruncated>true</IsTruncated>
  <Marker>EXAMPLEkakv9BCuUNFDtx9EKFLxm3jtbpl25FDWEXAMPLE</Marker>
  <Policies>
   <member>
    <Arn>arn:aws:iam::123456789012:policy/ExamplePolicy</Arn>
    <PolicyName>ExamplePolicy</PolicyName>
   </member>
   ...
  </Policies>
 </ListPoliciesResult>
</ListPoliciesResponse>
```

Request: https://iam.amazonaws.com/?
Action=GetPolicyVersion&PolicyArn=arn:aws:iam::123456789012:policy/ExamplePolicy

Response:

```
<GetPolicyVersionResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
 <GetPolicyVersionResult>
  <PolicyVersion>
   <Document>
   {"Version":"2012-10-17","Statement":[...]}
   </Document>
  </PolicyVersion>
 </GetPolicyVersionResult>
</GetPolicyVersionResponse>
```

Request: https://iam.amazonaws.com/?
Action=ListEntitiesForPolicy&PolicyArn=arn:aws:iam::123456789012:policy/ExamplePolicy

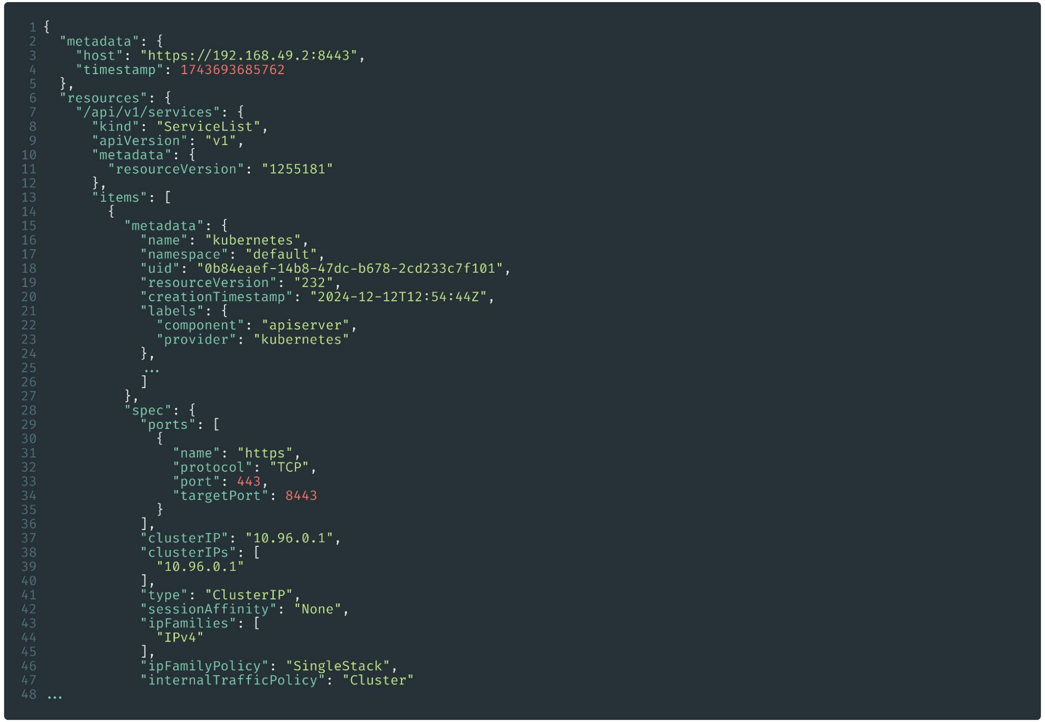
Response:

```
<ListEntitiesForPolicyResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
 <ListEntitiesForPolicyResult>
  <PolicyRoles>
   <member>
    <RoleName>a-role</RoleName>
    <RoleId>AROA1234567890EXAMPLE</RoleId>
   </member>
  </PolicyRoles>
  ...
