Identifying & Reducing Permission Explosion in AWS

By Pankaj Moolrajani
Security Engineering Lead @Motive

pmoolrajani@gmail.com

@p_moolrajani

I ❤️

Creating Software

Indian Food

Walking

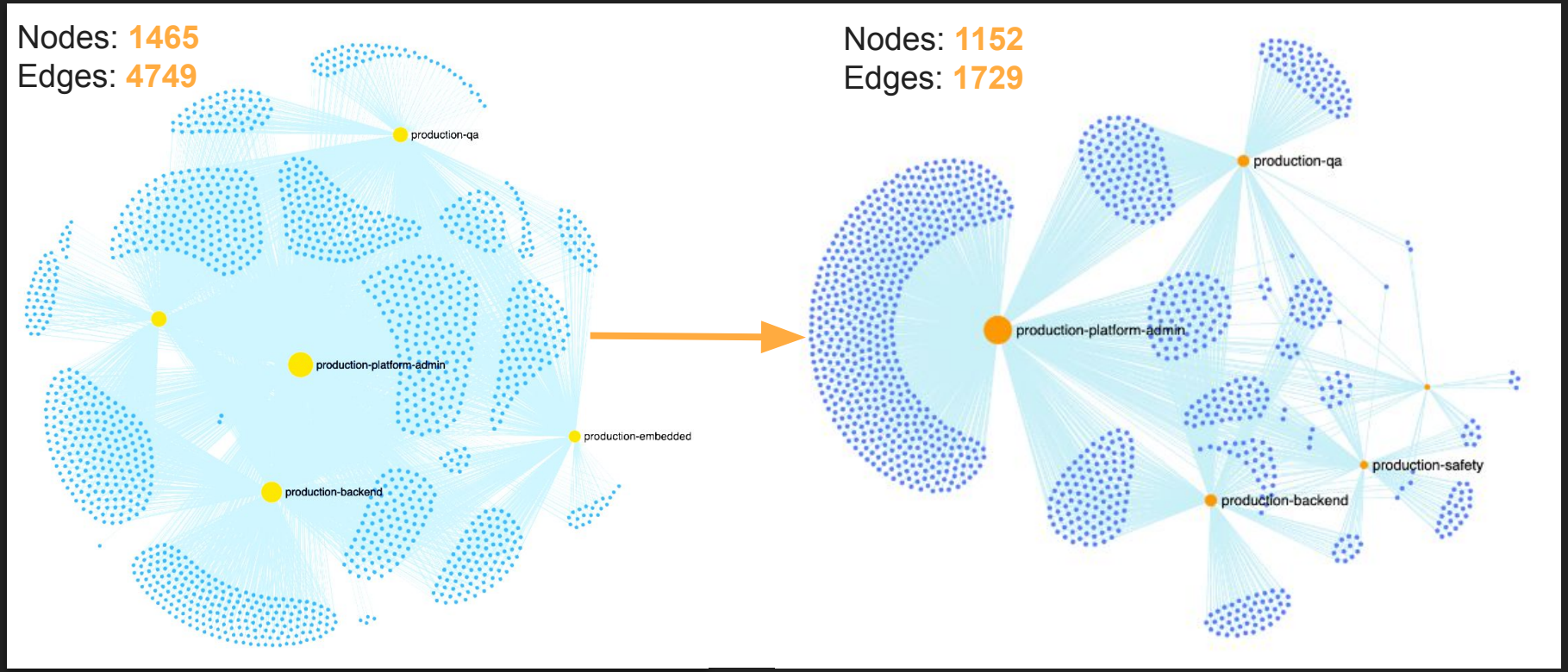
What are we learning today about Permission Explosion?

How to IDENTIFY?

How to FIX?

How to KEEP IT AWAY?

Sneak Peak


Nodes: 1465
Edges: 4749
production-qa
production-platform-admin
production-embedded
production-backend
Nodes: 1152
Edges: 1729
production-qa
production-platform-admin
production-safety
production-backend

Basics

What is a Permission?

A user’s right to perform an action on a resource

eg: Bob with Developer role is allowed to DeleteDBInstance - rds:fleet-db

What is Permission Explosion?

User’s having more permissions than they need for their job

Why does it occur in AWS?

Reason 1 - Permission Creep

User roles change over time, granting more access while rarely revoking unnecessary permissions.

Reason 2 - Temporary Access

Users request new permissions for ad-hoc tasks,
and when granted,
they become permanent for all users in that role.

Reason 3 - Easy to Grant Broad Access

Few broad roles with fixed permissions simplify management, but compromise fine-grained control.

Mathematics of Permission Explosion

Gaining Clarity in Chaos

Permission Utilization Ratio (PUR)

Represents TRUE utilization ratio of a permission in a role.

Permission Utilization Ratio (PUR)

PUR of a permission in a role can be determined by using <u>frequency of use</u> and the <u>number of users</u> who utilize it

Permission Utilization Ratio (PUR)

$$PU = \frac{\text{median num days used by users} \times \text{num users used}}{\text{total num days} \times \text{total num users}}$$

Under-Utilized Permission Ratio (UPR)

Proportion of the permissions within a role that are rarely or never used

Under-Utilized Permission Ratio (UPR)

$$UPR = \frac{1 - \text{Sum of } (PUR's)}{\text{Num of Permissions}}$$

Calculate UPR of a Permission in a Role

```
# Role - piam_subteam_platform_security
# Resource - s3:rnd-bucket
# Action - PutObject
# Permission - s3:rnd-bucket-PutObject

# Input data
median_days_used_by_users = 150
num_users_used = 10
total_num_days = 365
total_num_users = 42

# Calculate Permission Utilization Ratio (PUR)
pur = (median_days_used_by_users * num_users_used) / (total_num_days *
total_num_users)

# Calculate Under Utilized Permission Ratio
upr = 1 - permission_usage_ratio

# Output
print("PUR:", pur)
print("UPR:", upr)
```

```
PUR: 0.1
UPR: 0.9
```

AWS Setup

AWS IAM Roles

production-qa

production-safety

production-backend

production-embedded

production-platform-admin

AWS Resource Types

ECS

EC2

Route 53

Cloud Front

Secrets Manager

s3

RDS

Backup

Tools

Tools

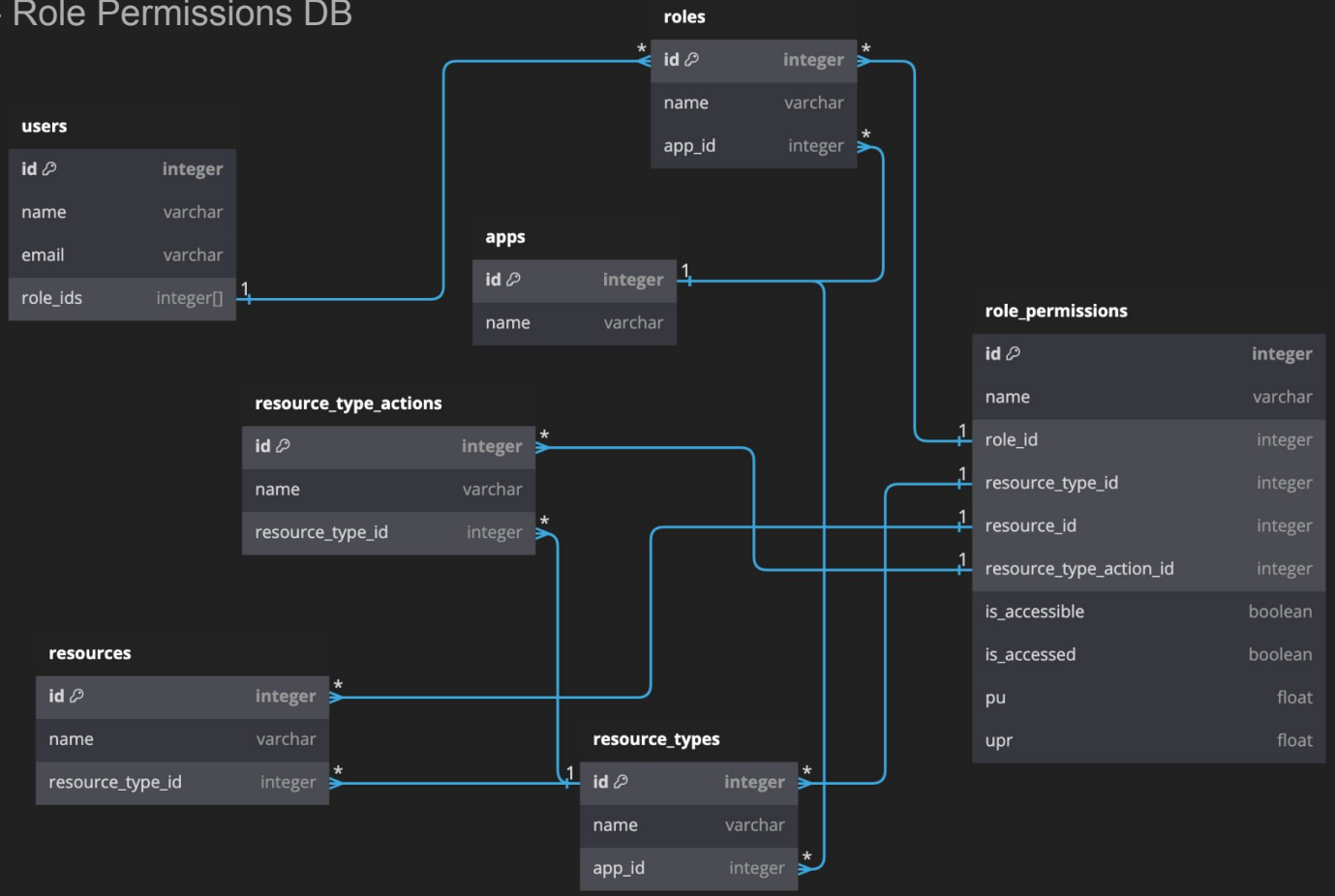
Python

Google Colab

SigmaJS

Sqlite Database

Data Schema - Role Permissions DB



Sample Graph (ipysigma)

piam-split-roles.ipynb

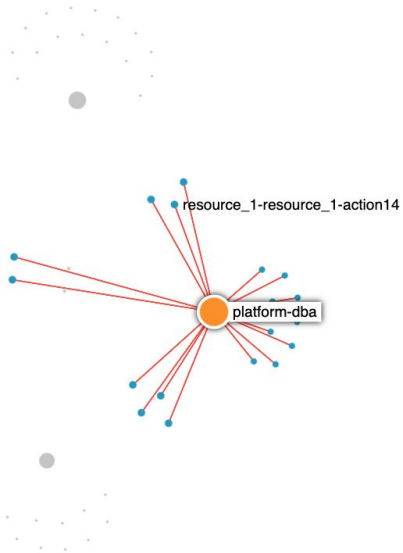
File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text Reconnect

```
graph = get_graph(splitted_rp, "splitted_rp.gexf")
graph
```

Undirected Graph
79 nodes
94 edges



platform-dba

legend · info hide

Node *platform-dba*

From kwargs:
node_size 18

Attributes:
node_type role
louvain 4

Known viz data:
color #FF9900
label platform-dba
x 87.28271484375
y -3.9546258449554443

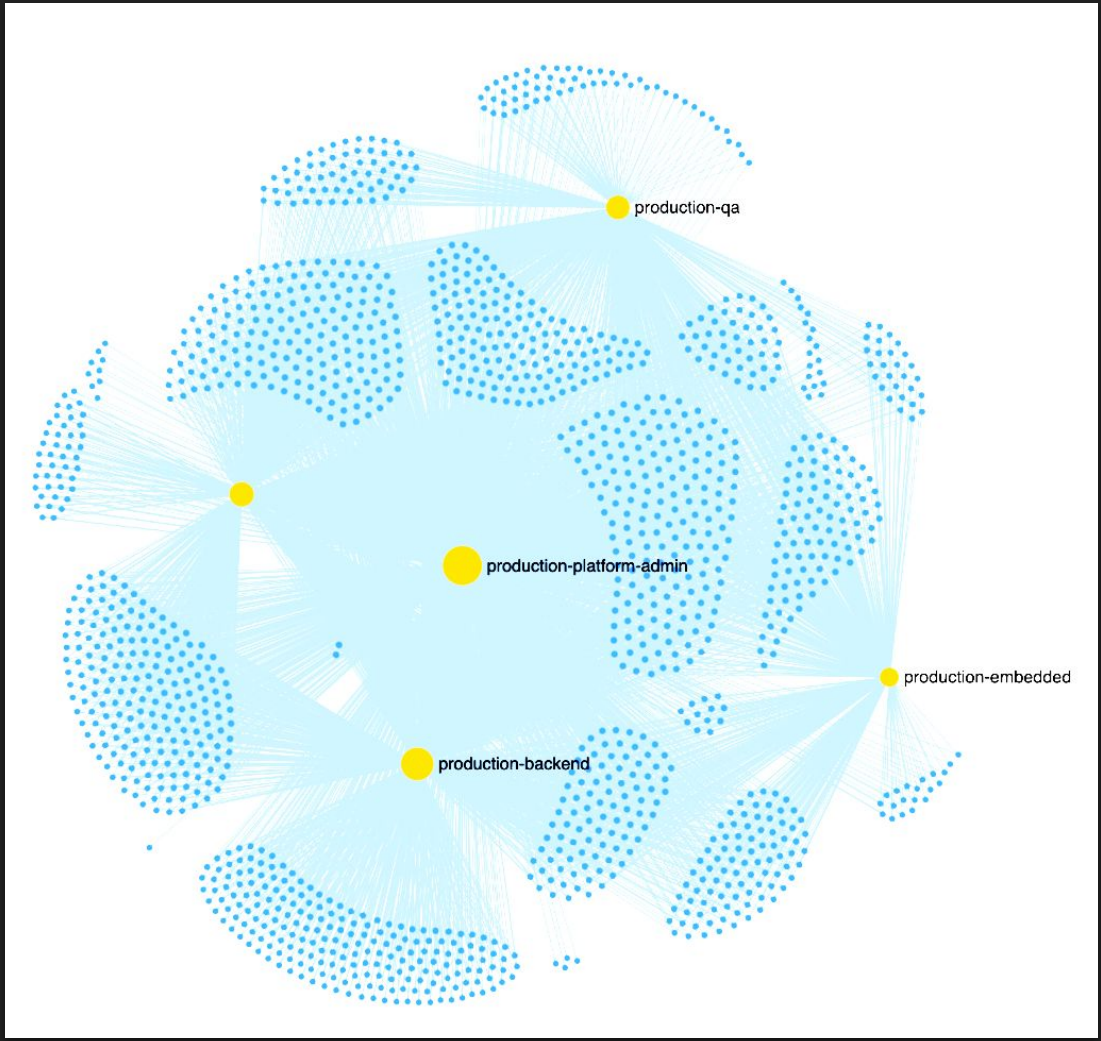
Computed metrics:
degree 18

png svg gexf json

How to IDENTIFY?

