ORY / summit-22

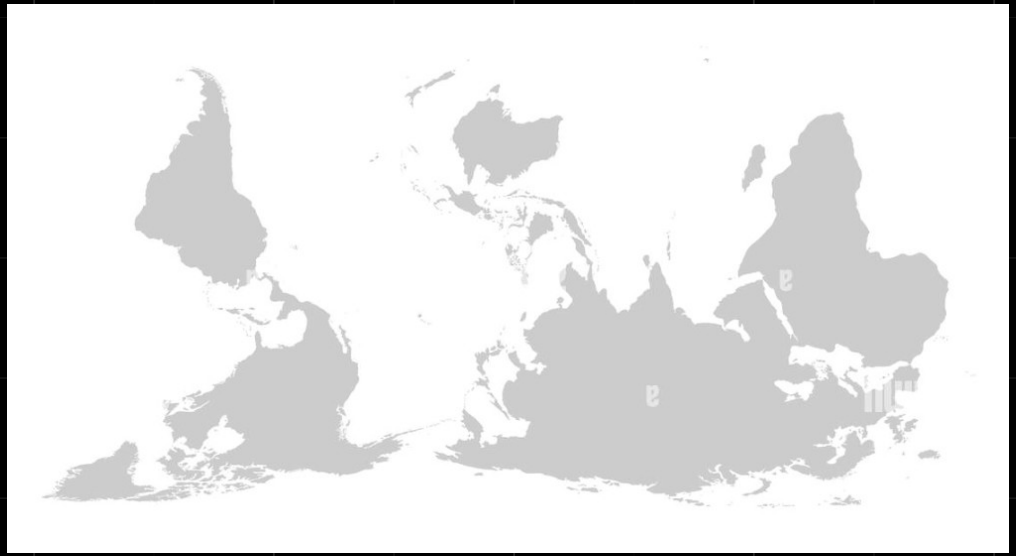
Mal Curtis

mal@mal.co.nz

ORY in the inMusic Cloud

20 - 10 - 2022

@ Ory Summit 2022



inMusic

HOME OF THE WORLD'S PREMIER MUSIC AND AUDIO TECHNOLOGY BRANDS

DENON DJ


Browse
Search
1A Blackallit (Original Mix)
1B Darkside
DENON DJ
SC6000
CUE
SHIFT
HOT CUE
LOOP
ROLL
SLICER


DENON DJ
PRIME 4

Numark


Numark
MIXSTREAM PRO
SYNC
CUE
engine os
SMART LIGHTING CONTROL

HEADRUSH®


SEARING LEAD
JRC-OD
FILTER
DOUBLER
STOMP/RIG
LOOPER
TUNER
HEADRUSH


HEADRUSH
PEDALBOARD
TANGERINE DREAM
RIG ▲
VOLUME
PARA EQ
GRAY C
JRC-OD
LOOPER
RIG ▼
WHAM
FLANGER
8-BIT
PHASER
TEMPO
HEADRUSH

AKAI PROFESSIONAL

BFD


File ▾ Tools ▾ Help ▾
Expansion
BFD3
Presets
Dashboard
Emre Hiphop 1
Gain
Presets Kits Drums Grooves Auto
Kit Effects Groove Editor Key Map
Tech Model
Audience Drummer Select Link
OH OH
Fav Kit Mix Grv Key Auto Glo
Library
Include when loading...
JM BFD3 109 Pop
JM BFD3 110 Pop
JM BFD3 111 Funk
JM BFD3 112 Funk
JM BFD3 112 Swing Jazz
JM BFD3 119 Blues
JM BFD3 119 Blues 2
JM BFD3 119 Rock
JM BFD3 146 Jazz
JM BFD3 160 Rock
Options
Export View ▾
Faders Effects Sends Tweaks
Mini Mixer
Kick Kick 2 Snare Snare 2 Hihat Floor Tom Mid Tom High Tom Toms Crash 1 Cymbal 1 Ride 1 Perc AmbMix SideChain
DW Mardi Gras Sparkle Kick 1
General
Pan
Tuning
Bleed
Loudness
Ambient Mics
Flip LR
Link

Overview

The inMusic Cloud

What ORY services inMusic uses

How we use Hydra

How we use Oathkeeper

How we use Keto

Questions

The inMusic Cloud

Product Purchases and Registrations

Software Authorization / Unlock

Software / Firmware updates

Brand Specific Profiles and “Clouds”

- Headrush Cloud
- Engine Cloud

The inMusic Cloud

Homogenous REST and GraphQL API

Authenticated via inMusic Profile (Hydra + Magento)

Heterogenous service based architecture

Go, Python, Javascript, PHP

Provided to internal teams as a service.