 </ListEntitiesForPolicyResult>
</ListEntitiesForPolicyResponse>
```

Role: a-role
Inline Policies: ExampleInlinePolicy
Managed Policies: arn:aws:iam::123456789012:policy/ExamplePolicy

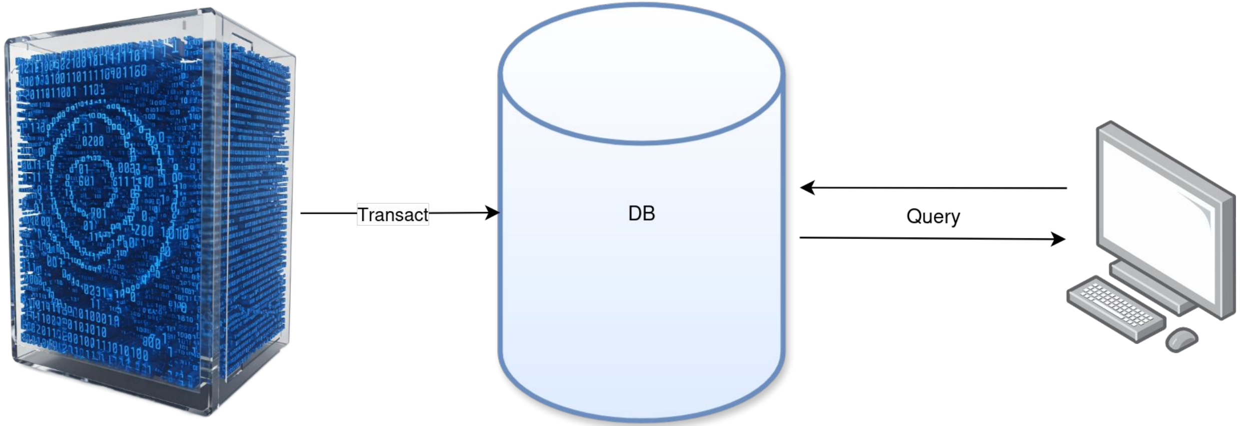
Snapshots

Snapshots

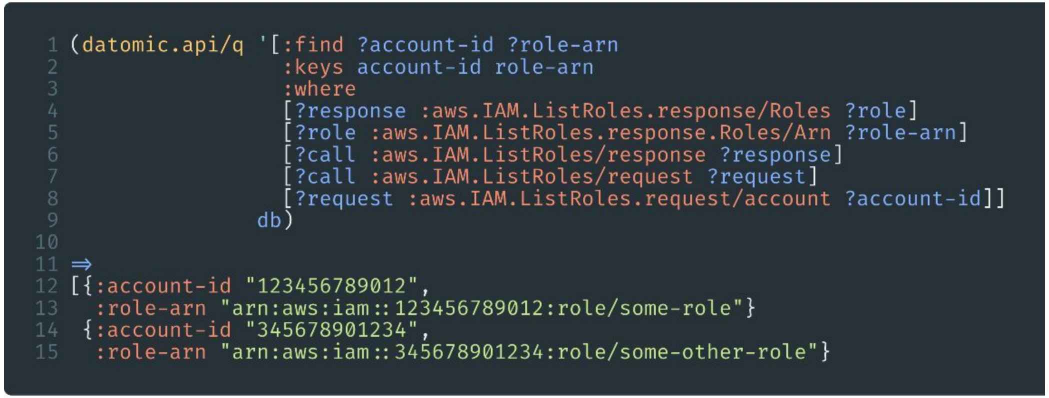


```
{
  "metadata": {
    "host": "https://192.168.49.2:8443",
    "timestamp": 1743693685762
  },
  "resources": {
    "/api/v1/services": {
      "kind": "ServiceList",
      "apiVersion": "v1",
      "metadata": {
        "resourceVersion": "1255181"
      },
      "items": [
        {
          "metadata": {
            "name": "kubernetes",
            "namespace": "default",
            "uid": "0b84eaef-14b8-47dc-b678-2cd233c7f101",
            "resourceVersion": "232",
            "creationTimestamp": "2024-12-12T12:54:44Z",
            "labels": {
              "component": "apiserver",
              "provider": "kubernetes"
            },
            ...
            ]
          },
          "spec": {
            "ports": [
              {
                "name": "https",
                "protocol": "TCP",
                "port": 443,
                "targetPort": 8443
              }
            ],
            "clusterIP": "10.96.0.1",
            "clusterIPs": [
              "10.96.0.1"
            ],
            "type": "ClusterIP",
            "sessionAffinity": "None",
            "ipFamilies": [
              "IPv4"
            ],
            "ipFamilyPolicy": "SingleStack",
            "internalTrafficPolicy": "Cluster"
...