Starting Point

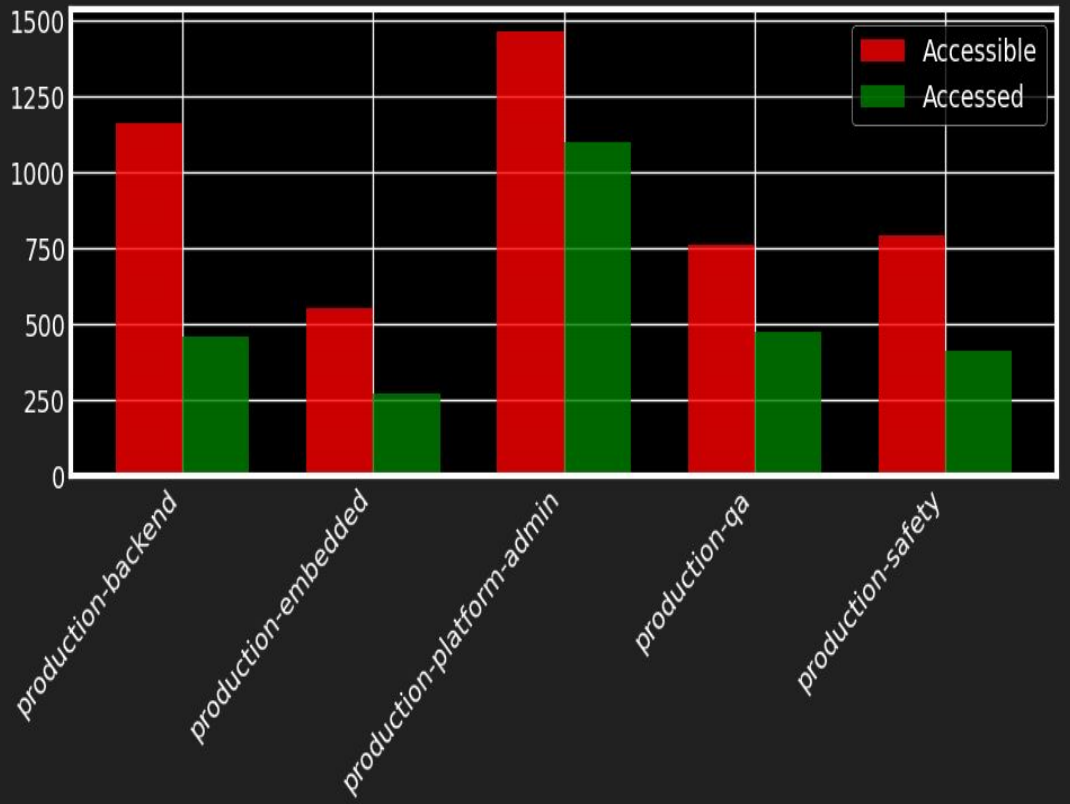
Permission Explosion



Accessible vs Accessed Permissions in a Role

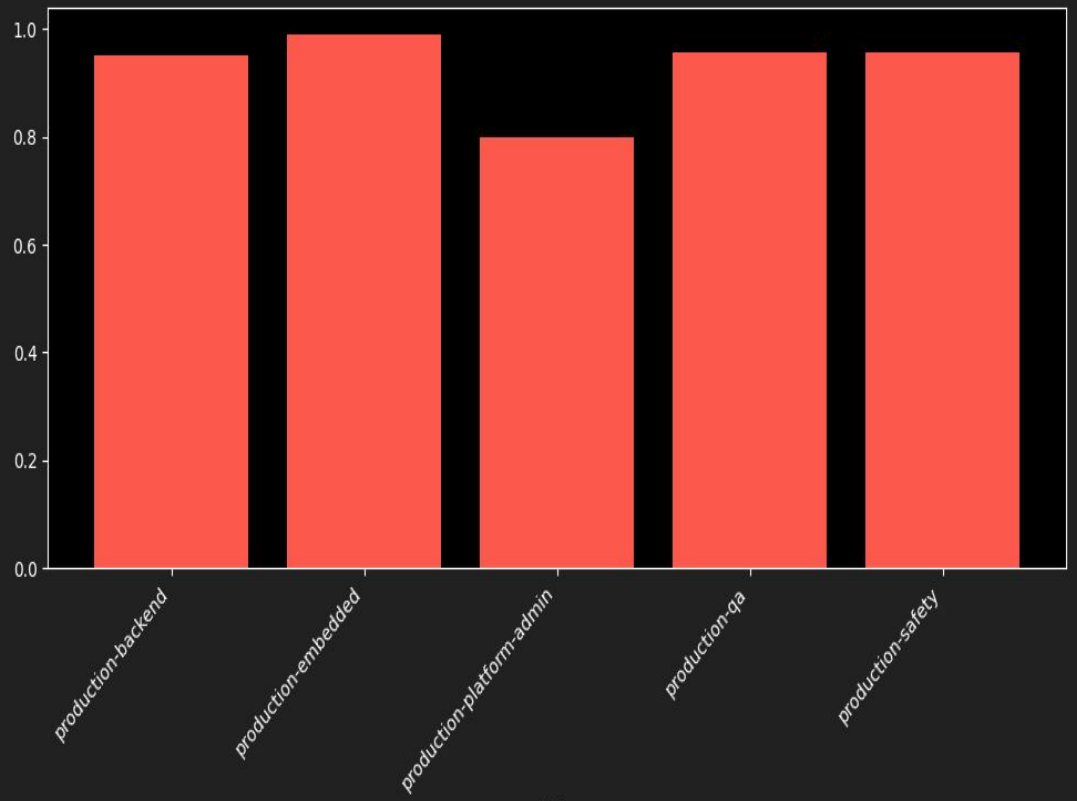
Permissions Unused

50%



Under-Utilized Permission Ratio

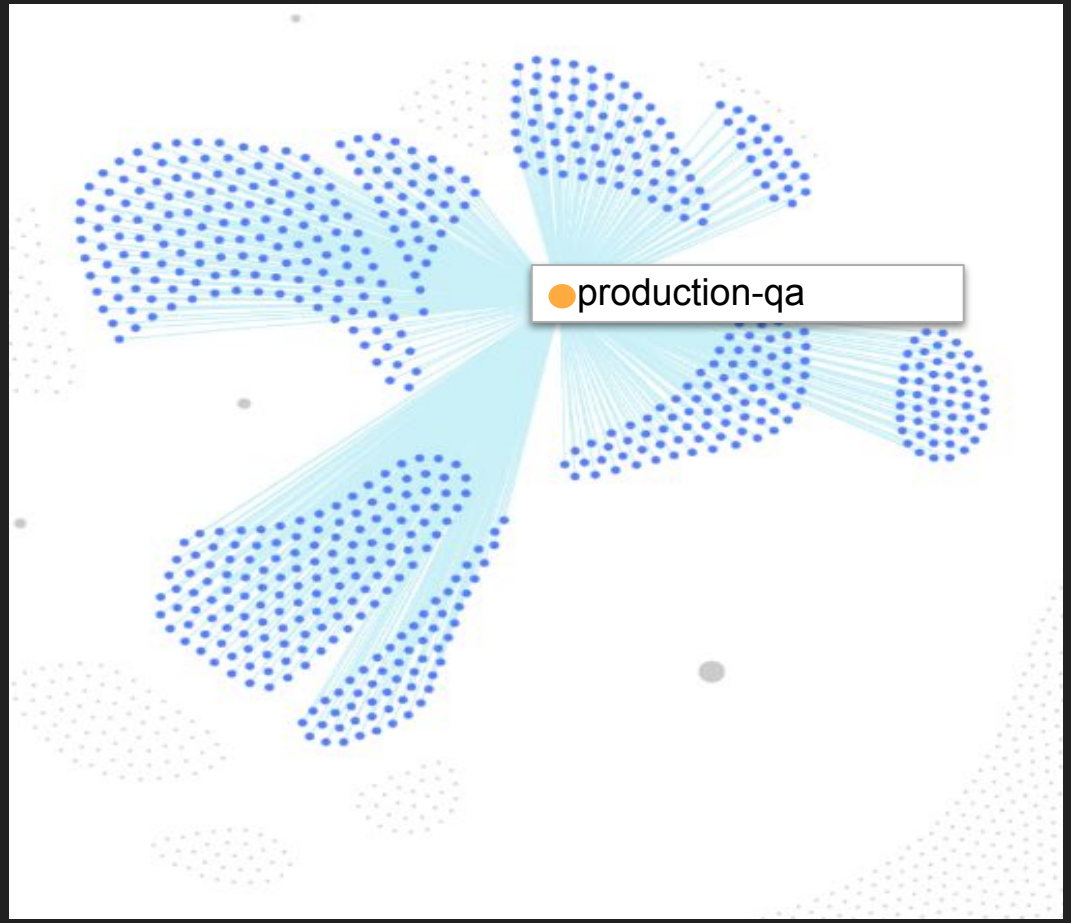
Overall UPR
0.93



Permissions Per User

QA Role

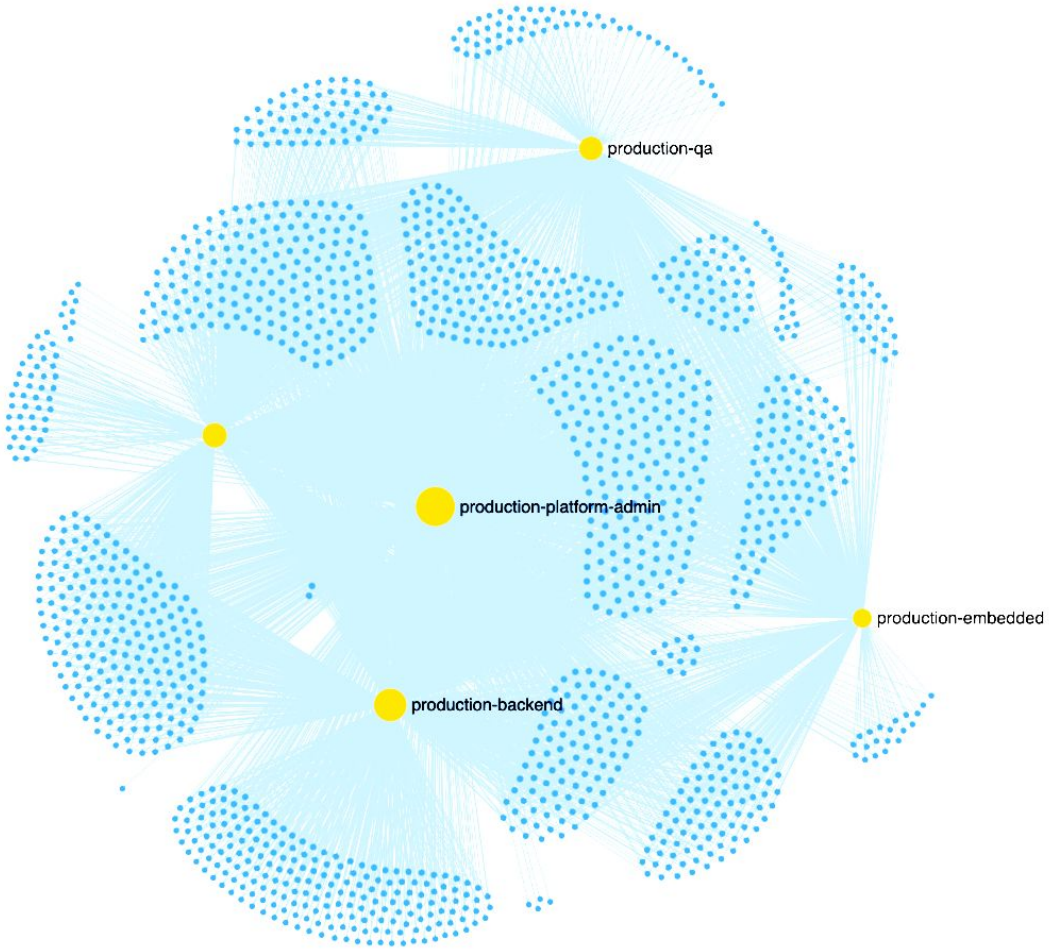
786



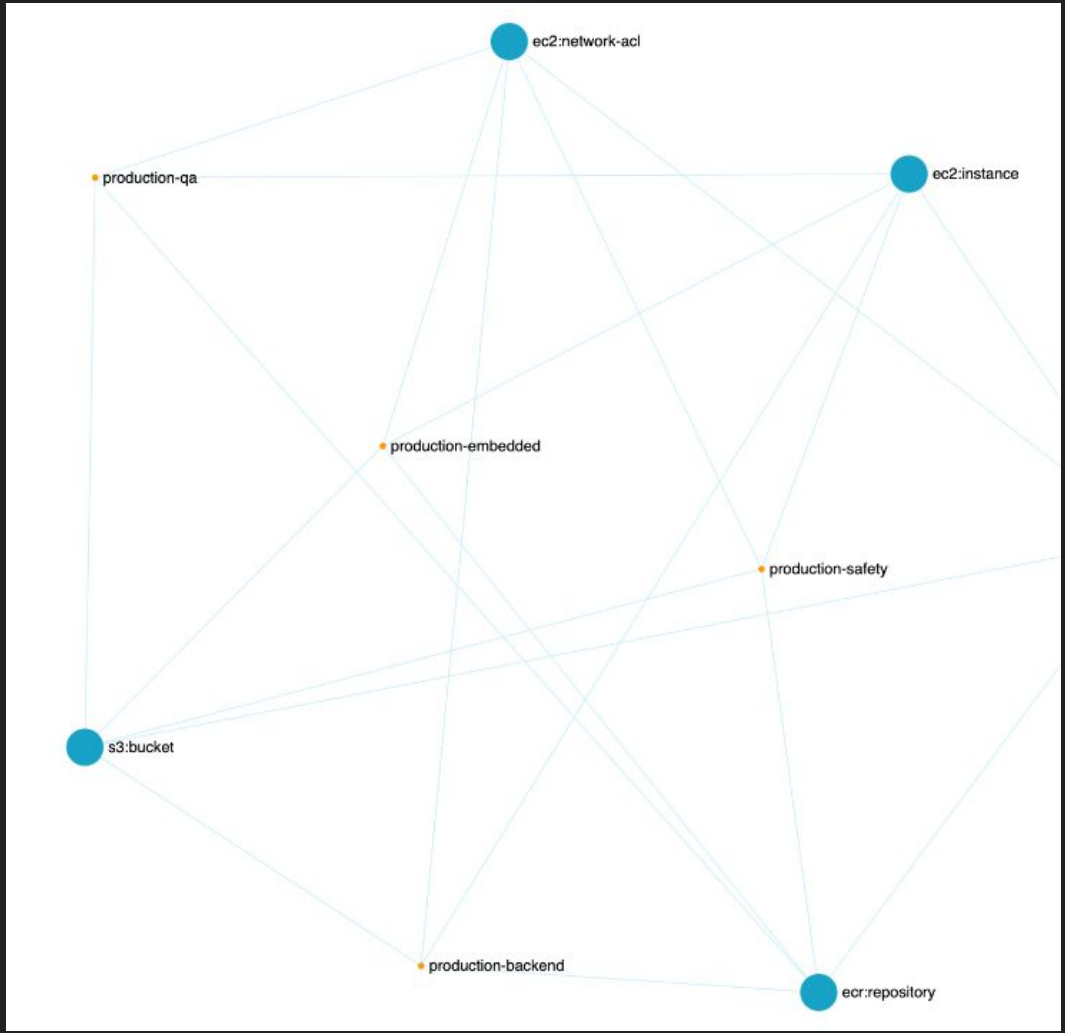
How to FIX?