The inMusic Cloud

Monorepo. Docker. Gitlab. AWS ECS. Terraform.

55 services at last count

Contract driven: RPC services defined with Protocol Buffers

The inMusic Cloud

Protocol Buffer RPC definitions are the source of truth

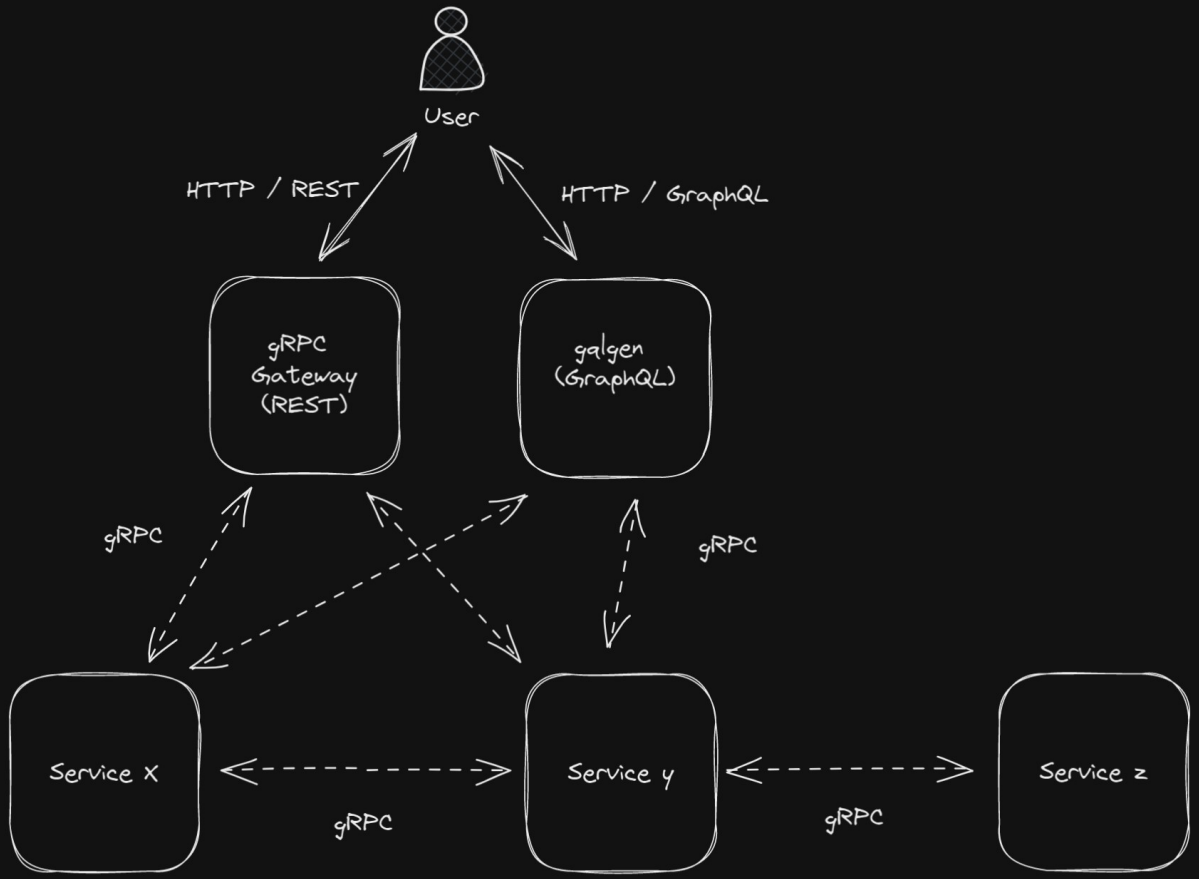
REST via grpc HTTP Gateway protoc-gen-grpc-gateway

GraphQL via 99designs gqlgen protoc-gen-gogql

Validation via Envoy protoc-gen-validate

Documentation Site

- docusaurus
- GraphQL Explorer - GraphiQL
- REST Explorer Stoplight Elements
- protoc-gen-openapiv2


User
HTTP / REST
HTTP / GraphQL
gRPC Gateway (REST)
galgen (GraphQL)
gRPC
gRPC
Service X
Service y
Service z
gRPC
gRPC

List Connectable Accounts

GET https://api.inmusicbrands.com/v1/connectable_accounts/{namespace}

List a set of account providers for a specific namespace.