```

Queries

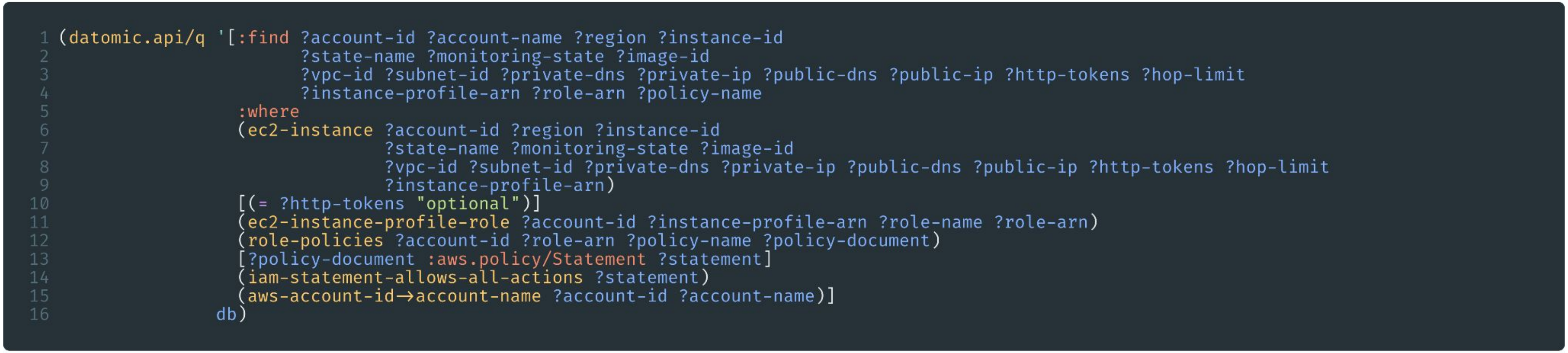


Querying with Datomic



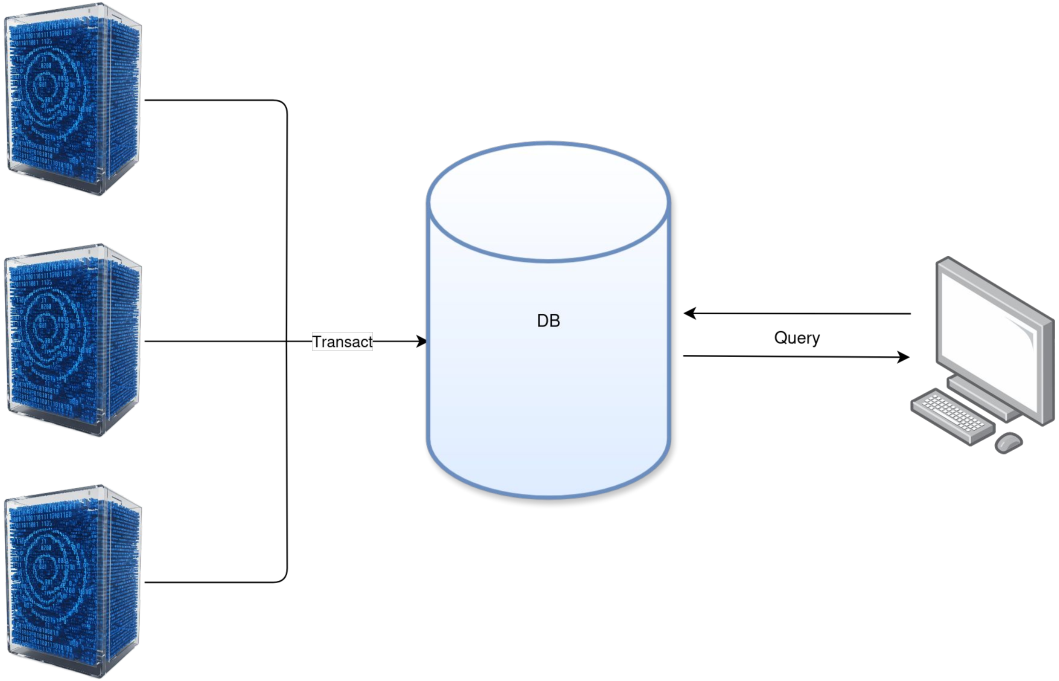
```
(datomic.api/q '[:find ?account-id ?role-arn
                 :keys account-id role-arn
                 :where
                 [?response :aws.IAM.ListRoles.response/Roles ?role]
                 [?role :aws.IAM.ListRoles.response.Roles/Arn ?role-arn]
                 [?call :aws.IAM.ListRoles/response ?response]
                 [?call :aws.IAM.ListRoles/request ?request]
                 [?request :aws.IAM.ListRoles.request/account ?account-id]]
               db)

⇒
[{:account-id "123456789012",
  :role-arn "arn:aws:iam::123456789012:role/some-role"}
 {:account-id "345678901234",
  :role-arn "arn:aws:iam::345678901234:role/some-other-role"}
```

EC2 instances with IMDSv1 enabled & an IAM role with excessive permissions

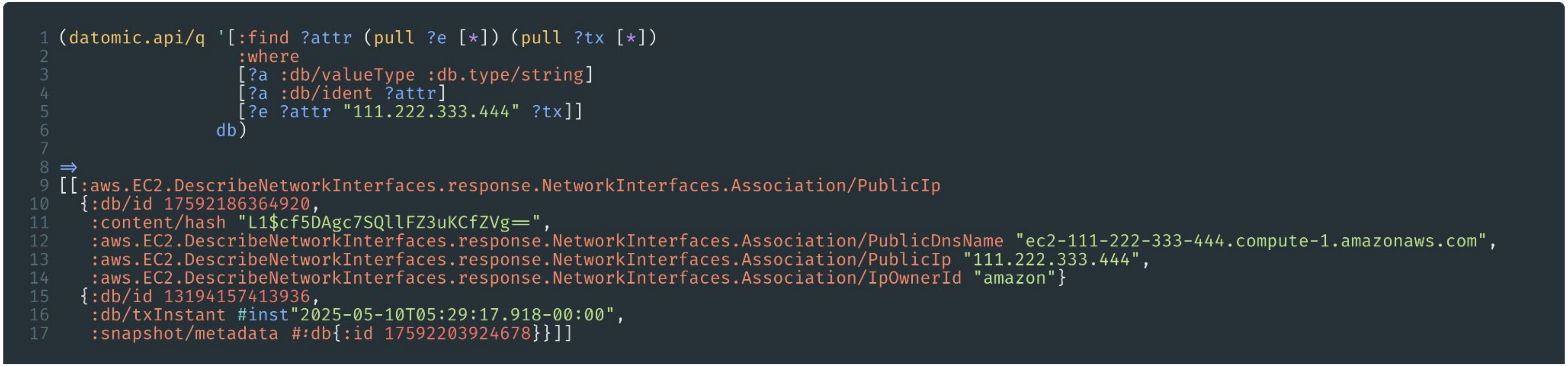


```
(datomic.api/q '[:find ?account-id ?account-name ?region ?instance-id
                       ?state-name ?monitoring-state ?image-id
                       ?vpc-id ?subnet-id ?private-dns ?private-ip ?public-dns ?public-ip ?http-tokens ?hop-limit
                       ?instance-profile-arn ?role-arn ?policy-name
                 :where
                 (ec2-instance ?account-id ?region ?instance-id
                               ?state-name ?monitoring-state ?image-id