FIX Strategy 1

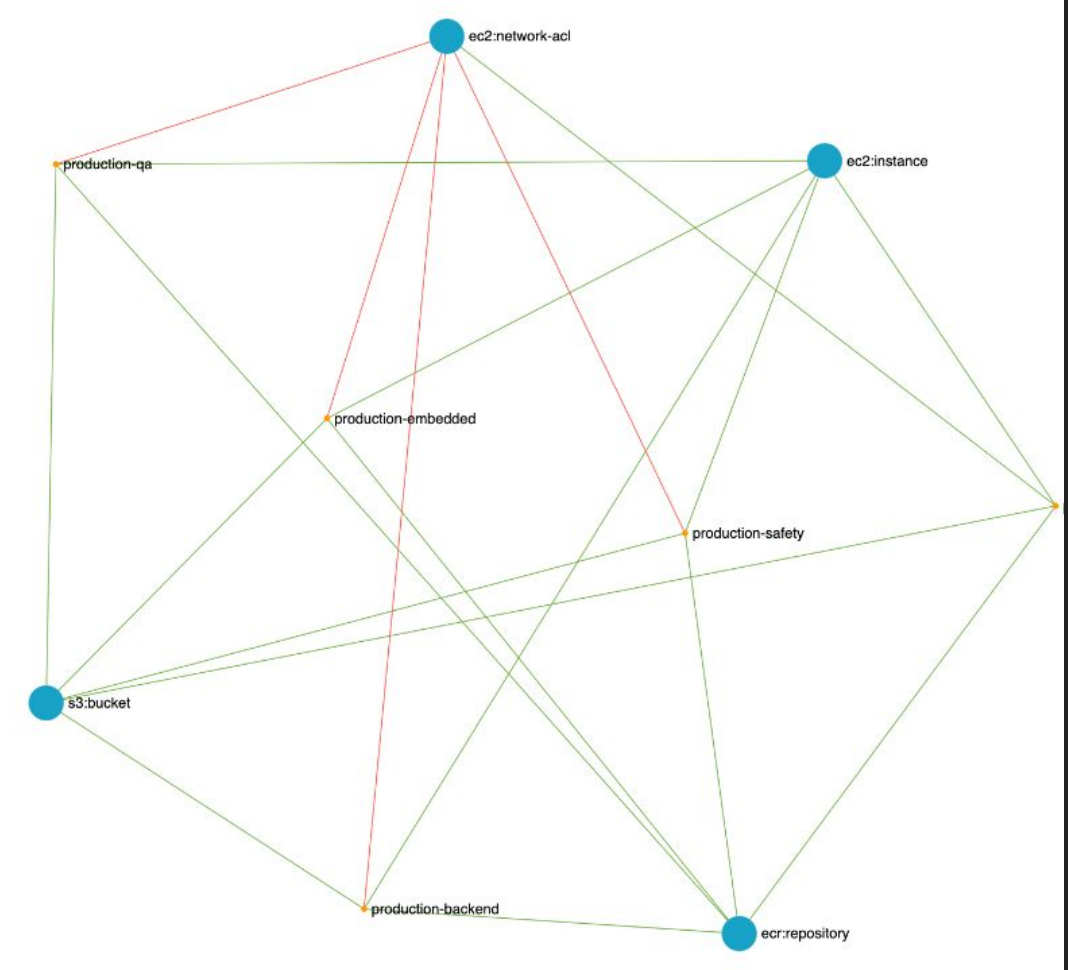
Let’s Simplify



Roles & Resource Types

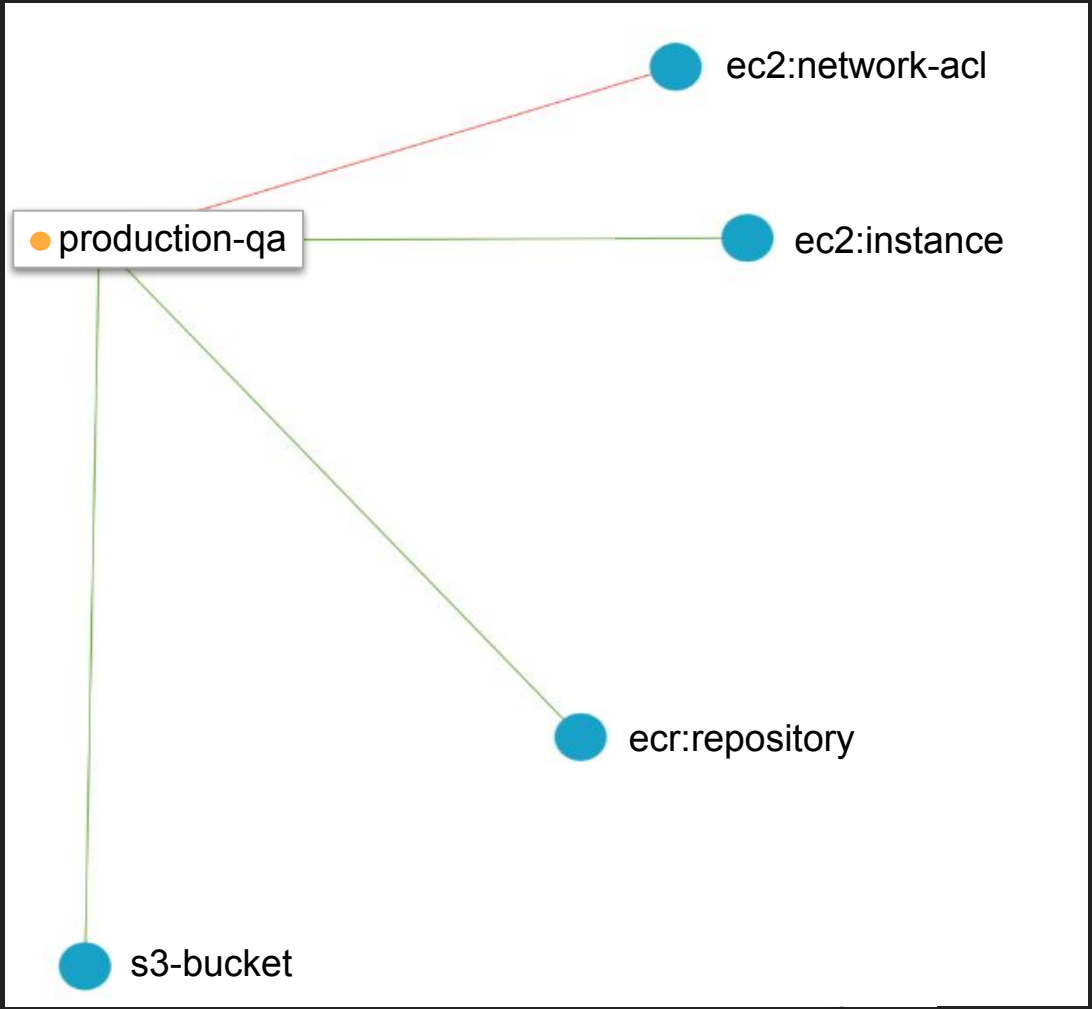


Unused Resource Types



Data Insight

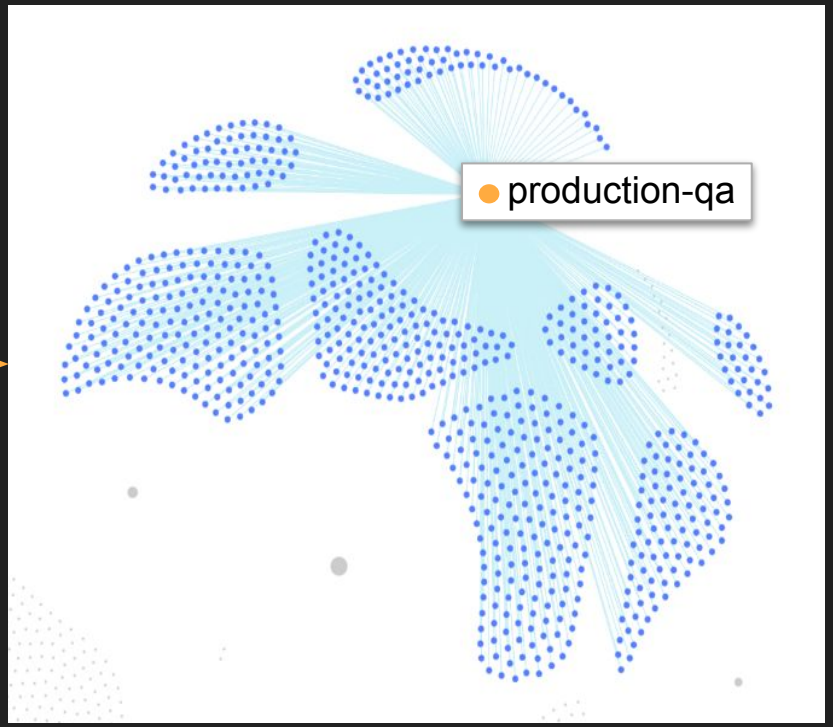
Unused Resource Types = 1



Strategy 1: Remove Unused Resource Types

Permissions: 763

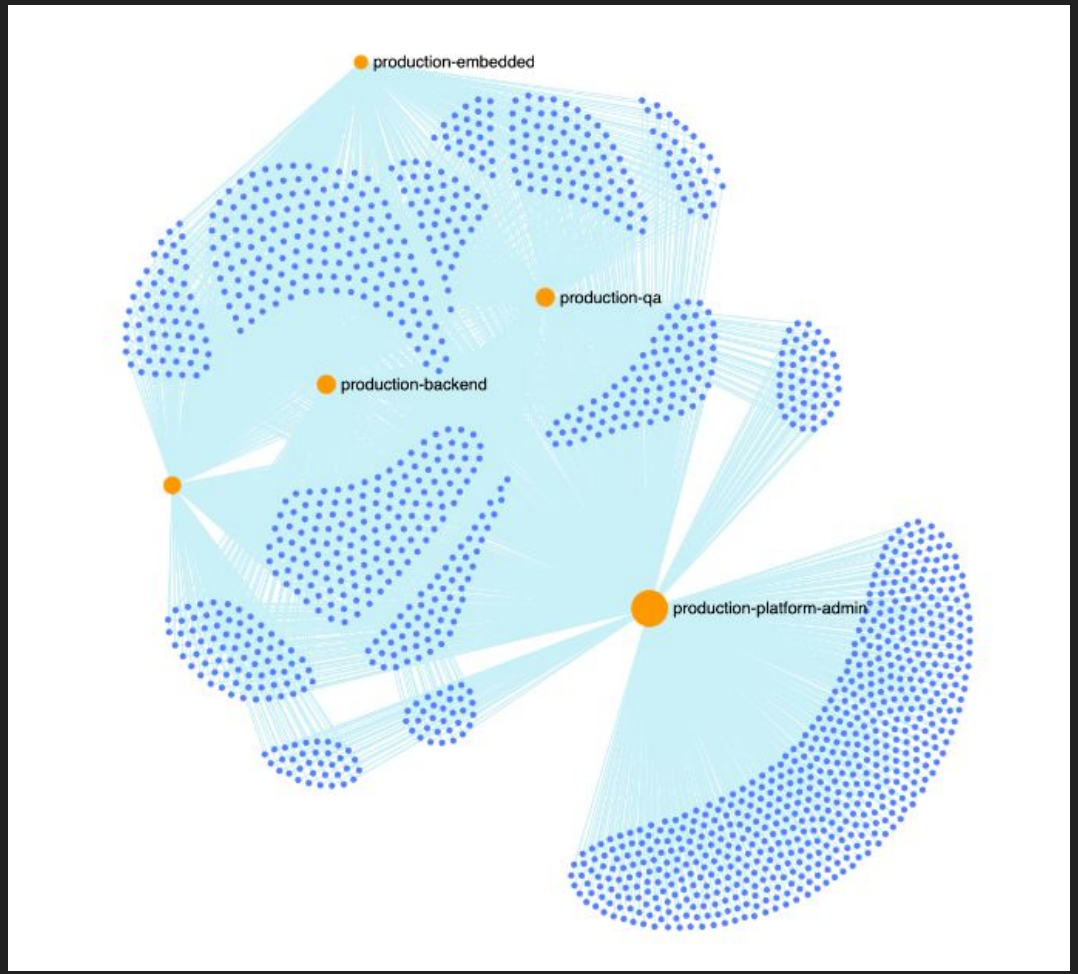
-20%



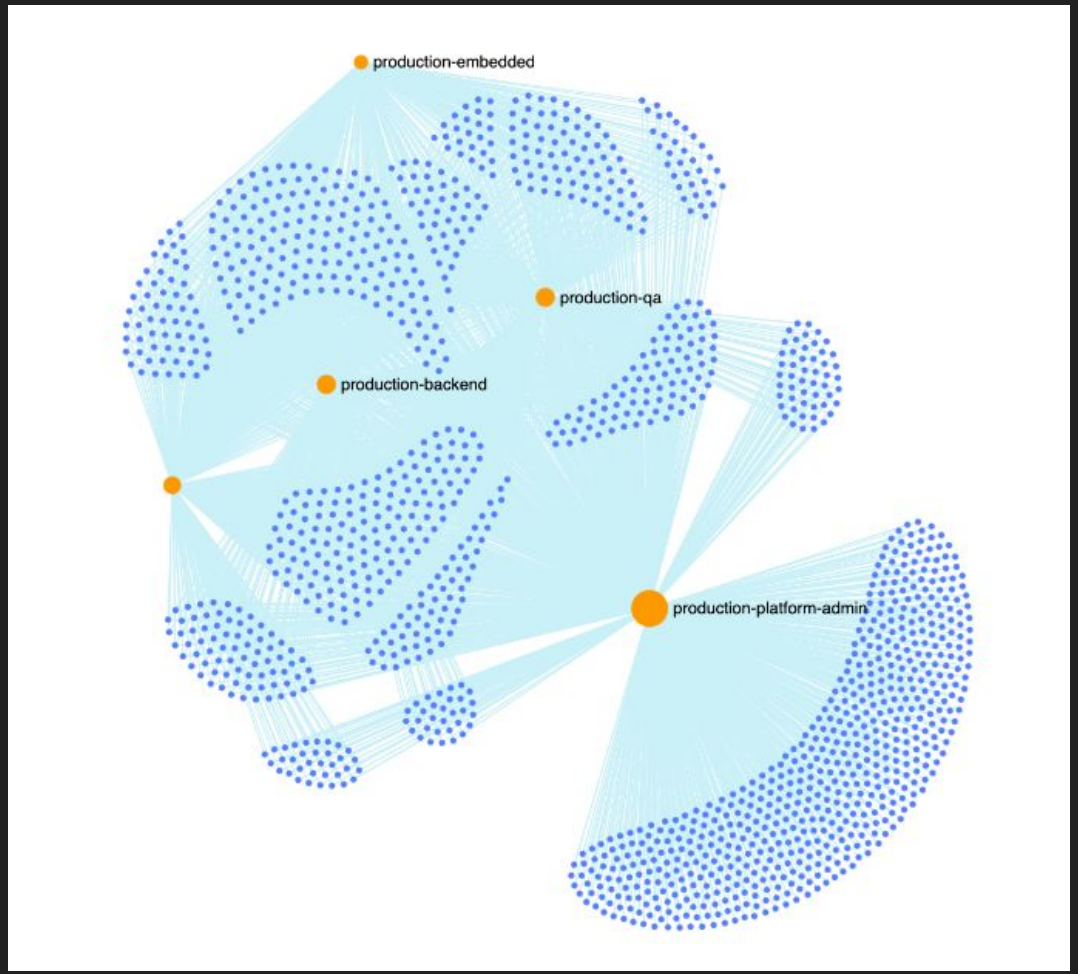
Permissions: 618

FIX Strategy 2

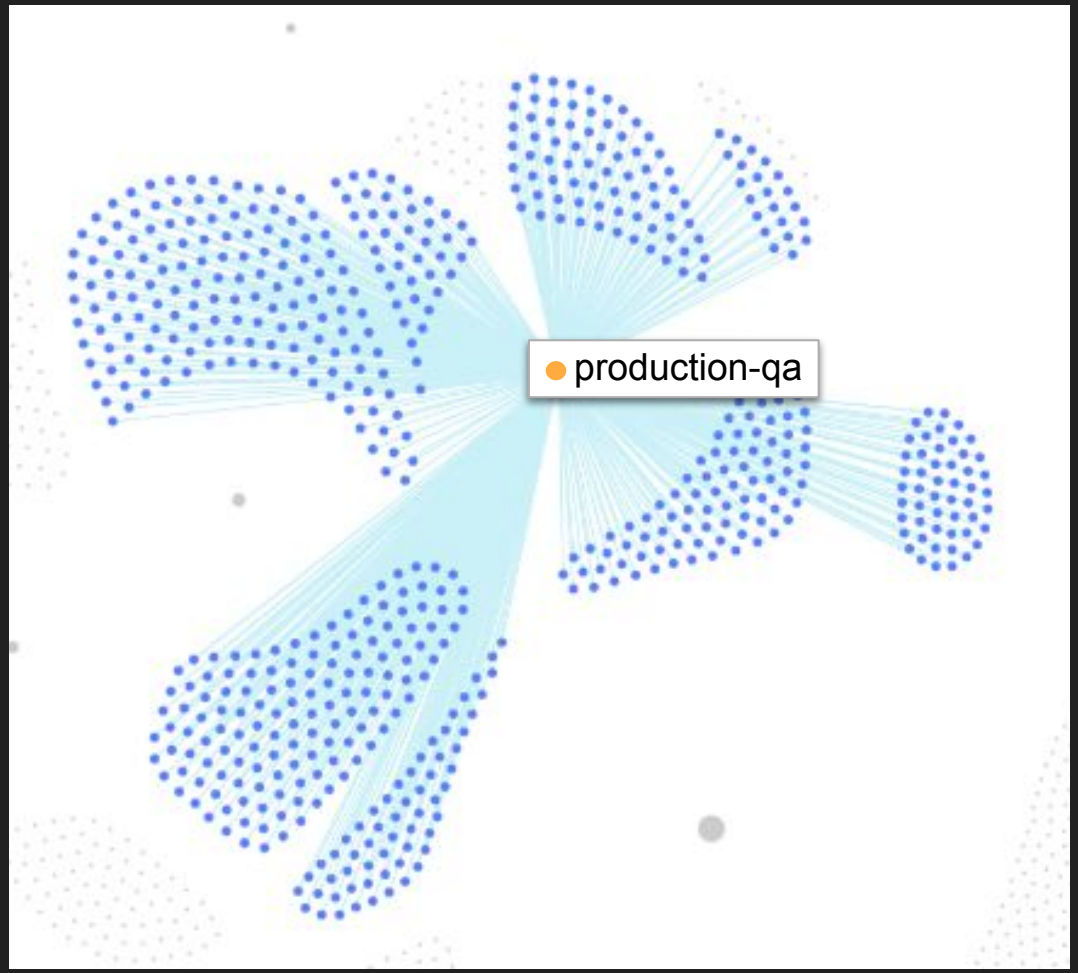
Permissions for Resource Types in Use



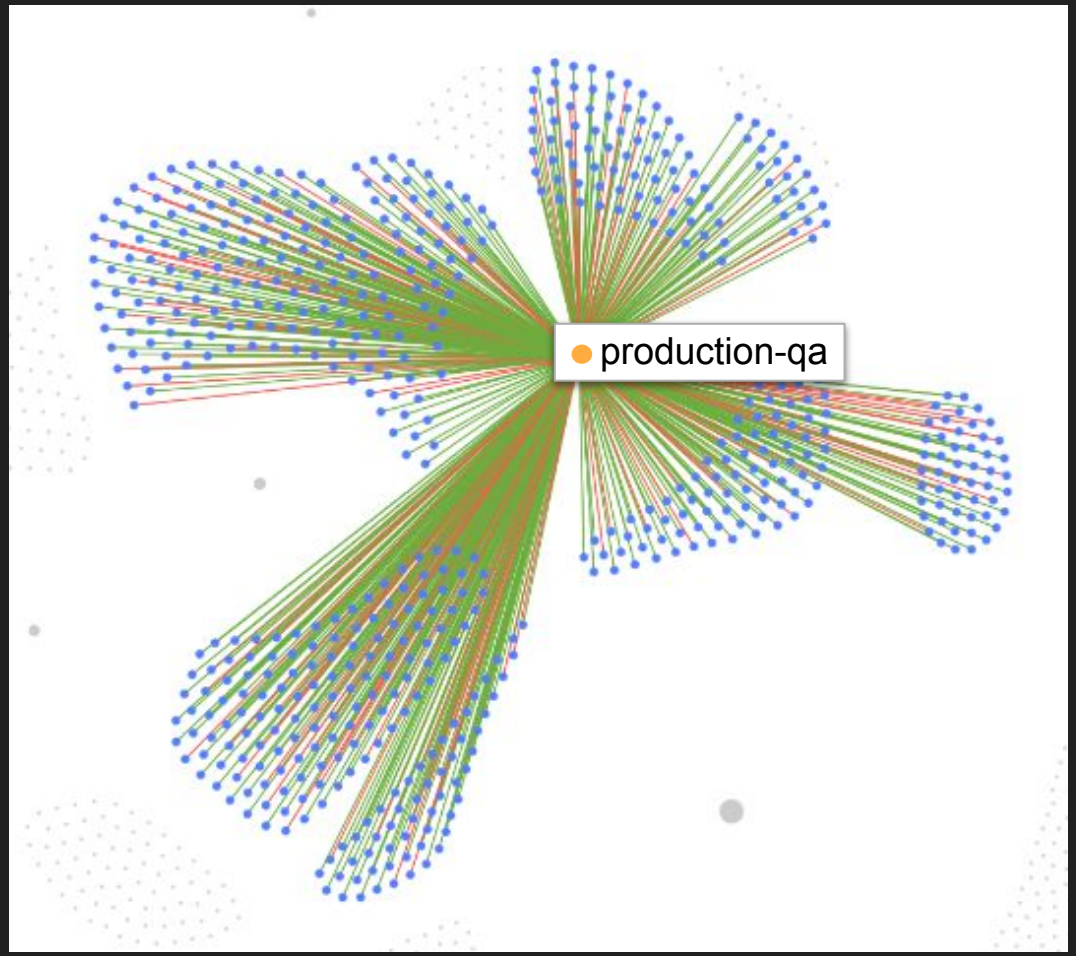
Let’s Simplify



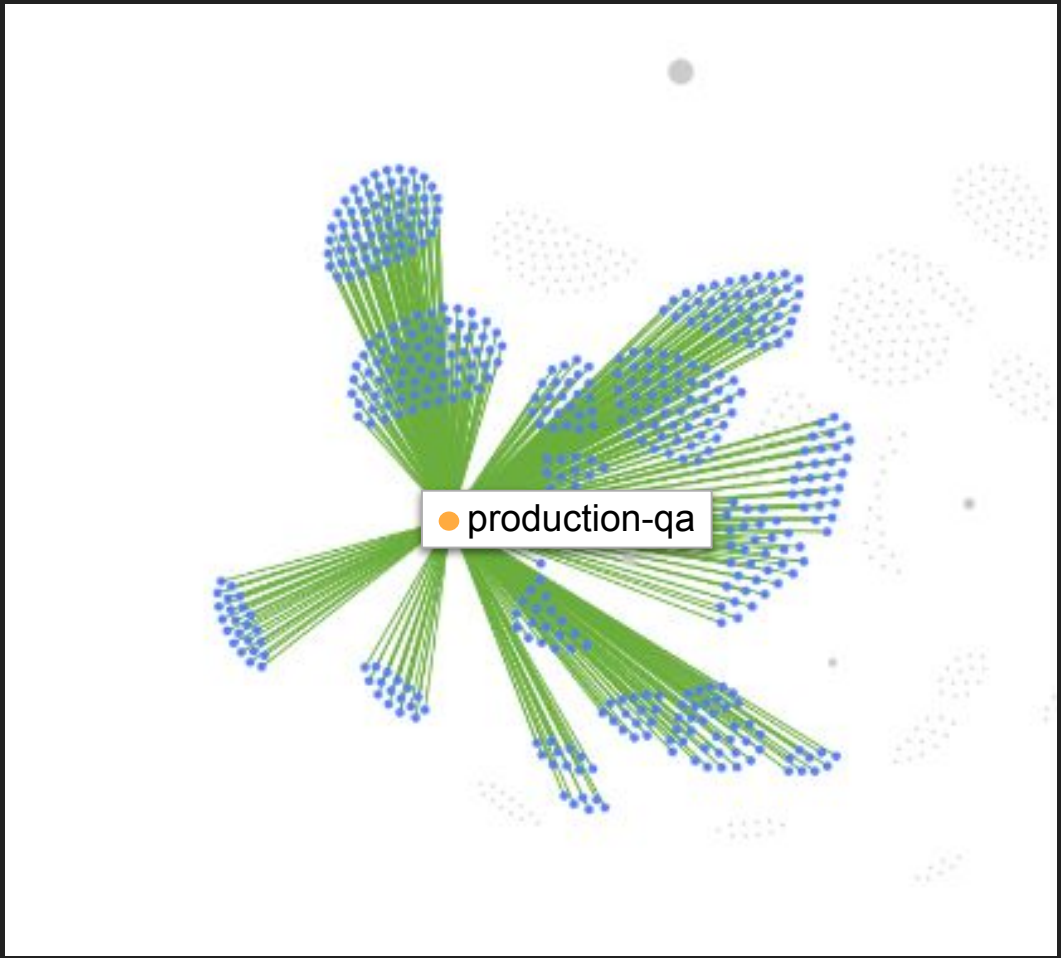
Permission Reduction



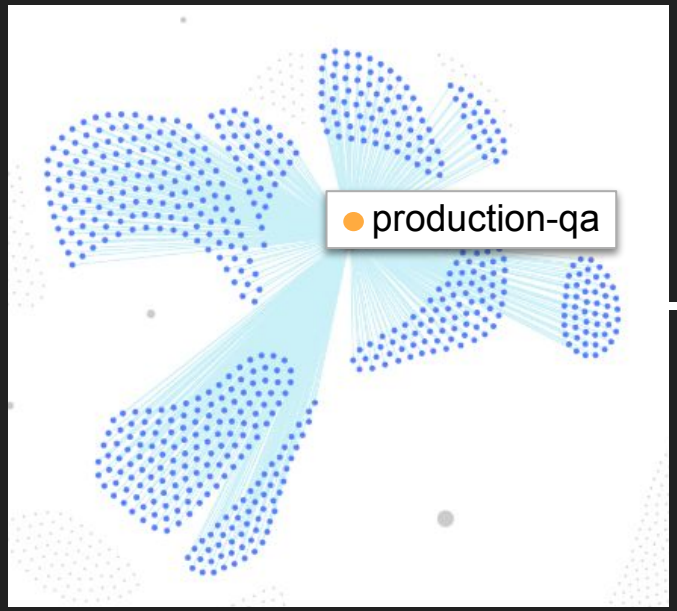
Used vs Unused Permissions



Unused Permissions Removed

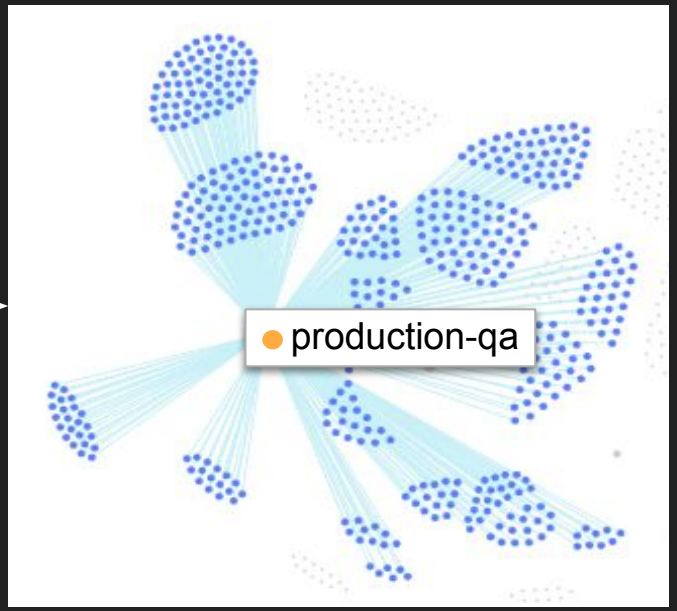


Strategy: Remove Unused Permissions



Permissions: 618

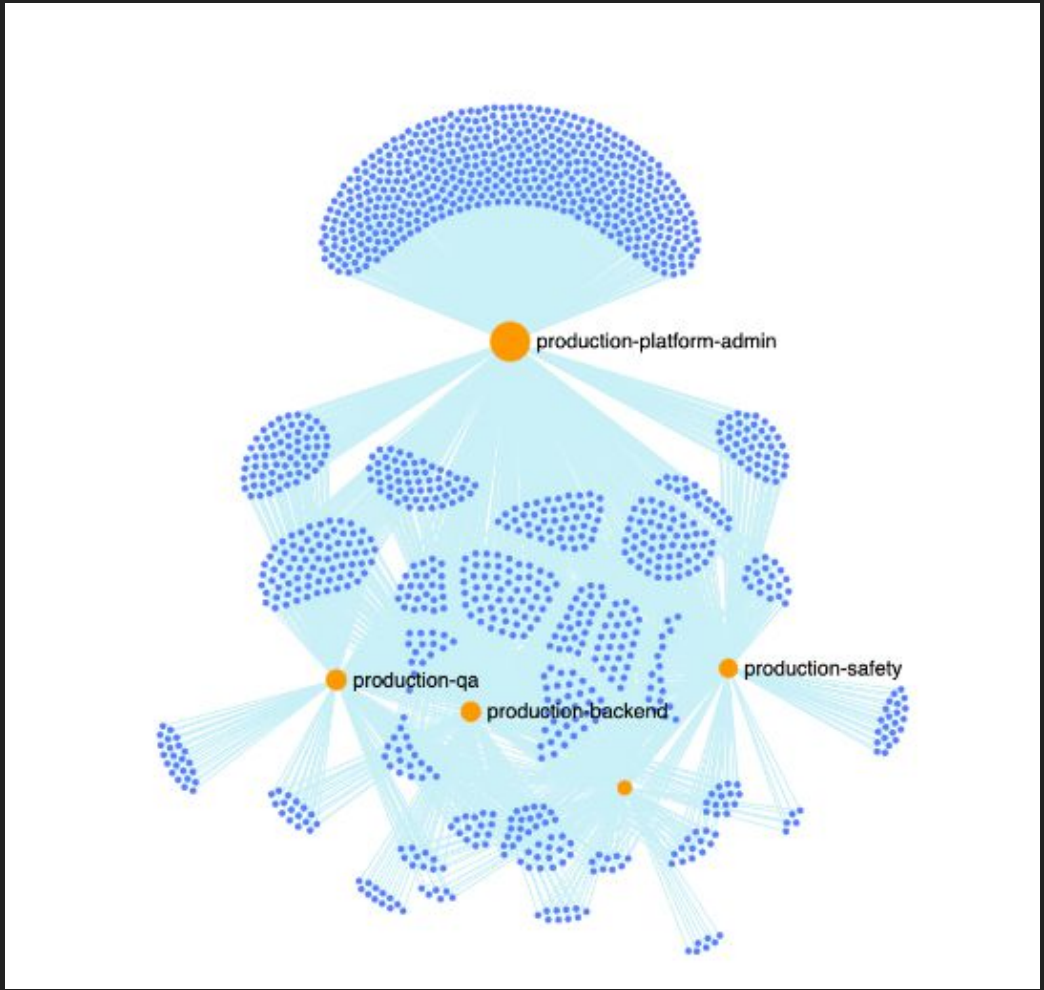
-23%



Permissions: 476

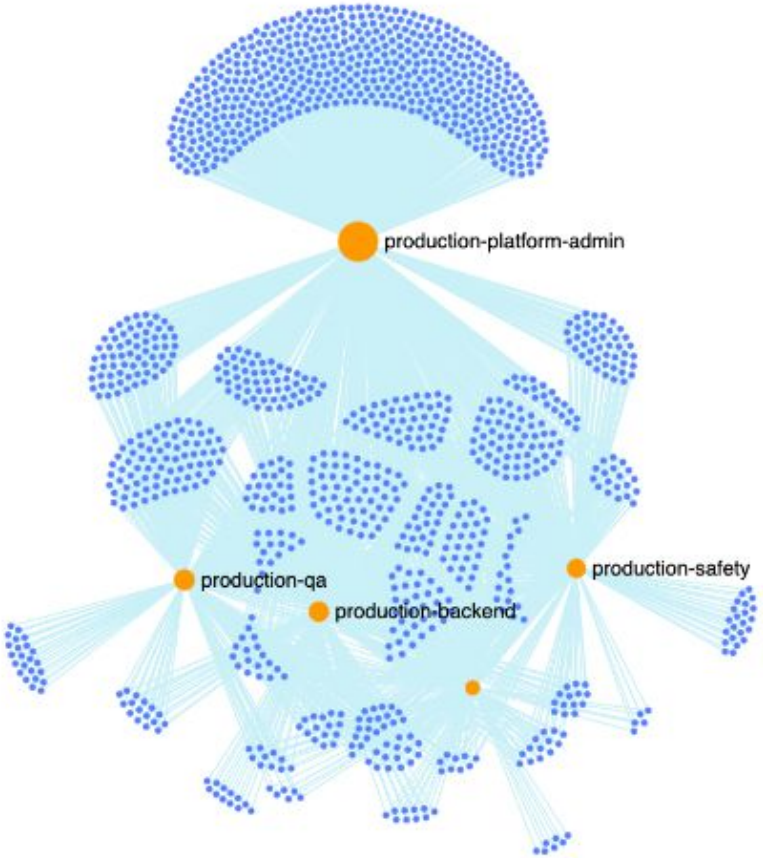