Request

Security: OAuth 2.0

Authorization Code OAuth Flow

Authorize URL: https://auth.profile.inmusicbrands.com/oauth2/auth

Token URL: https://auth.profile.inmusicbrands.com/oauth2/token

Path Parameters

namespace string required

Allowed values: NAMESPACE_INMUSIC_PROFILE, NAMESPACE_INMUSIC_SOFTWARE_INSTALLER, NAMESPACE_ENGINE, NAMESPACE_HEADRUSH

Responses

200 default

A successful response.

Body

accounts array[object] required

- provider string
  Allowed values: ACCOUNT_PROVIDER_DROPBOX, ACCOUNT_PROVIDER_BEATPORT, ACCOUNT_PROVIDER_BEATSOURCE, ACCOUNT_PROVIDER_SOUNDCLOUD, ACCOUNT_PROVIDER_TIDAL, ACCOUNT_PROVIDER_AMAZON_MUSIC
- name string required
  Example: Dropbox
- is_connected boolean required
- initiate_connection_url string<url> required
  Example: https://api.inmusicbrands.com/connected_accounts/dropbox/initiate

Auth

Token : Bearer ECB1kJrP6orETlM7o1-Nq

Parameters

namespace* : NAMESPACE_ENGINE

Send API Request

Request Sample: Shell / cURL

```
curl --request GET \
  --url https://api.inmusicbrands.c
  --header 'Authorization: Bearer E
  --header 'Content-Type: applicati
```

Response Example

```
{
  "accounts": [
    {
      "provider": "ACCOUNT_PRO
      "name": "Dropbox",
      "is_connected": true,
      "initiate_connection_url
    }
  ]
}
```

```
// devices that need to be authenticated against a servic
// Accounts are namespaced so that each namespace can hav
// linked account for each Account Provider. This allows
// account for say, Akai Pro, than you have for Engine.
service ConnectedAccountsService {
  option (graphql.v1.svc) = {
    type : QUERY
    name : ""
    upstream : UPSTREAM_CLIENT
  };

  // List Connectable Accounts
  //
  // List a set of account providers for a specific names
  rpc ListConnectableAccounts(ListConnectableAccountsRequ
      returns (ListConnectableAccountsResponse) {
    option (google.api.http) = {
      get : "/v1/connectable_accounts/{namespace}"
    };
    option (grpc.gateway.protoc_gen_openapiv2.options.ope
      tags : "Connected Accounts 🔑"
    };
  }

  // Get Account Credentials
  //
  // Returns current credentials for the supplied namesp
  // When requesting credentials that are expired, they
  // the request.
  rpc GetAccountCredentials(GetAccountCredentialsRequest)
      returns (GetAccountCredentialsResponse) {
    option (google.api.http) = {
      get : "/v1/connectable_accounts/{namespace}/{provid
    };
    option (grpc.gateway.protoc_gen_openapiv2.options.ope
      tags : "Connected Accounts 🔑"
    };
  }

  // Disconnected Account
  //
  // Deletes account credentials and marks the account ty
  rpc DisconnectAccount(DisconnectAccountRequest)
      returns (DisconnectAccountResponse) {
    option (google.api.http) = {
      delete : "/v1/connectable_accounts/{namespace}/{pro
    };
    option (graphql.v1.rpc) = {
      type : MUTATION
    };
    option (grpc.gateway.protoc_gen_openapiv2.options.ope
      tags : "Connected Accounts 🔑"
    };
  }
```

BETA inMusic Webservices Cloud APIs Explorers Websites

GraphiQL ▶ Prettify Merge Copy History

```
# Welcome to GraphiQL
#
# GraphiQL is an in-browser tool for
# testing GraphQL queries.
#
# Type queries into this side of the
# typeaheads aware of the current Gr
# validation errors highlighted with
#
# GraphiQL queries typically start wi
# with a # are ignored.
#
# An example GraphQL query might loc
#
#     {
#       field(arg: "value") {
#         subField
#       }
#     }
#
# Keyboard shortcuts:
#
#  Prettify Query:  Shift-Ctrl-P (or
#
#      Merge Query:  Shift-Ctrl-M (or
#
#        Run Query:  Ctrl-Enter (or p
#
#    Auto Complete:  Ctrl-Space (or
#
query {
```

QUERY VARIABLES

REQUEST HEADERS

What ORY services?