                               ?vpc-id ?subnet-id ?private-dns ?private-ip ?public-dns ?public-ip ?http-tokens ?hop-limit
                               ?instance-profile-arn)
                 [(= ?http-tokens "optional")]
                 (ec2-instance-profile-role ?account-id ?instance-profile-arn ?role-name ?role-arn)
                 (role-policies ?account-id ?role-arn ?policy-name ?policy-document)
                 [?policy-document :aws.policy/Statement ?statement]
                 (iam-statement-allows-all-actions ?statement)
                 (aws-account-id->account-name ?account-id ?account-name)]
               db)
```

Beyond simple snapshots

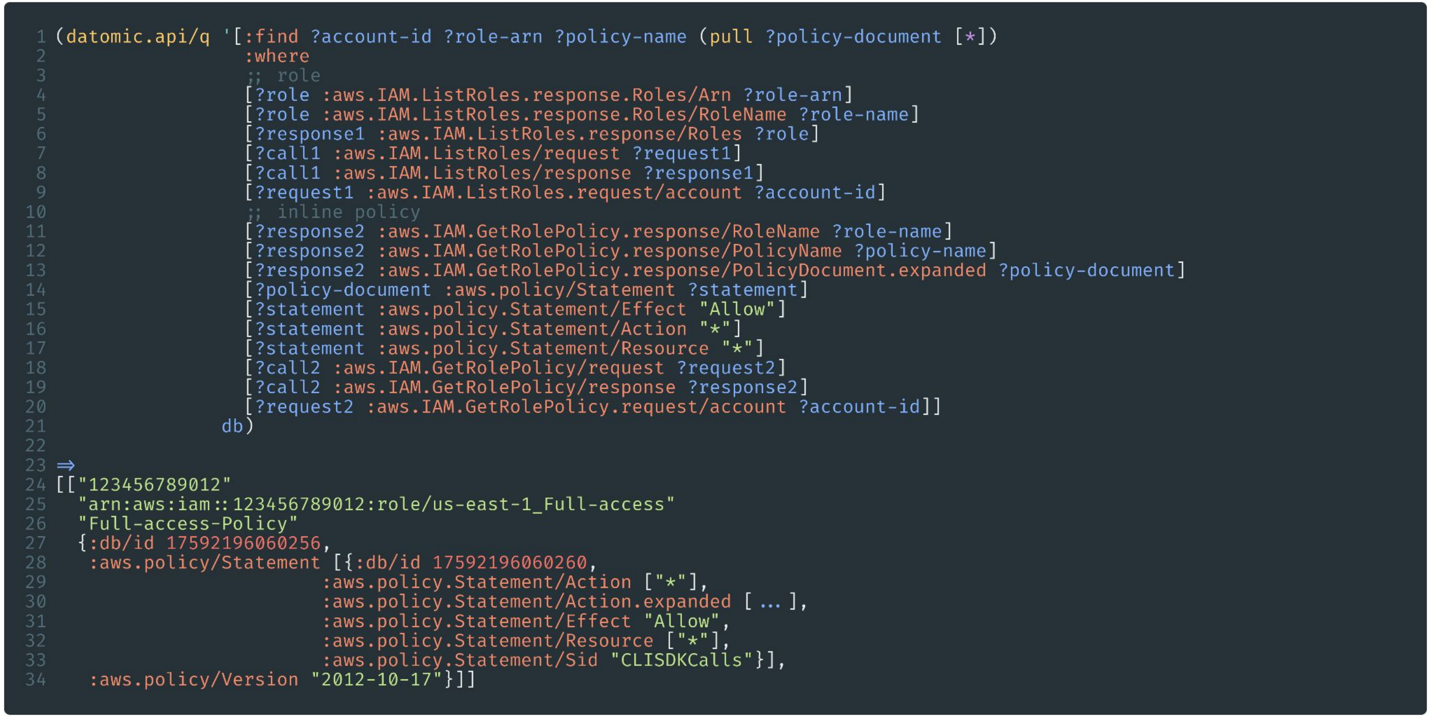


Finding the location of an IP across AWS



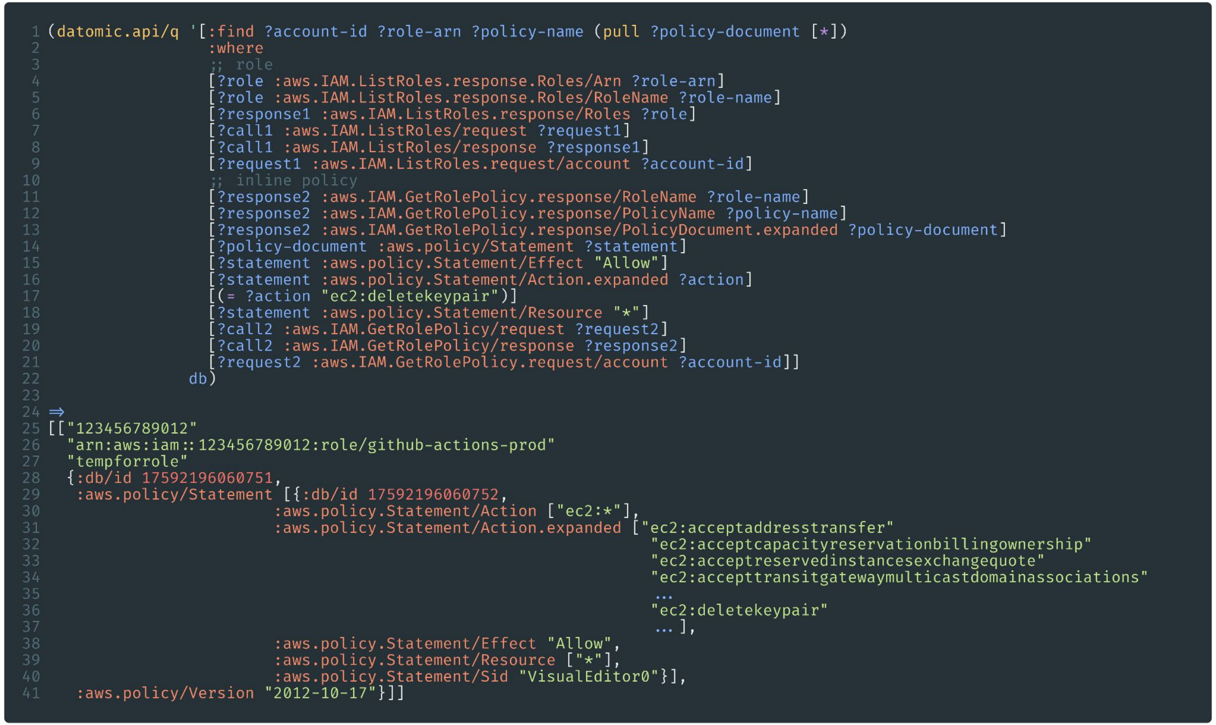
```
(datomic.api/q '[:find ?attr (pull ?e [*]) (pull ?tx [*])
                 :where
                 [?a :db/valueType :db.type/string]
                 [?a :db/ident ?attr]
                 [?e ?attr "111.222.333.444" ?tx]]
               db)

⇒
[[:aws.EC2.DescribeNetworkInterfaces.response.NetworkInterfaces.Association/PublicIp
  {:db/id 17592186364920,
   :content/hash "L1$cf5DAgc7SQllFZ3uKCfZVg==",
   :aws.EC2.DescribeNetworkInterfaces.response.NetworkInterfaces.Association/PublicDnsName "ec2-111-222-333-444.compute-1.amazonaws.com",