FIX Strategy 3

Only Used Permissions



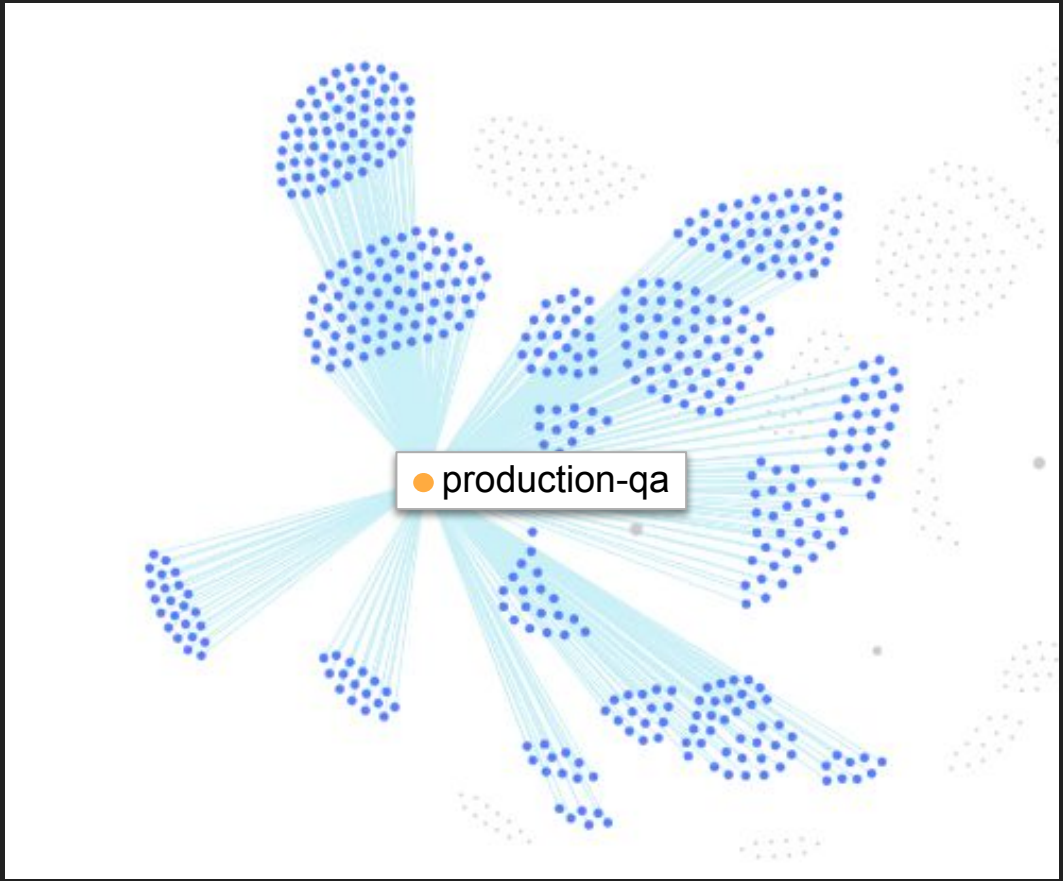
Role & Permissions - Used At Least Once

Let’s Simplify



Permissions Used At Least Once

Permissions - 476

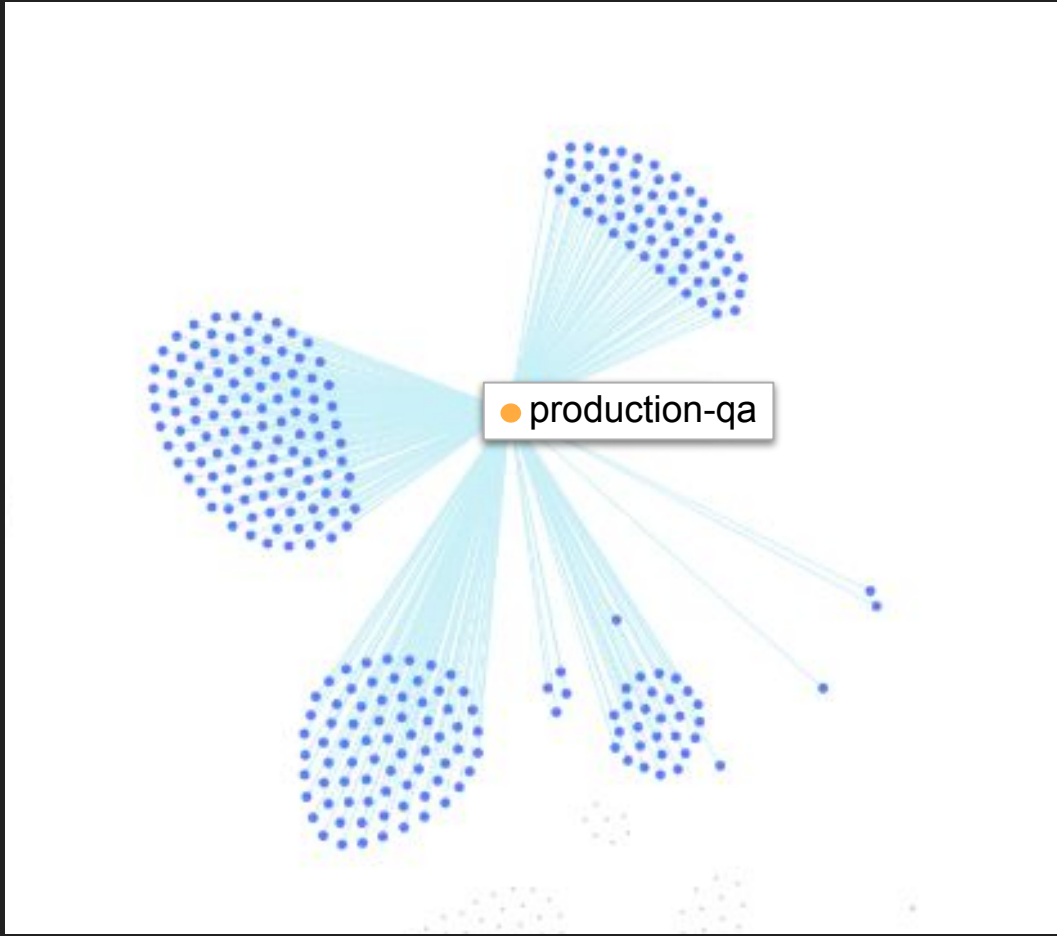


DB Table - Role Permissions

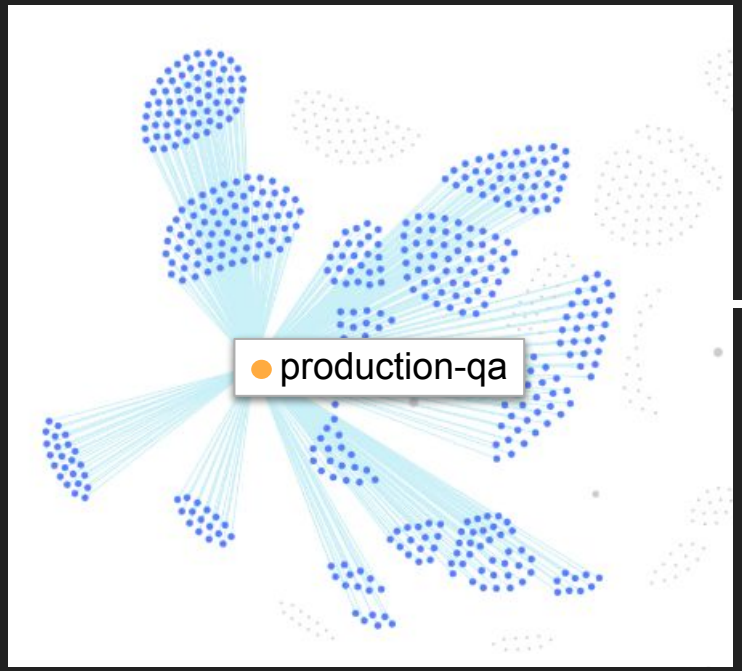
	role	resource	action	upr	pur
1	production-qa	arn:aws:ecr:us-east-1:933794580186:repository/fleet/k2web	BatchDeleteImage	0.89	0.11
2	production-qa	arn:aws:ecr:us-east-1:933794580186:repository/fleet/k2web	BatchGetImage	0.91	0.09
3	production-qa	arn:aws:ecr:us-east-1:933794580186:repository/fleet/k2web	CompleteLayerUpload	0.88	0.12
4	production-qa	arn:aws:ecr:us-east-1:933794580186:repository/fleet/k2web	CreateRepository	0.9	0.1
5	production-qa	arn:aws:ecr:us-east-1:933794580186:repository/fleet/k2web	DeleteLifecyclePolicy	0.93	0.07
6	production-qa	arn:aws:ecr:us-east-1:933794580186:repository/fleet/k2web	DeletePullThroughCacheRule	0.88	0.12