Hydra

Backed by Magento

Custom Device Auth implementation

Oathkeeper

- Public API only
- Authed by Hydra OAuth2 Introspection
- JWT for Internal Services

Keto

User Groups

Service based authorization

How We Use Hydra

~~How~~ Why we use Hydra

inMusic Profile due to be released - Magento

Needed "Desktop Login" via inMusic Profile for BFD Player

No Session API existed

Existing user profiles merged from many brands

Hydra was best fit

- Portable: not tied to Magento
- Cloud Native, OSS, Standards Based (OAuth2)
- Not trying to do too much

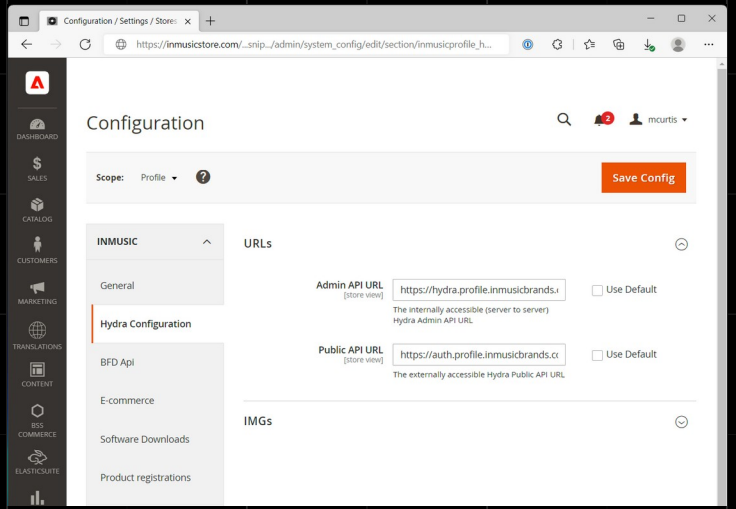
How We Use Hydra

Custom Magento Module

Copied Login Controller and Views

Created Consent Controller

Managed via Admin Settings

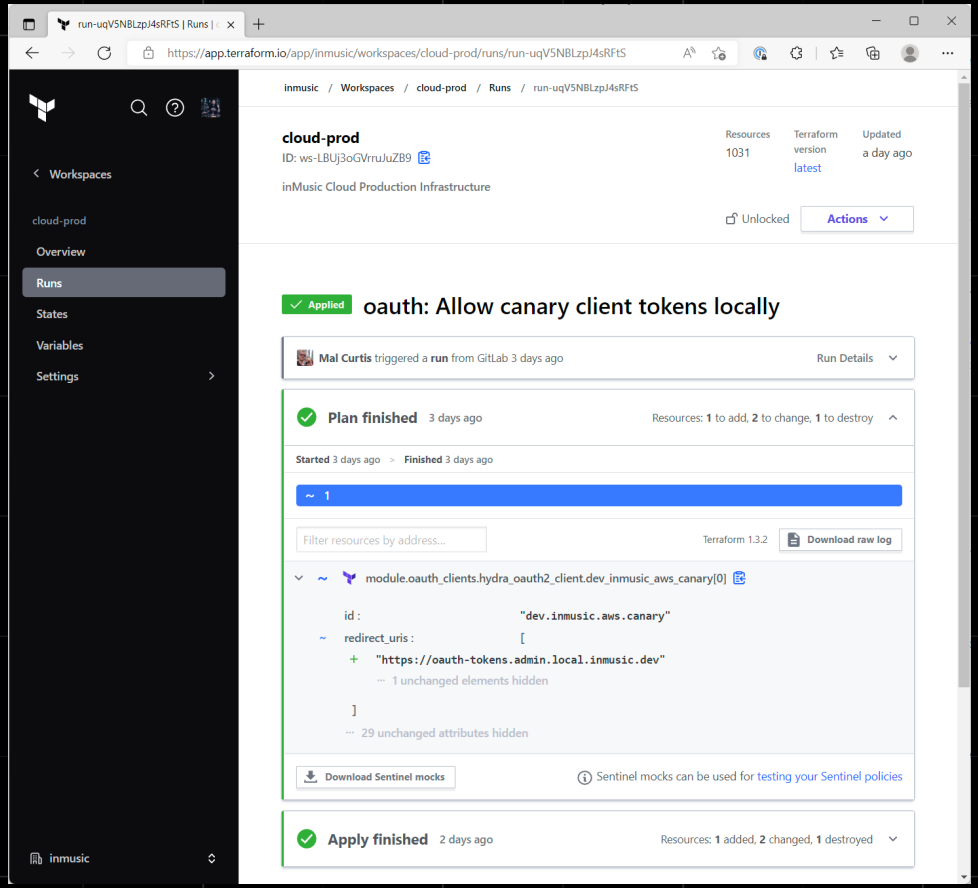


How We Use Hydra

Clients Managed via Terraform

```
resource "hydra_oauth2_client" "dev_inmusic_aws_canary" {
  client_id   = "dev.inmusic.aws.canary"
  client_name = "AWS Canary Client"

  grant_types = ["authorization_code", "refresh_token"]
  redirect_uris = [
    "https://oauth-tokens.admin.local.inmusic.dev",
    "http://127.0.0.1:5555/callback",
  ]
  response_types              = ["code"]
  scopes                      = ["email", "openid", "offline"]
  token_endpoint_auth_method = "none"
  lifecycle {
    ignore_changes = [
      client_secret
    ]
  }
}
```