   :aws.EC2.DescribeNetworkInterfaces.response.NetworkInterfaces.Association/PublicIp "111.222.333.444",
   :aws.EC2.DescribeNetworkInterfaces.response.NetworkInterfaces.Association/IpOwnerId "amazon"}
  {:db/id 13194157413936,
   :db/txInstant #inst"2025-05-10T05:29:17.918-00:00",
   :snapshot/metadata #:db{:id 17592203924678}}]]
```

Identifying IAM permissions at scale



```
(datomic.api/q '[:find ?account-id ?role-arn ?policy-name (pull ?policy-document [*])
                 :where
                 ;; role
                 [?role :aws.IAM.ListRoles.response.Roles/Arn ?role-arn]
                 [?role :aws.IAM.ListRoles.response.Roles/RoleName ?role-name]
                 [?response1 :aws.IAM.ListRoles.response/Roles ?role]
                 [?call1 :aws.IAM.ListRoles/request ?request1]
                 [?call1 :aws.IAM.ListRoles/response ?response1]
                 [?request1 :aws.IAM.ListRoles.request/account ?account-id]
                 ;; inline policy
                 [?response2 :aws.IAM.GetRolePolicy.response/RoleName ?role-name]
                 [?response2 :aws.IAM.GetRolePolicy.response/PolicyName ?policy-name]
                 [?response2 :aws.IAM.GetRolePolicy.response/PolicyDocument.expanded ?policy-document]
                 [?policy-document :aws.policy/Statement ?statement]
                 [?statement :aws.policy.Statement/Effect "Allow"]