Rarely Used Permissions Removed

UPR > 0.95
Permissions - 286

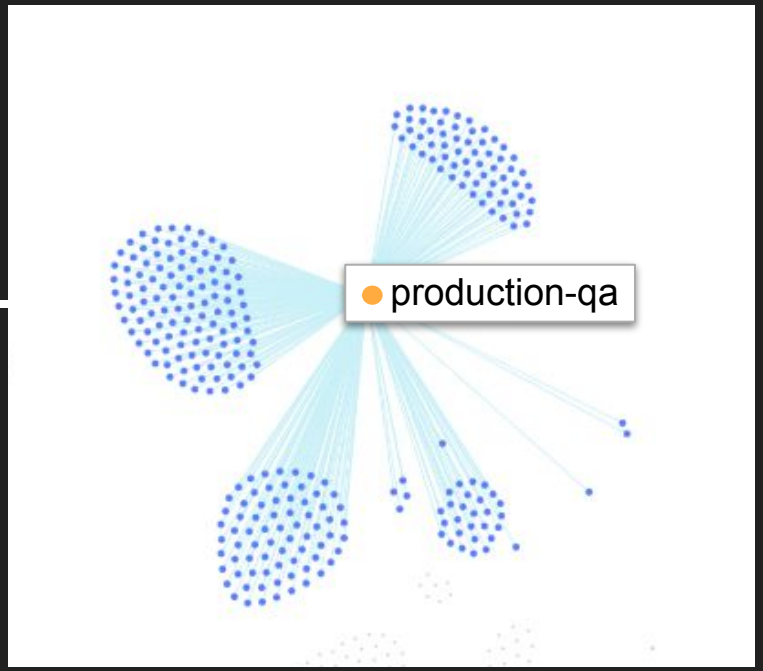


Strategy: Remove Rarely Used Permissions (UPR >0.90)



Permissions: 476

-40%



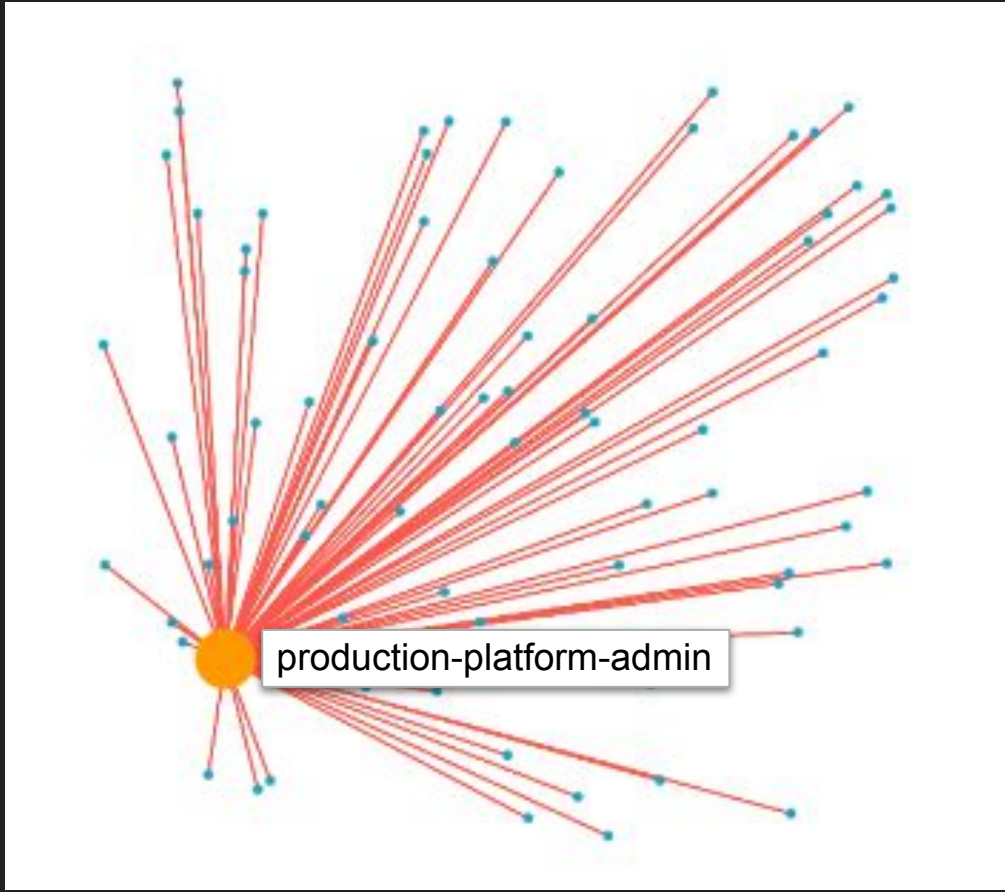
Permissions: 286

Pop Quiz

When a Company Grows,
Permissions per User Increases or Decreases?