How We Use Hydra

Custom RFC 8628 Device Auth Implementation

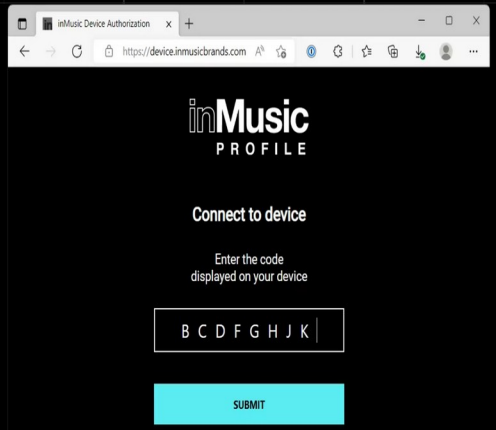
Adds Device Authorization API Endpoint

Handles "device code" Grant Type

Proxies other token requests

"Sidequests" force UI flows

"Brand Aware" UI



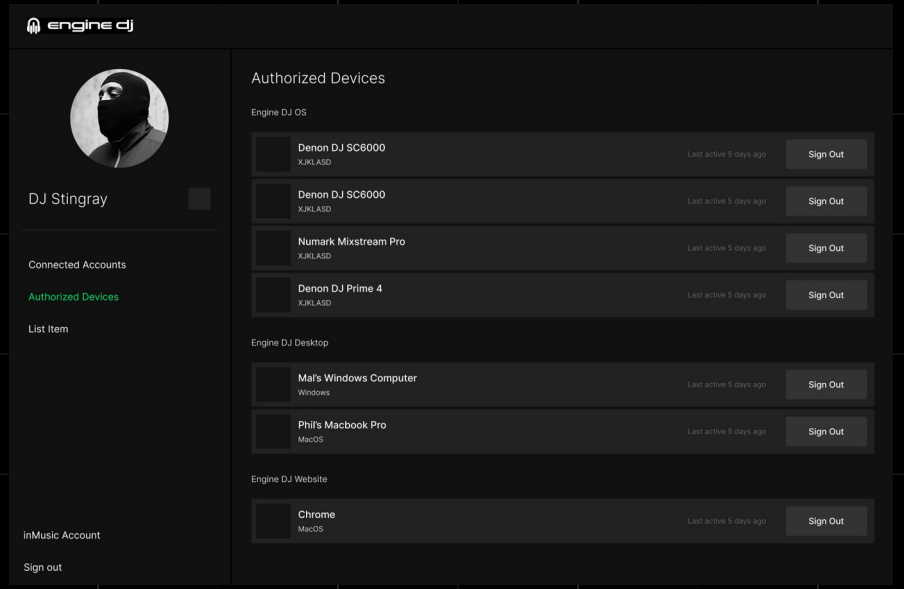
How We Use Hydra

Authorized Devices

Associates user agent data to consents in Hydra

A bit of a hack!