                 [?statement :aws.policy.Statement/Action "*"]
                 [?statement :aws.policy.Statement/Resource "*"]
                 [?call2 :aws.IAM.GetRolePolicy/request ?request2]
                 [?call2 :aws.IAM.GetRolePolicy/response ?response2]
                 [?request2 :aws.IAM.GetRolePolicy.request/account ?account-id]]
               db)

⇒
[["123456789012"
  "arn:aws:iam::123456789012:role/us-east-1_Full-access"
  "Full-access-Policy"
  {:db/id 17592196060256,
   :aws.policy/Statement [{:db/id 17592196060260,
                        :aws.policy.Statement/Action ["*"],
                        :aws.policy.Statement/Action.expanded [ … ],
                        :aws.policy.Statement/Effect "Allow",
                        :aws.policy.Statement/Resource ["*"],
                        :aws.policy.Statement/Sid "CLISDKCalls"}],
   :aws.policy/Version "2012-10-17"}]]
```

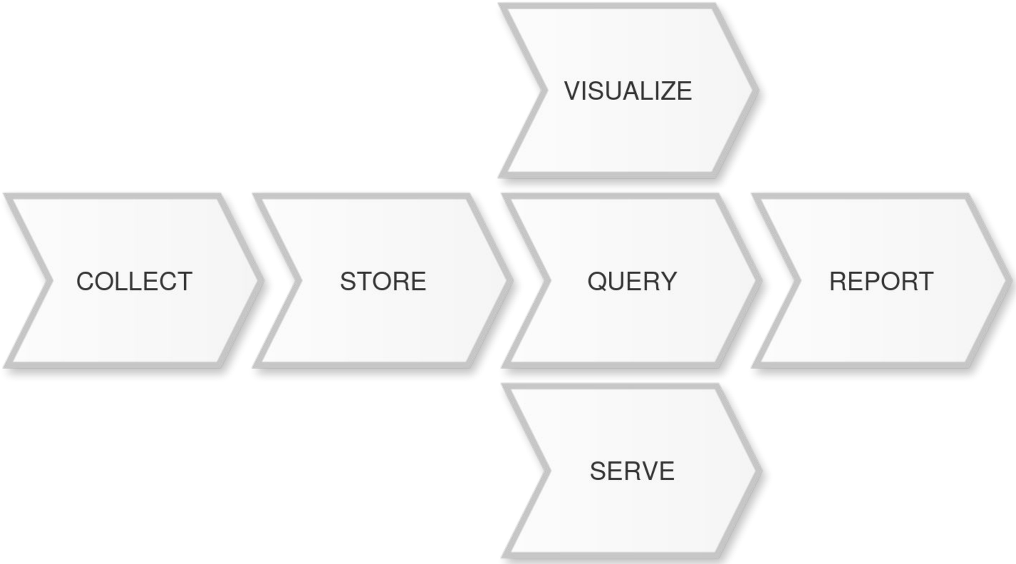
Expanding IAM permissions



```
(datomic.api/q '[:find ?account-id ?role-arn ?policy-name (pull ?policy-document [*])
                 :where
                 ;; role
                 [?role :aws.IAM.ListRoles.response.Roles/Arn ?role-arn]
                 [?role :aws.IAM.ListRoles.response.Roles/RoleName ?role-name]
                 [?response1 :aws.IAM.ListRoles.response/Roles ?role]