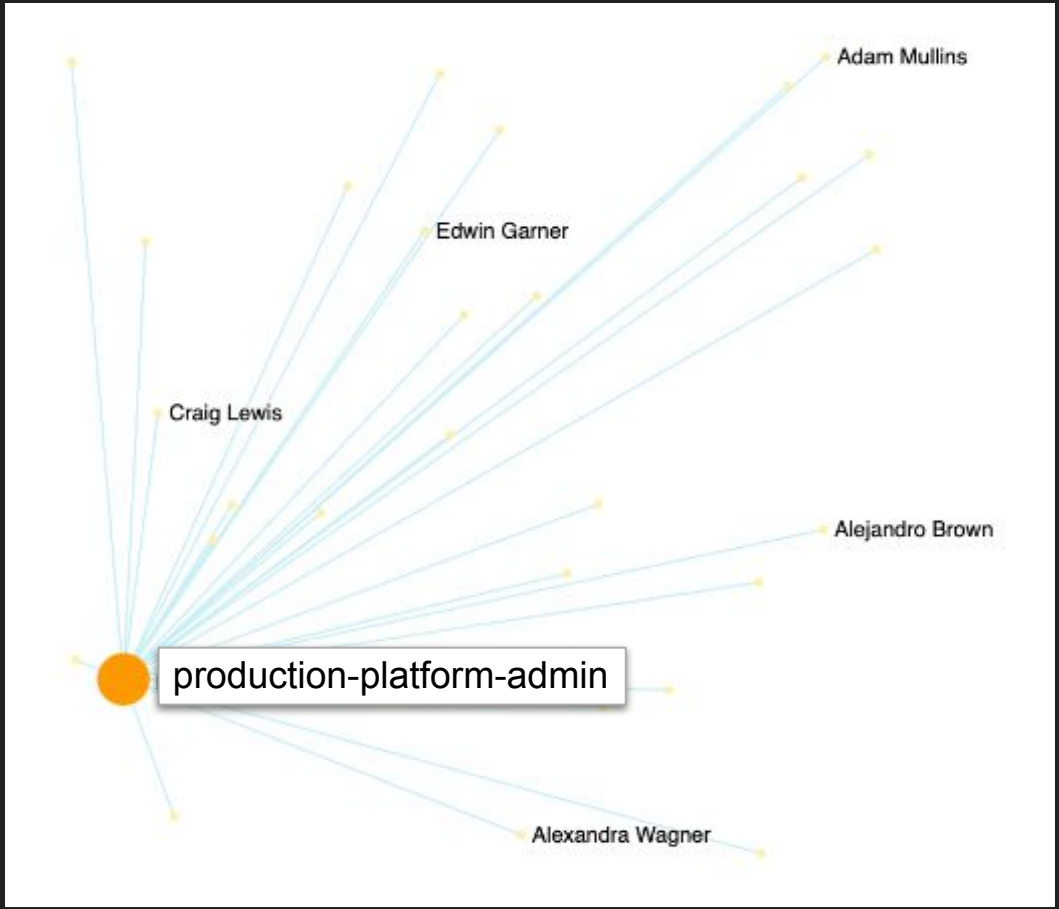
FIX Strategy 4

Production Platform Admin Role



Production Platform Admin Role

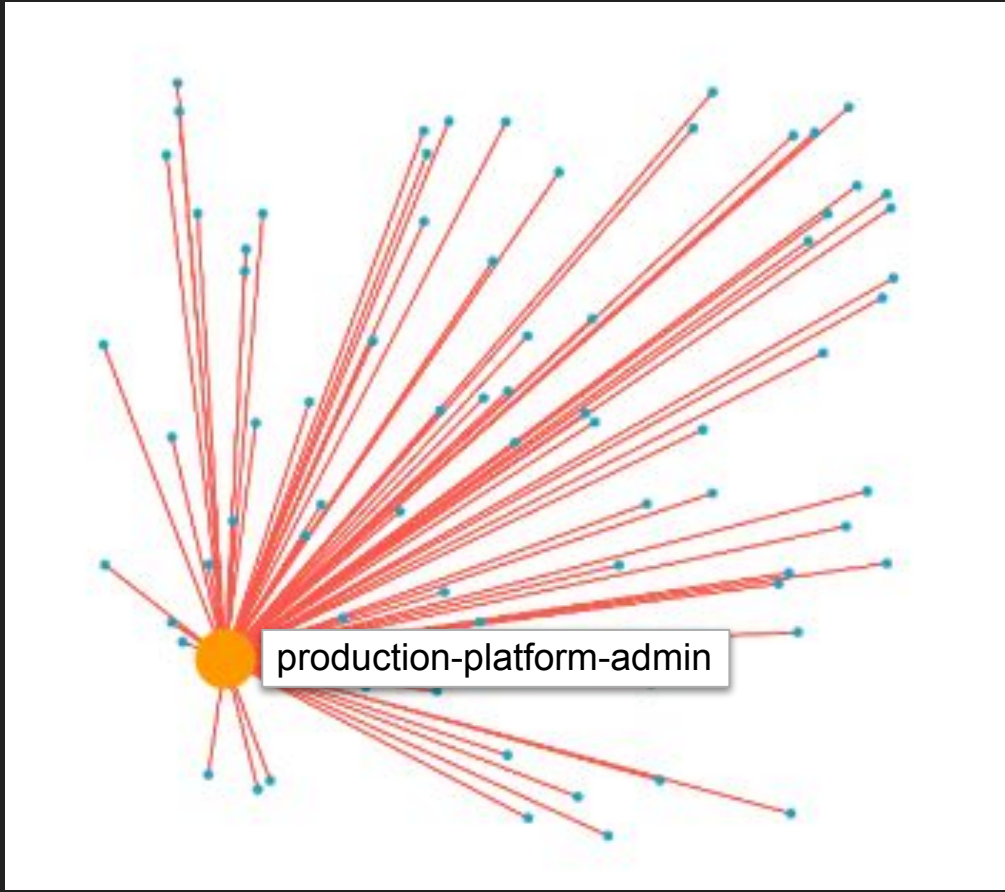
Users
29



Production Platform Admin Role

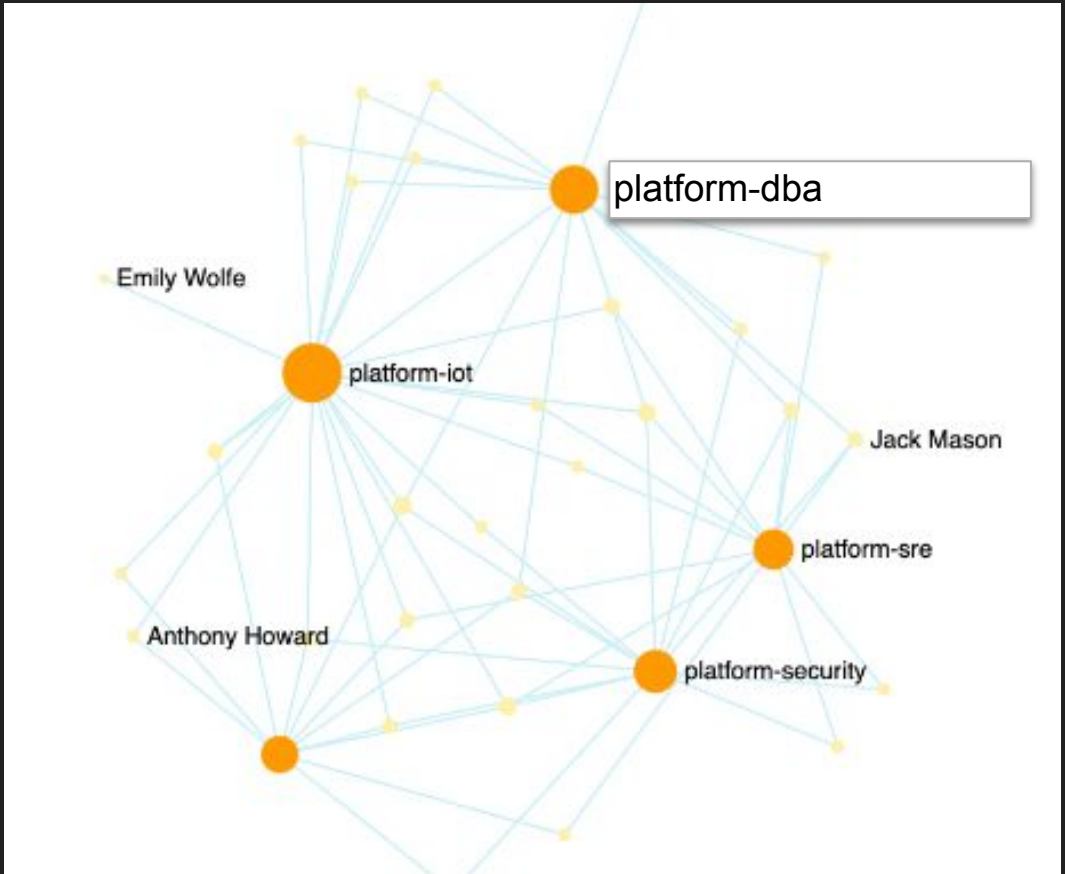
Permissions Per User

75



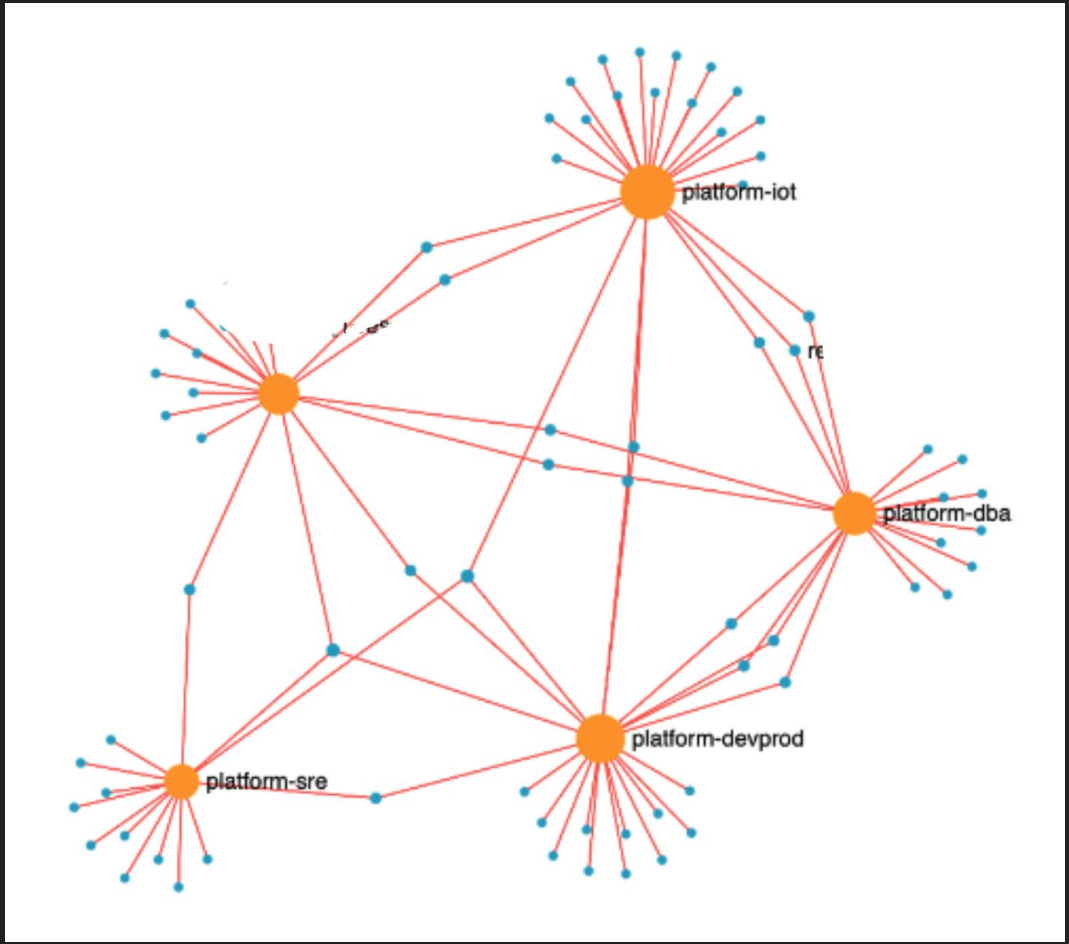
Production Platform Admin Role

Split the Role



Role - Users & Permissions

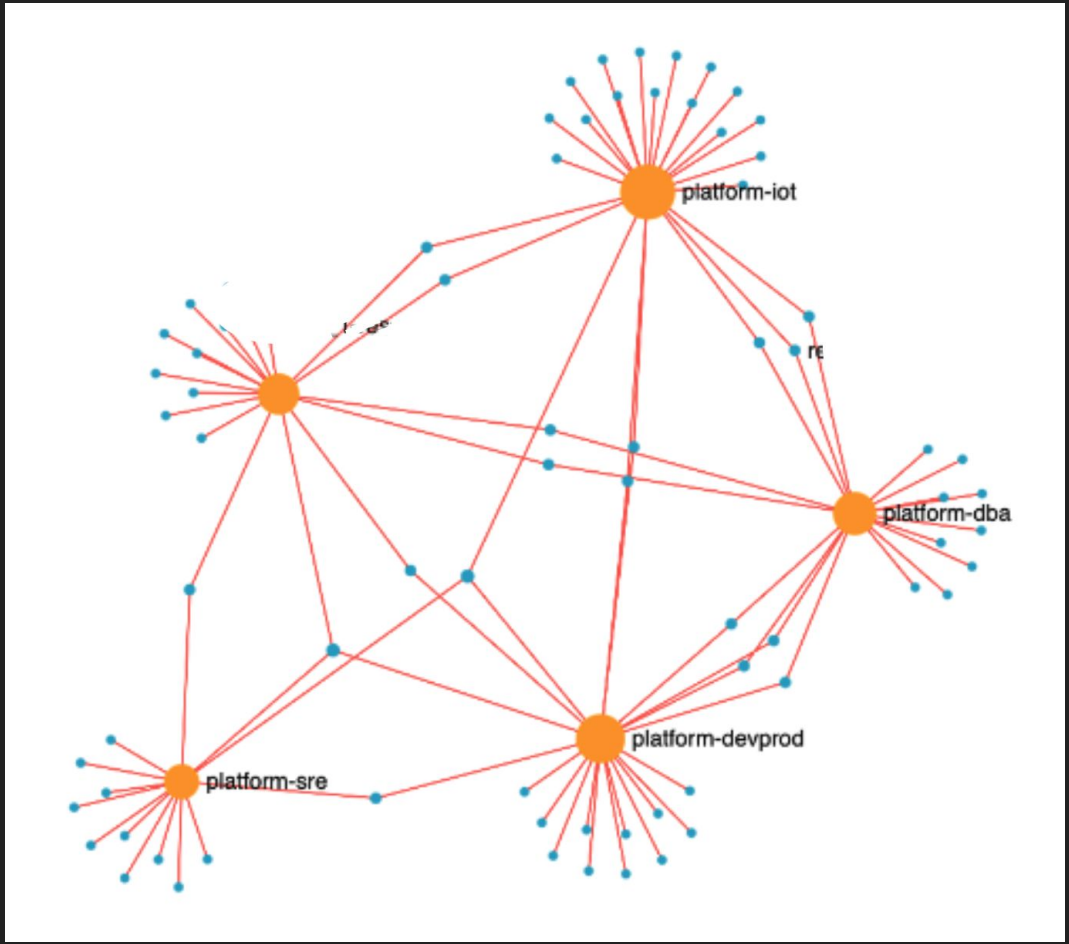
Permissions Splitted



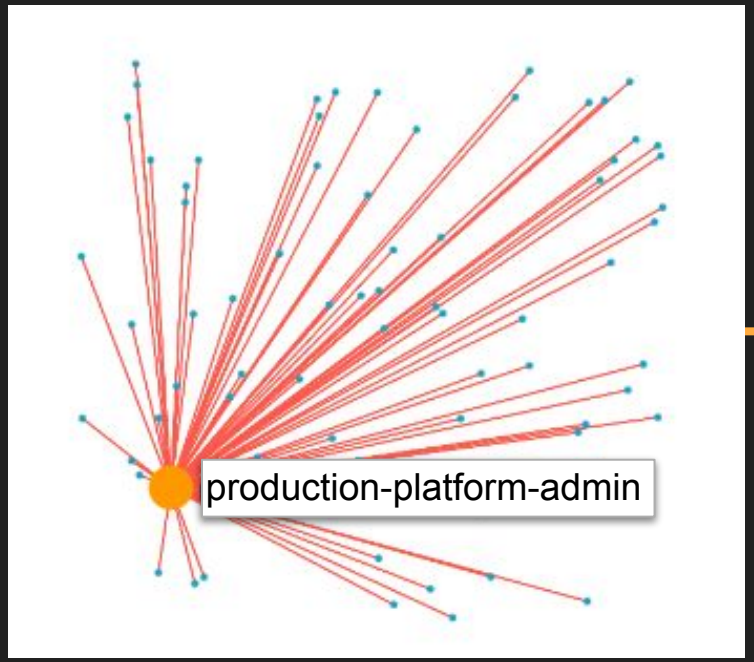
Role - Users & Permissions

Permissions per User

32

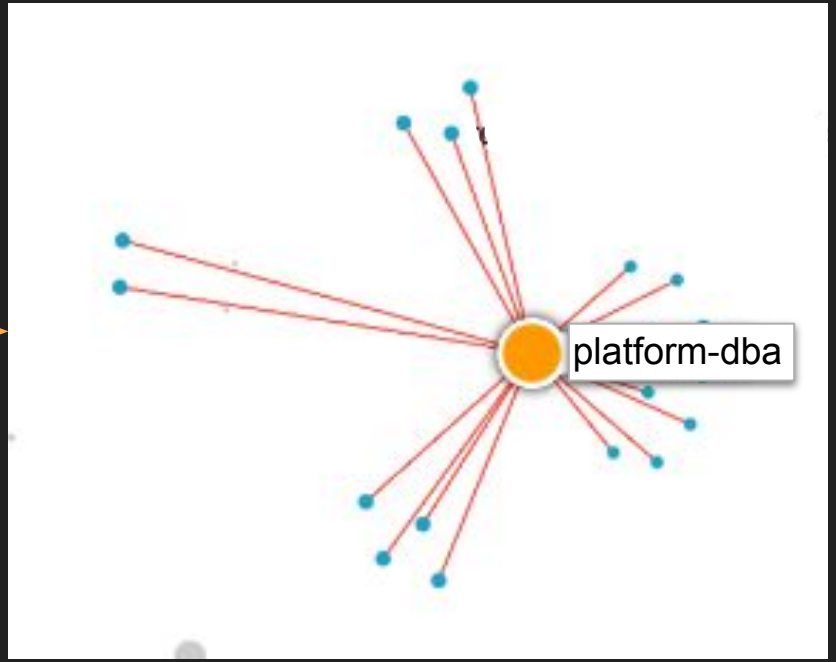


Strategy: Split the Permissions



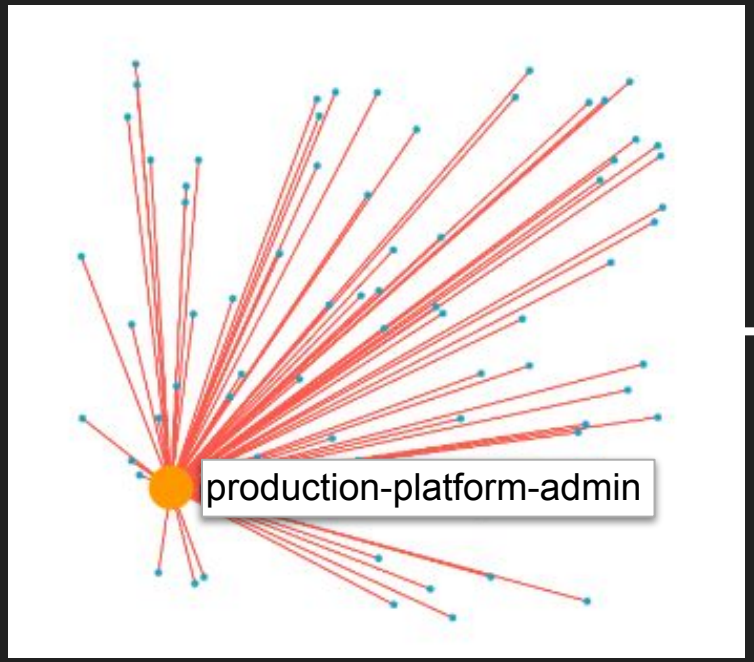
Permissions: 75

-76%



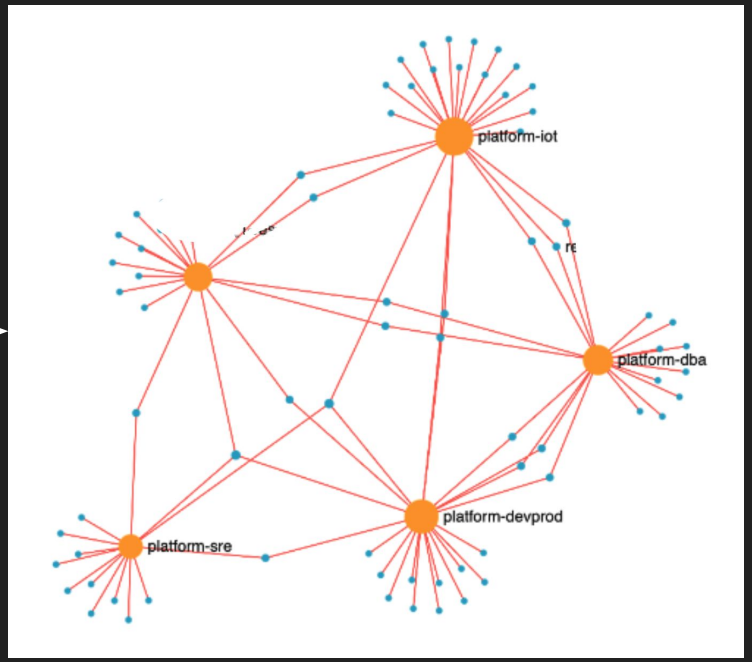
Permissions: 18

Strategy: Split the Permissions



Permissions Per User: 75

-57%

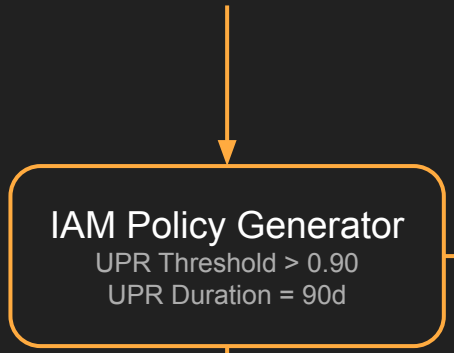


Permissions Per User: 32

How to KEEP IT AWAY?

Automated Policy Generation

Runs Every Week




```
data "aws_iam_policy_document" "piam_team_fleet_card_policy" {
  statement {
    actions = [
      "sqs:SendMessage",
      "sqs:SetQueueAttributes",
      "sqs:SendMessageBatch",
      "sqs:PurgeQueue",
      "sqs:ReceiveMessage",
      "sqs:ChangeMessageVisibility",
      "sqs:DeleteMessage"
    ]
    resources = [
      "arn:aws:sqs:us-east-1:933794580186:file-ingestion-*",
      "arn:aws:sqs:us-east-1:933794580186:v2c-candidates-*"
    ]
  }
}
```

AWS IAM Role

Role Permissions

RECAP

Identifying & Reducing Permission Explosion is a Data Problem

Strategies to Fix Permission Explosion:

Reason for Permission Explosion	Solution
Permission Creep	Remove Unused Permissions
Temporary Access	Remove Rarely Used Permissions
Broad Access Roles	Create Smaller Team/Subteam Specific Roles

How to GET STARTED?

How to GET STARTED ?