Provides remote logout



How We Use Oathkeeper

How We Use Oathkeeper

Traefik "forward auth" middleware

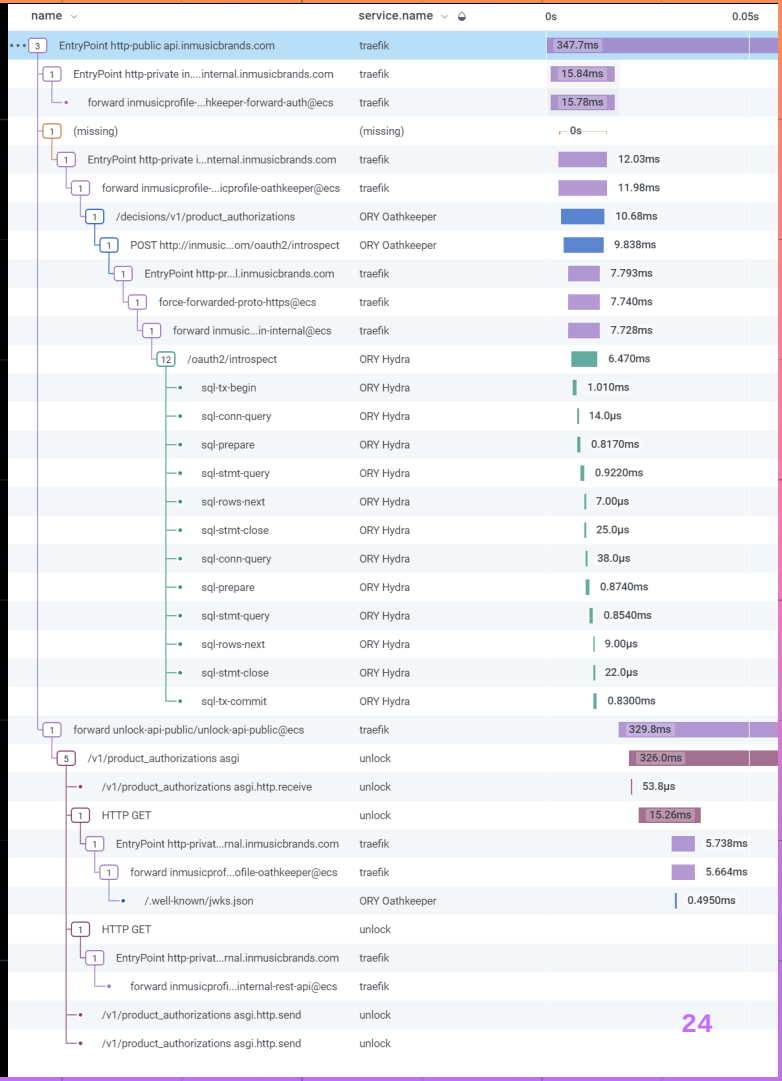
Forwards Request to Oathkeeper

Hydra Introspects OAuth2 token

Oathkeeper Writes JWT on success

Traefik forwards request w/ JWT

Service Validates JWT



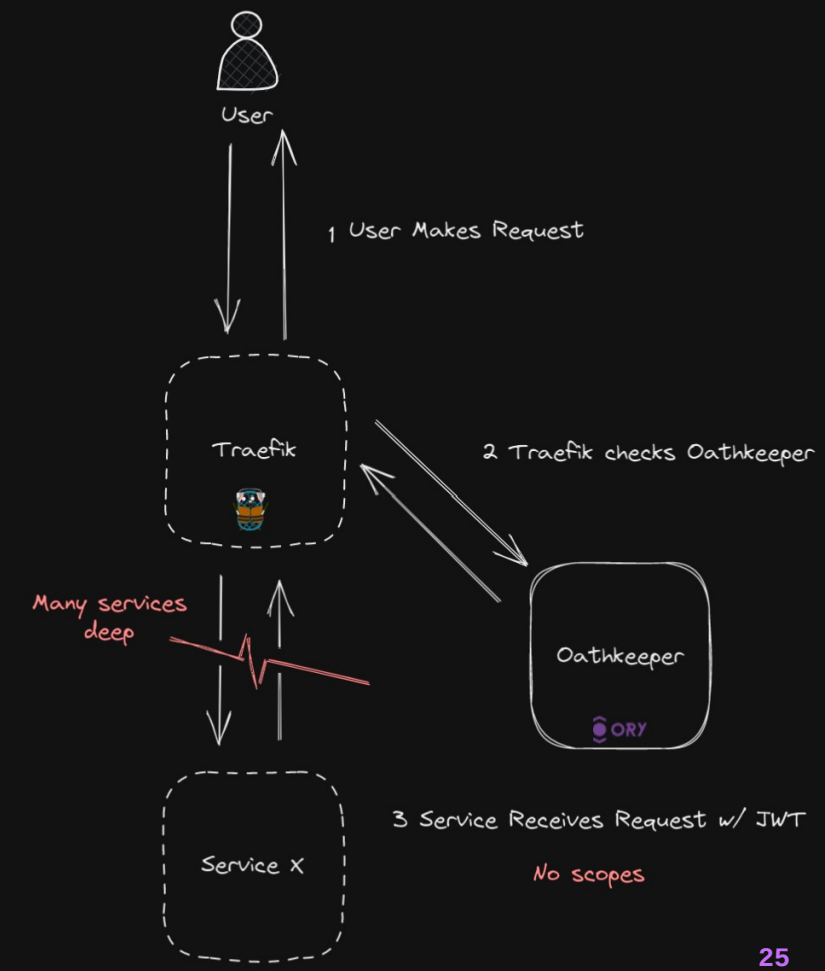
Oathkeeper Problem

How to check scopes for GraphQL requests?