                 [?call1 :aws.IAM.ListRoles/request ?request1]
                 [?call1 :aws.IAM.ListRoles/response ?response1]
                 [?request1 :aws.IAM.ListRoles.request/account ?account-id]
                 ;; inline policy
                 [?response2 :aws.IAM.GetRolePolicy.response/RoleName ?role-name]
                 [?response2 :aws.IAM.GetRolePolicy.response/PolicyName ?policy-name]
                 [?response2 :aws.IAM.GetRolePolicy.response/PolicyDocument.expanded ?policy-document]
                 [?policy-document :aws.policy/Statement ?statement]
                 [?statement :aws.policy.Statement/Effect "Allow"]
                 [?statement :aws.policy.Statement/Action.expanded ?action]
                 [(= ?action "ec2:deletekeypair")]
                 [?statement :aws.policy.Statement/Resource "*"]
                 [?call2 :aws.IAM.GetRolePolicy/request ?request2]
                 [?call2 :aws.IAM.GetRolePolicy/response ?response2]
                 [?request2 :aws.IAM.GetRolePolicy.request/account ?account-id]]
               db)

⇒
[["123456789012"
  "arn:aws:iam::123456789012:role/github-actions-prod"
  "tempforrole"
  {:db/id 17592196060751,
   :aws.policy/Statement [{:db/id 17592196060752,
                        :aws.policy.Statement/Action ["ec2:*"],
                        :aws.policy.Statement/Action.expanded ["ec2:acceptaddresstransfer"