Push IAM Data to Role Permissions Database

Use Workbench Notebook to Identify Permission Explosion

Generate New Policies Based on Findings

Automate Policy Generation & Enforcement of Policies

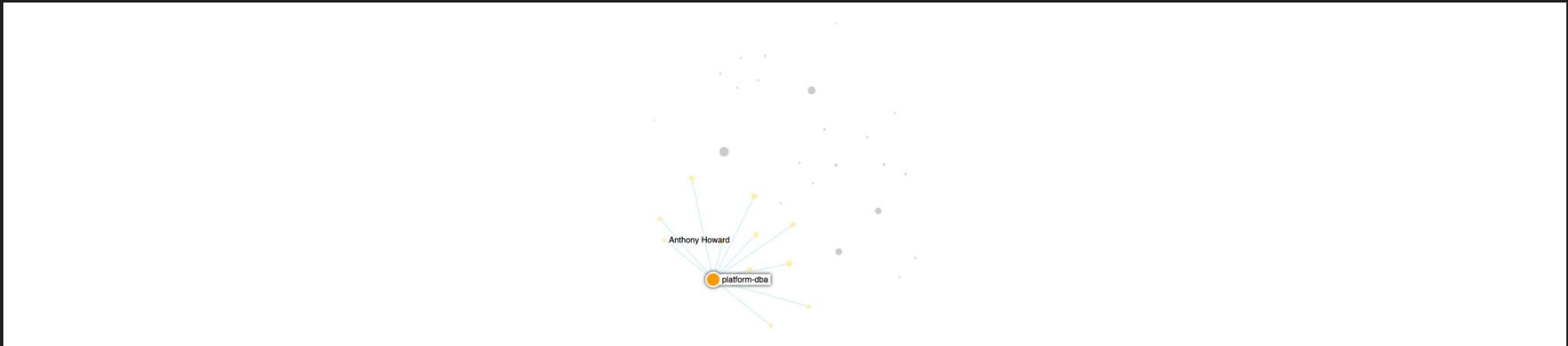
PermCutter

https://github.com/PankajMoolrajani/PermCutter

Q&A

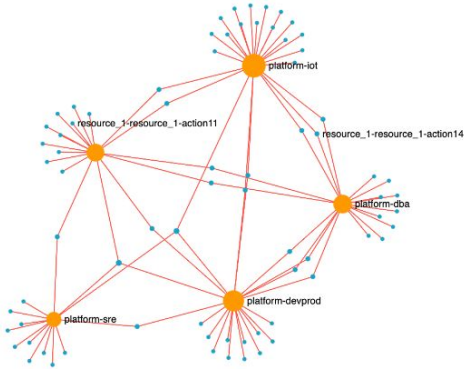
Thank You

Role - Users & Permissions



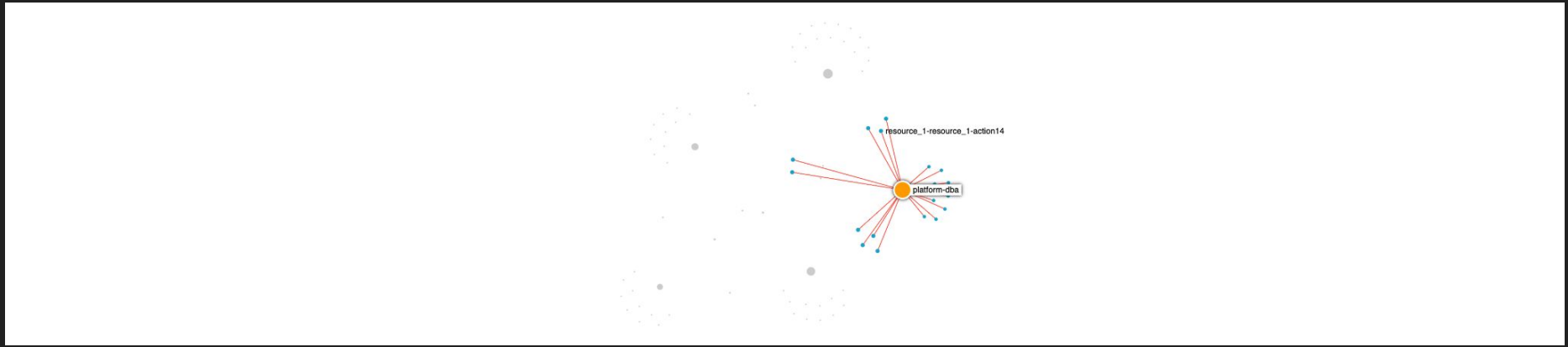
75 permissions 29 users Dba 18 lot 24

Role - Users & Permissions



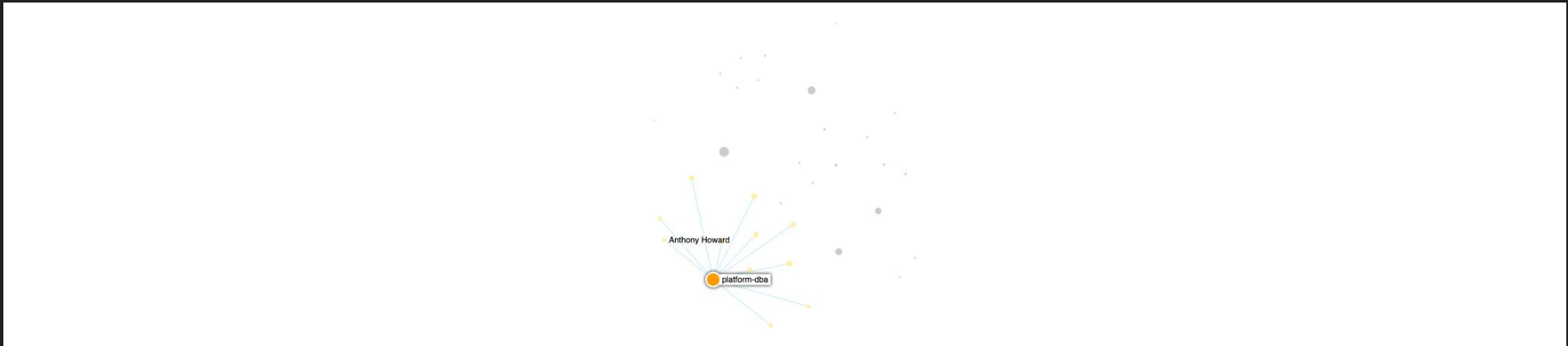
75 permissions 29 users Dba 18 Iot 24

Role - Users & Permissions



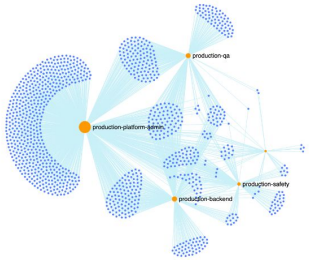
75 permissions 29 users Dba 18 lot 24

Role - Users & Permissions



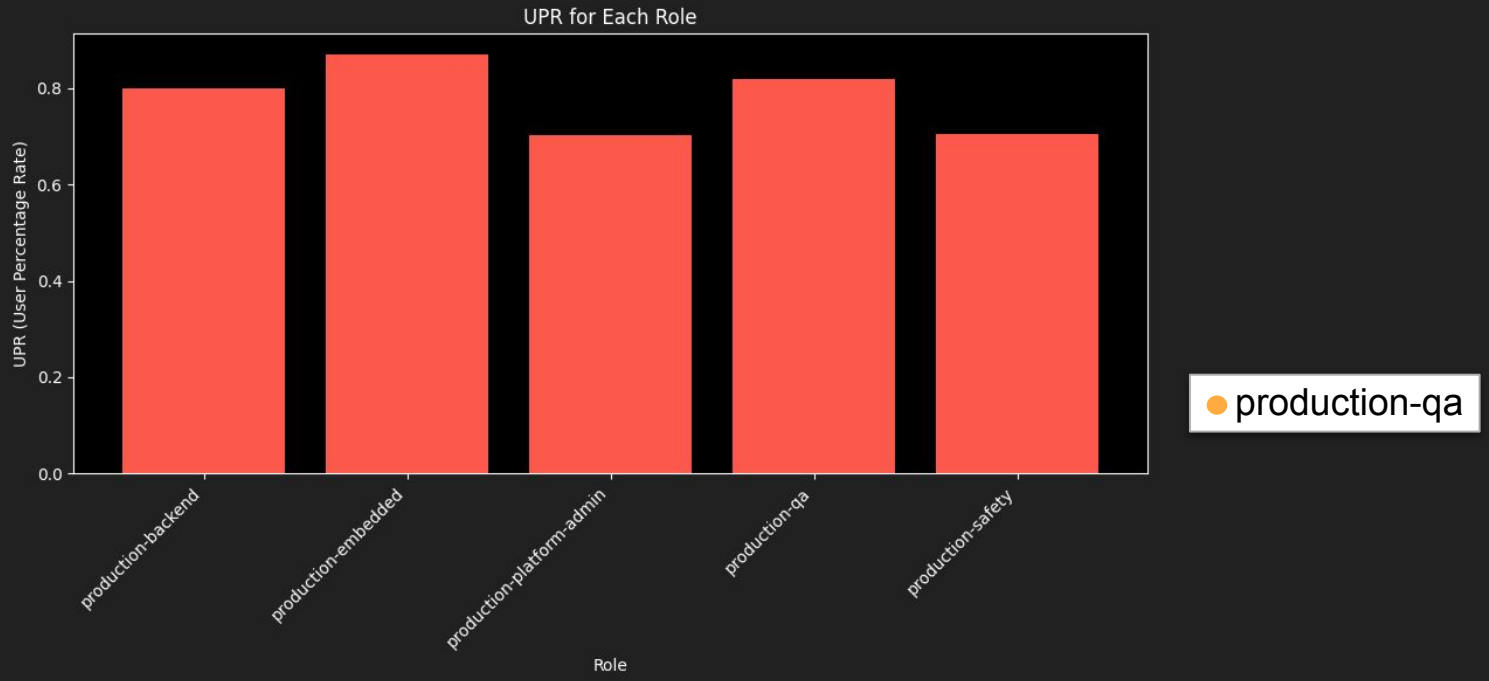
75 permissions 29 users Dba 18 lot 24

Strategy 3 - Remove Rarely Used Permissions
476 upr dropped to 286

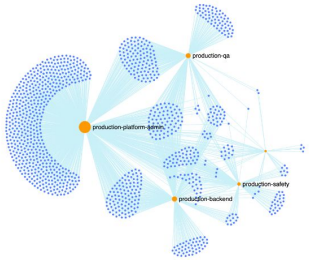


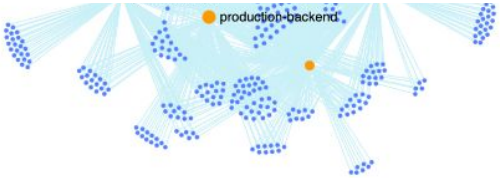
production-qa

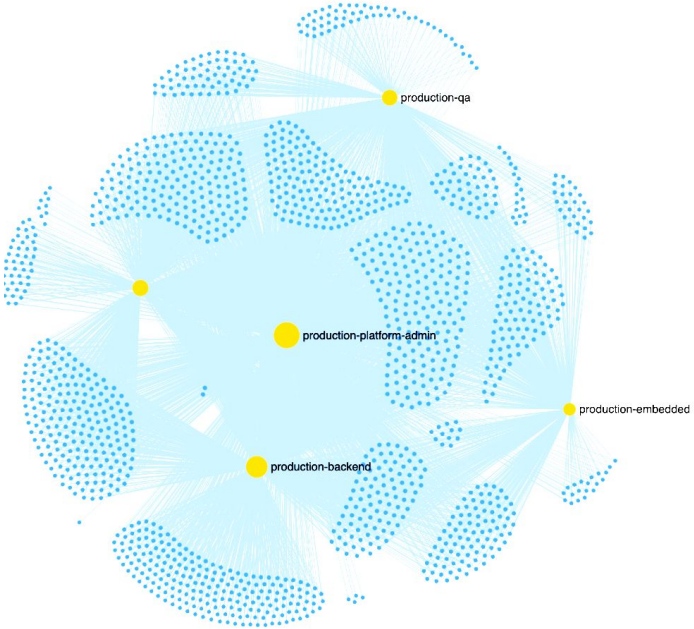
Strategy 3 - Remove Rarely Used Permissions
0.78