Single request may aggregate many different services

Services want to define their own scope requirements

Oathkeeper didn't assign scopes to JWTs



Oathkeeper Problem

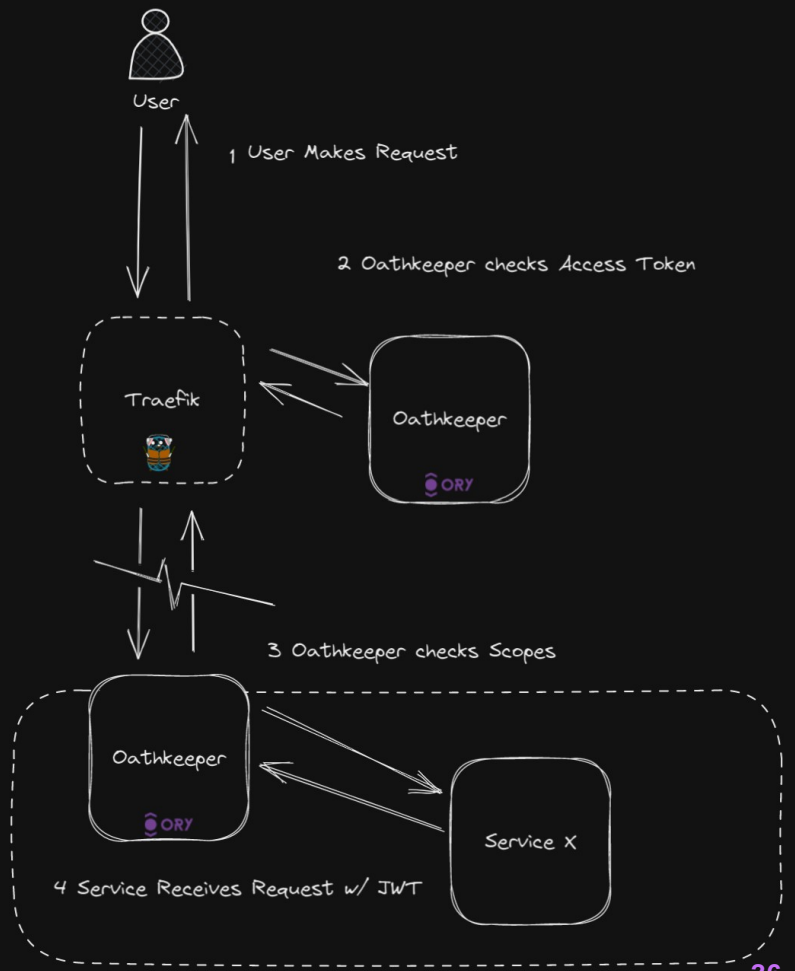
HTTP & GraphQL gateways "swallow" scopes

GraphQL especially hard to enforce at edge

Recently added Scopes to Oathkeeper JWT Claims

Plan to add Oathkeeper sidecar to enforce scopes

Remove service JWT check



How We Use Keto

How We Use Keto

Only recently spun up

User Groups

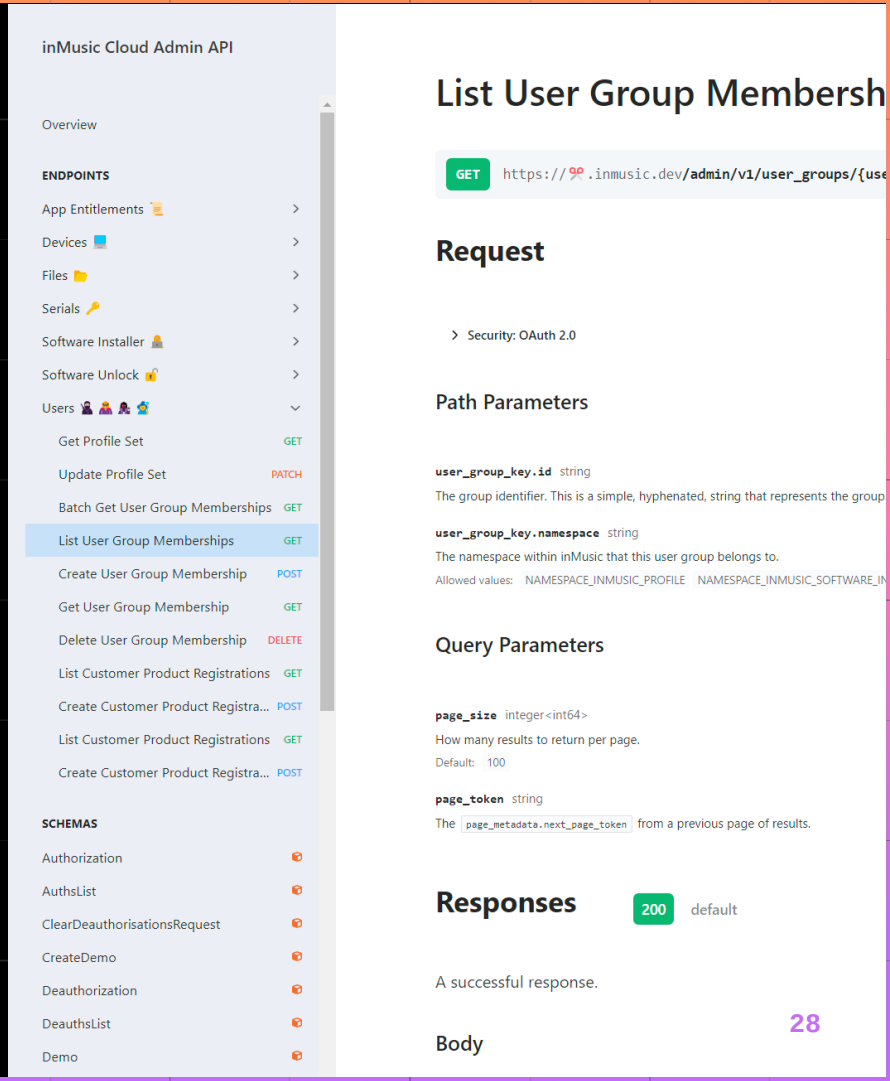
```
usergroup:engine/beta-testers#member@<user_id>

message UserGroupMembership {
  inmusicapi.v1.Namespace namespace = 1;
  string user_group = 2;
  string user_id = 3;
}
```