                                                                "ec2:acceptcapacityreservationbillingownership"
                                                                "ec2:acceptreservedinstancesexchangequote"
                                                                "ec2:accepttransitgatewaymulticastdomainassociations"
                                                                ...
                                                                "ec2:deletekeypair"
                                                                ... ],
                        :aws.policy.Statement/Effect "Allow",
                        :aws.policy.Statement/Resource ["*"],
                        :aws.policy.Statement/Sid "VisualEditor0"}],
   :aws.policy/Version "2012-10-17"}]]
```

https://github.com/iann0036/iam-dataset

Replik8s

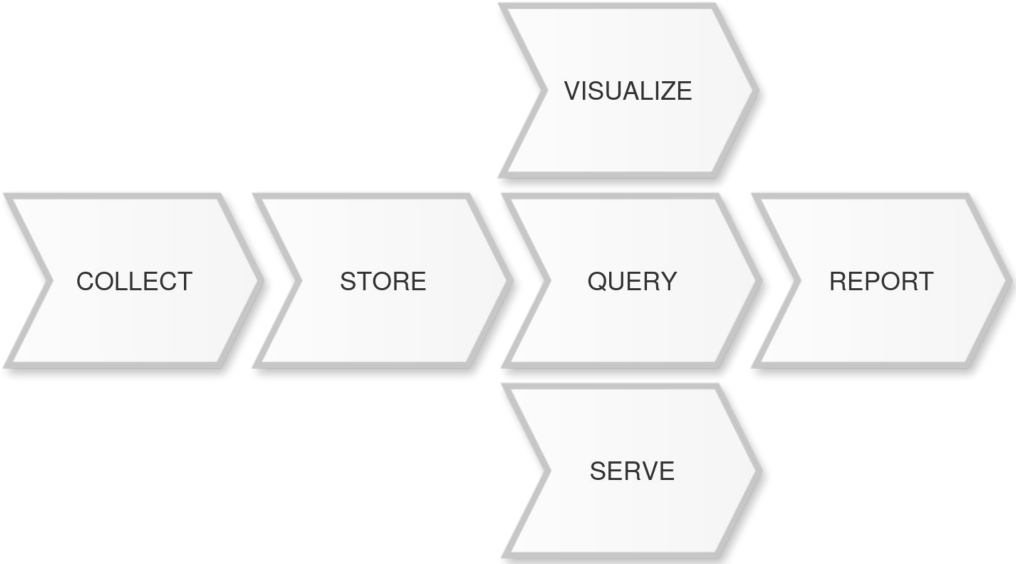


Replik8s

▷

DEMO

Replik8s



VISUALIZE

COLLECT

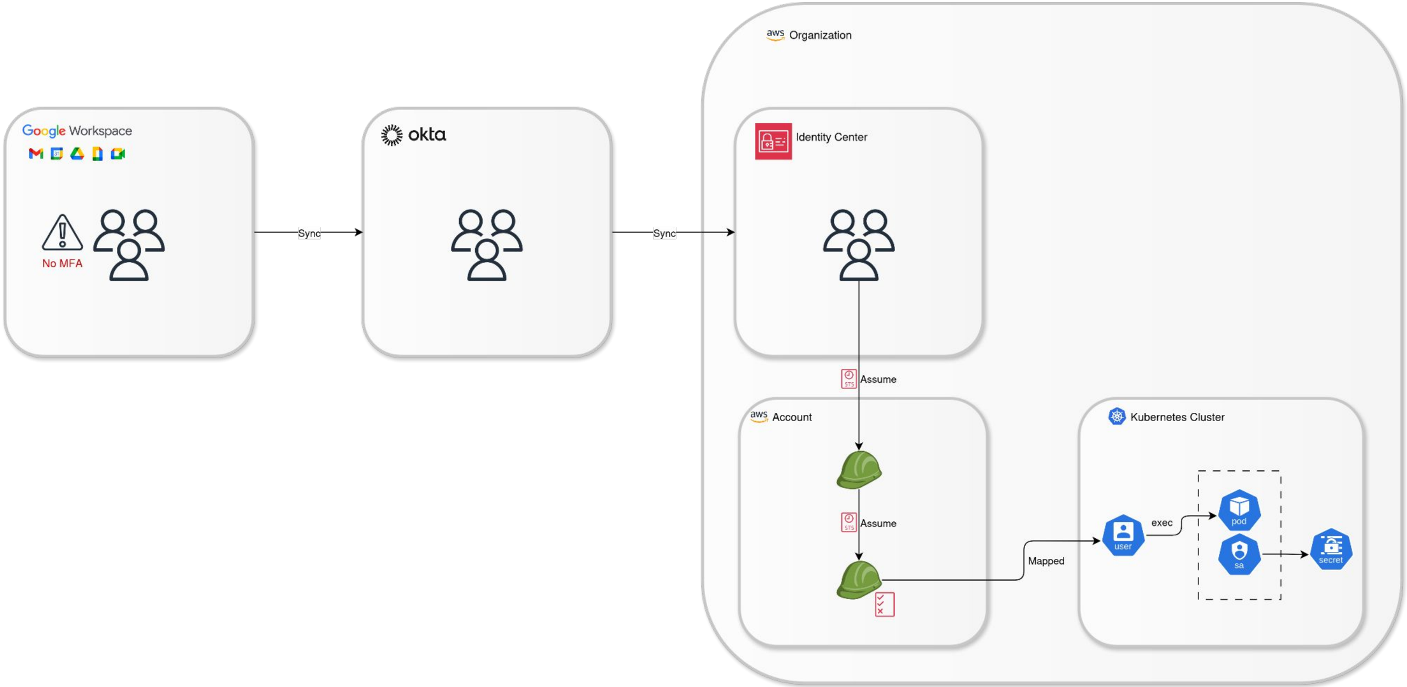
STORE

QUERY

REPORT

SERVE

Treating providers as equals



Going Forward

Further reading:

- https://www.latacora.com/blog/2023/11/01/our-approach-to-building-security-tooling/
- https://www.latacora.com/blog/2024/09/13/datomic-and-content-addressable-techniques/

Thank you!

We ask for **your feedback!**

PLEASE VOTE NOW!

★★★★★

devmio