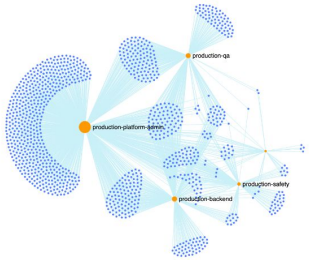
Pe

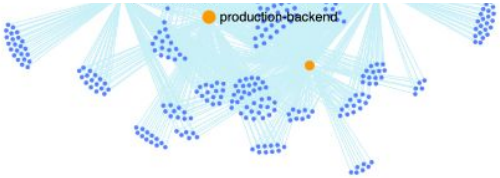

production-qa
production-platform-admin
production-safety
production-backend

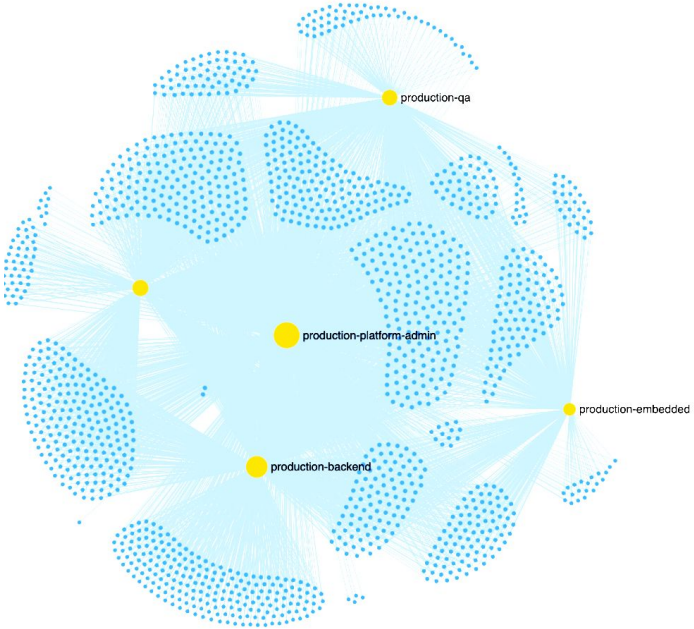

production-backend


production-qa
production-platform-admin
production-embedded
production-backend

Pe

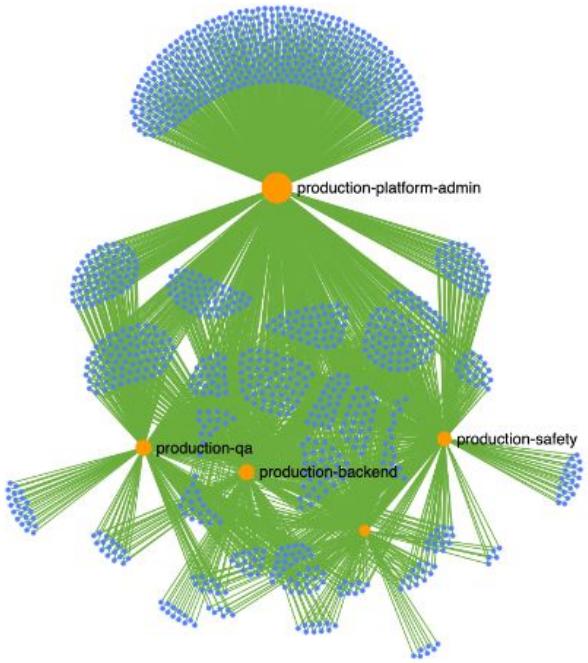

production-qa
production-platform-admin
production-safety
production-backend

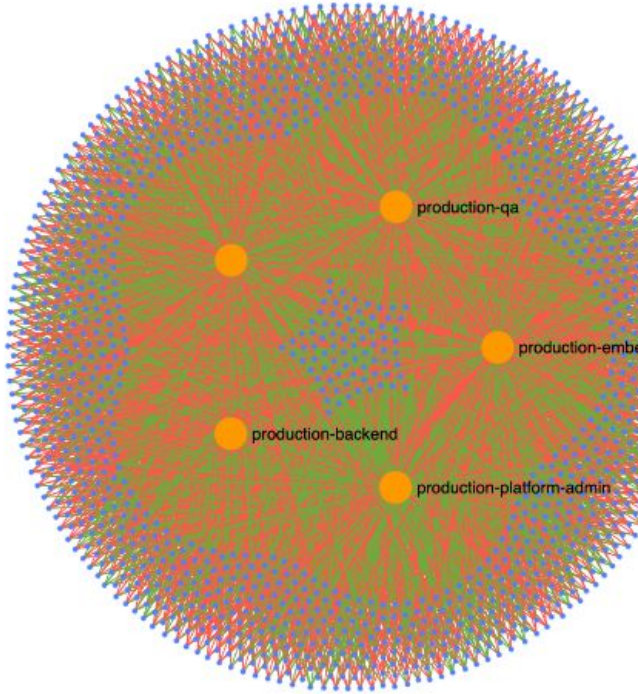

production-backend


production-qa
production-platform-admin
production-embedded
production-backend

Permission Reduction

Remove Permissions f

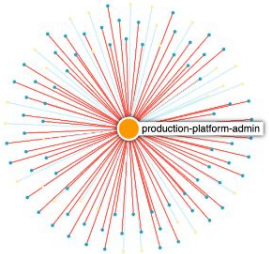




Split Roles

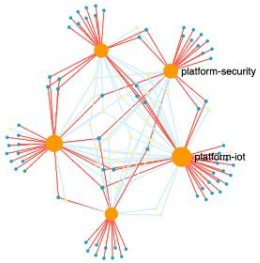
UPR: 0.46

Permission Reduction



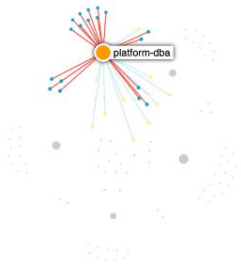
Avg per user 102

Permission Reduction

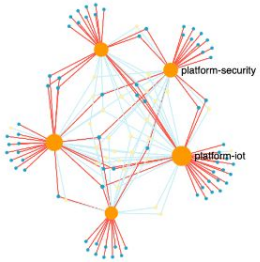


Average per user
32

Permission Reduction



Security - 30
Dba 30
Devprod - 36



Split Roles

UPR: 0.46

Split Roles

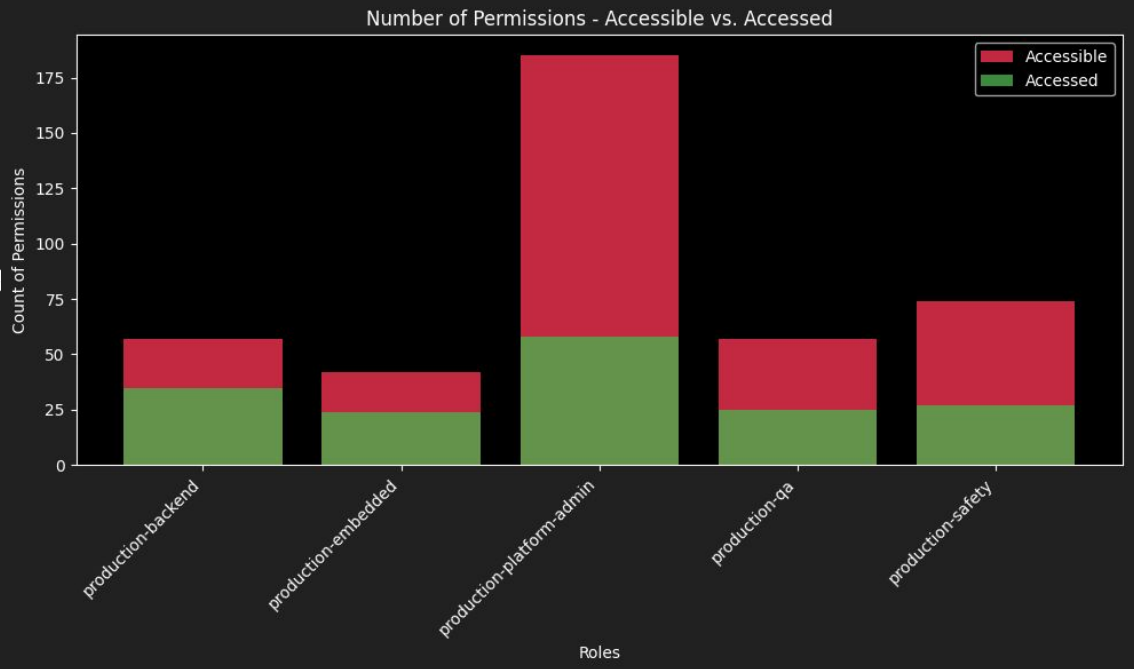
UPR: 0.46

Split Roles

UPR: 0.46

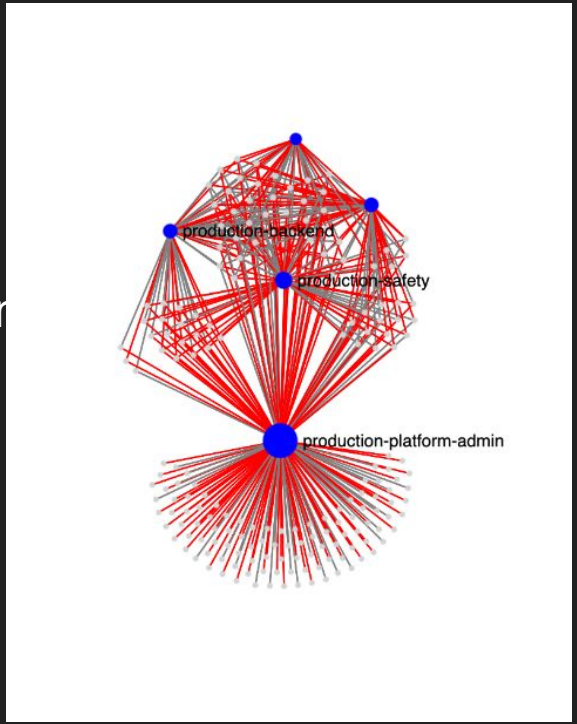
Permission Reduction

AWS Setup: P



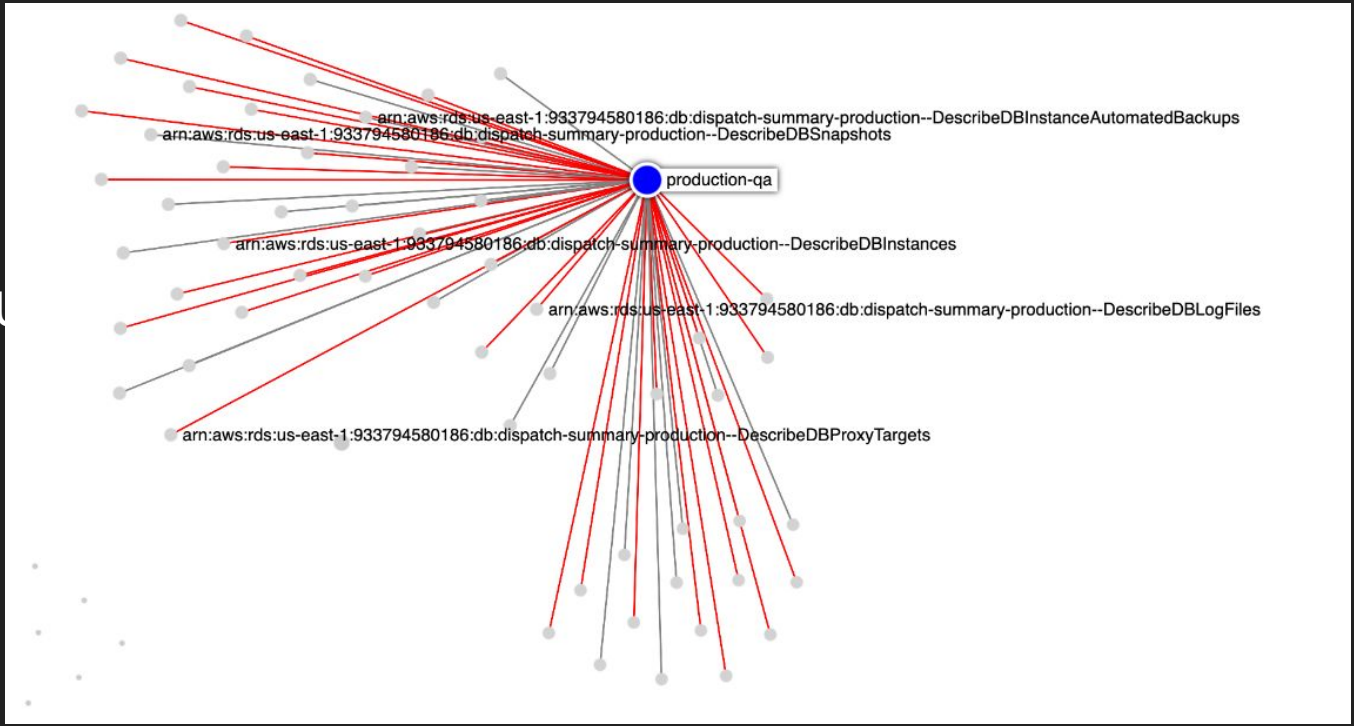
Permission Reduction

Remove Unused Permission



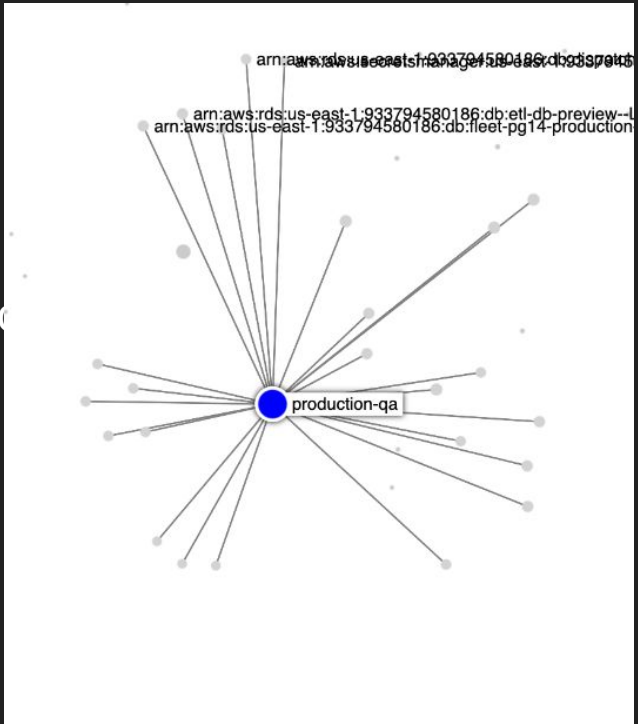
Permission Reduction

Remove U



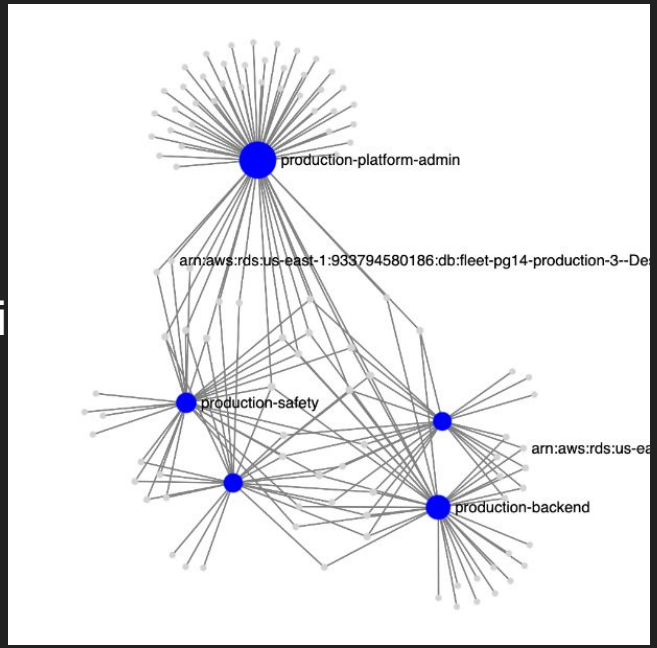
Permission Reduction

Remove Unused Permissi



Permission Reduction

Remove Unused Permissi



UPR: 0.0

Permission Reduction

Talk about as the companies go big, the scope decreases and number of resources people need access to decreases. Talk about platform team example.

Split the roles

Show the network graph + new UPR

Result

Show the network graph + new UPR

Result

- How can i do the same thing in my company

Key Takeaways

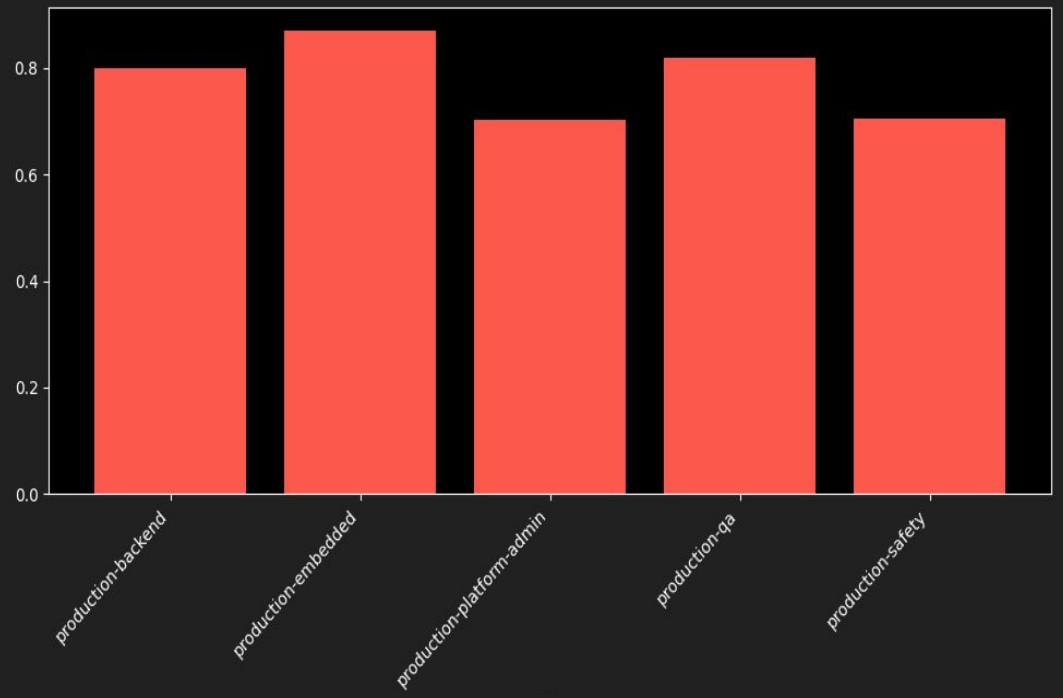
Permission Reduction

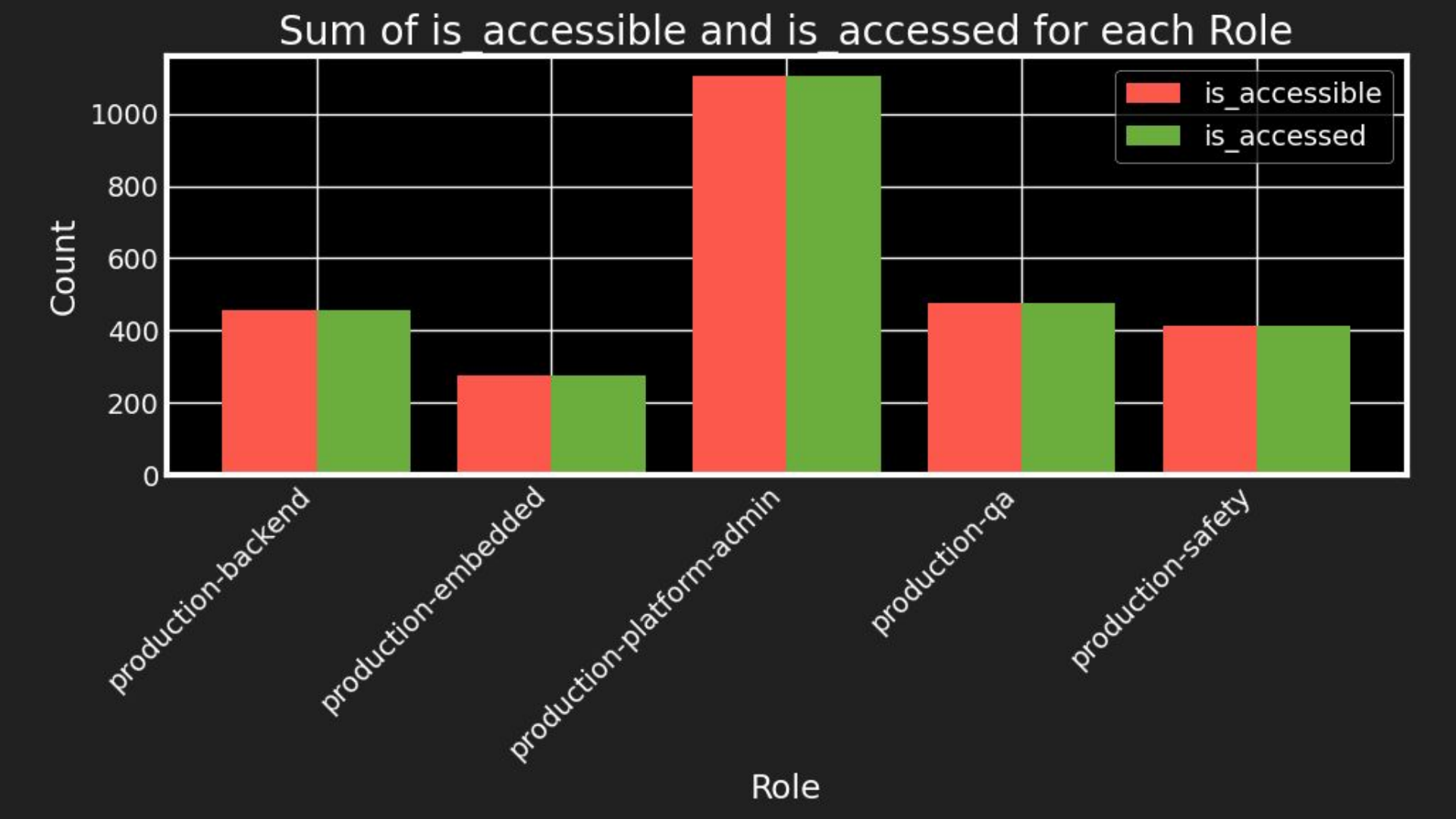
1. Get the data in the schema - github spec.
2. Tools - python + ipysigma to build visualizations. link

How do i do this.

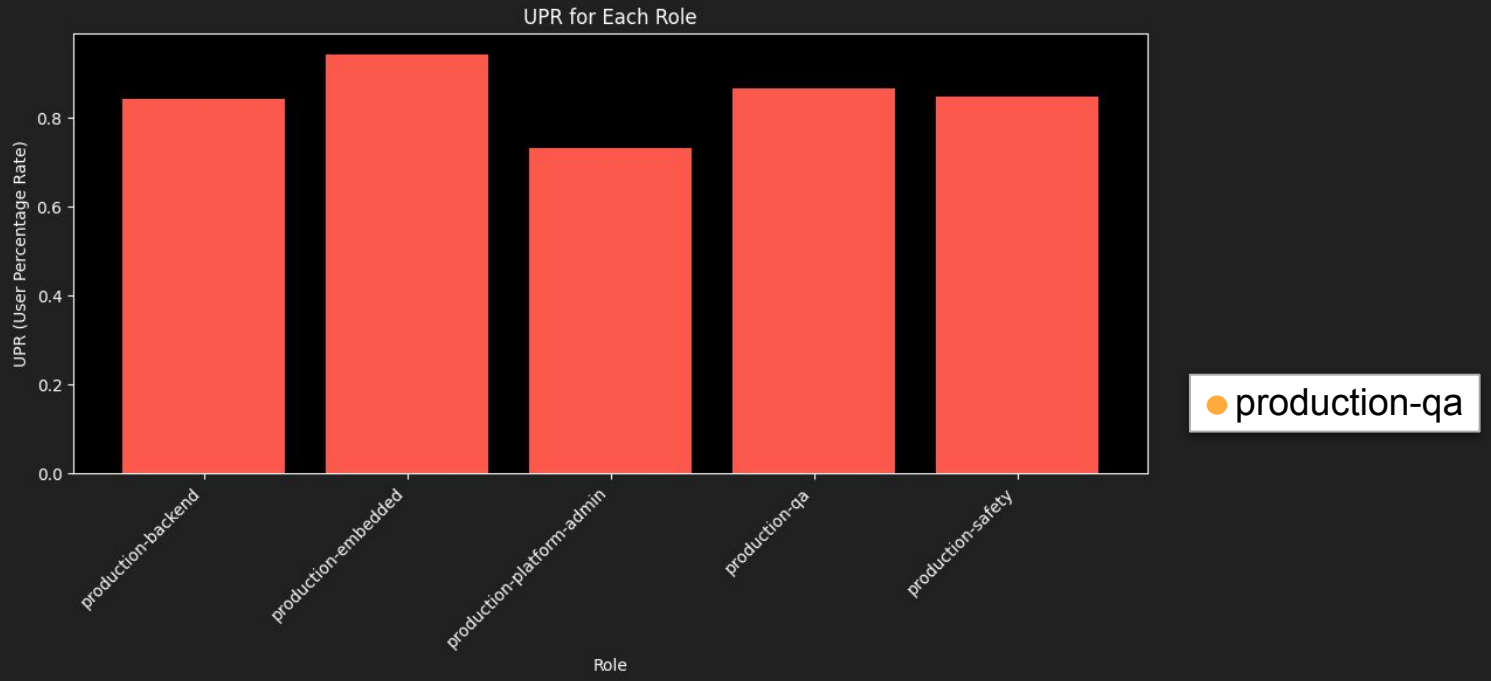
Strategy: Remove Unused Permissions

Overall UPR
0.93 → 0.78




Sum of is_accessible and is_accessed for each Role
is_accessible
is_accessed
Count
0
200
400
600
800
1000
production-backend
production-embedded
production-platform-admin
production-qa
production-safety
Role

Strategy 3 - UPR df_isaccess=1
0.85



Background Tones

(35, 47, 60)	(59, 76, 89)	(78, 88, 98)	(73, 86, 90)
Base01	Base02	Base03	Base04

Foreground Tones

(143, 168, 194)	(250, 250, 250)
Base2	Base1

Accent Colors

(253, 152, 49)	(249, 90, 83)	(252, 85, 139)	(157, 113, 249)	(78, 134, 249)	(34, 163, 196)	(92, 191, 168)	(112, 171, 74)
Orange	Red	Pink	Purple	Blue	Cyan	Turquoise	Green

Root Cause Analysis

350 services x 13.75k Permissions = Decisions

Why does it occur in AWS?

How did it happen in our company?

50 engineers working on everything → 500 engineers working on specific areas.

Roles - platform-admin, fuel, safety, cards - manage their own infra had access to pretty much everything

Then we scaled from 5 to 20 teams - but continued using the same roles as the team level roles gave us the access we needed to do the job.