Service Permissions

```
software-installer/version:<sku>/<platform>/<version>#access@<user_id>
arkaos/lighting-stream:<owner_id>/<stream_id>#access@<user_id>

message DomainObjectAccess {
  inmusicapi.v1.Domain domain = 1;
  string object = 2;
  string user_id = 3;
}
```



How We Use Keto

"Permissions" service

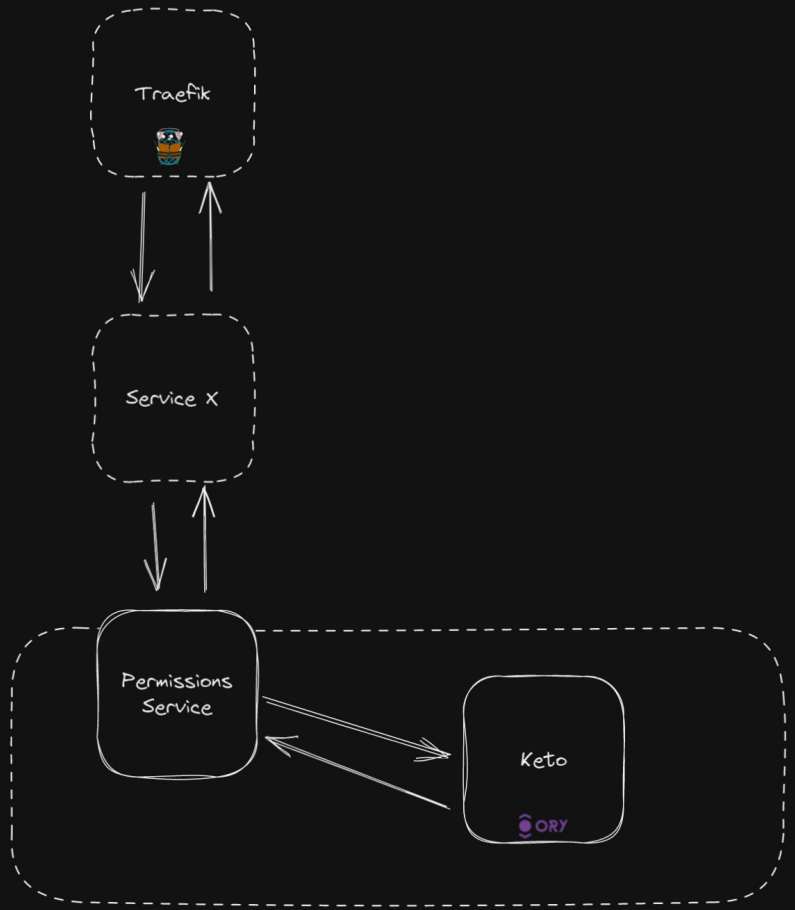
abstracts Keto away

enforces "inMusic" concepts

Keto runs as a sidecar

Not exposed directly to services

Managing Migrations TBD



ORY / summit-22

Mal Curtis

mal@mal.co.nz

Questions?

Hit me up on the ORY Slack

20 - 10 - 2022

@ Ory Summit 2022