MunkiDevOpsYVR

Rod Christiansen

Emily Carr University of Art + Design

Vancouver, Canada

Fully Running Your Munki Repo Operations in DevOps

Quick about me

Migration from local ops to cloud git ops

Manifests orchestration from asset inventory

The new DevOps Munki infra and architecture

Examples!

Fully Running Your Munki Repo Operations in DevOps

focused.systems/munki-devops

about.me/rodc

Studied Civil Engineering



about.me/rodc

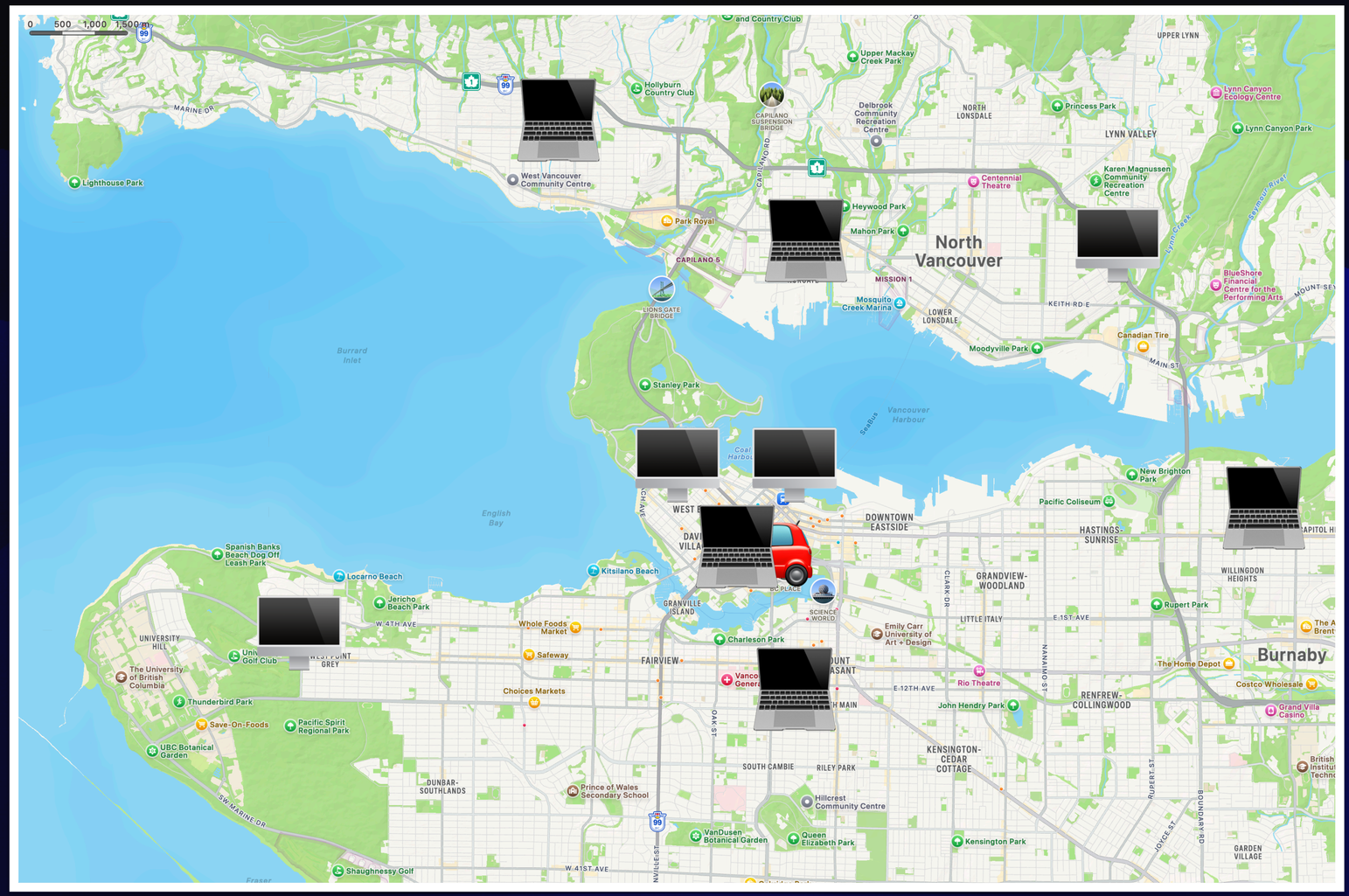
Migrated by myself from
Brazil to Canada in 2011



about.me/rodc

Mac in home





about.me/rodc

Emily Carr



MAY 27, 2014

Rod Christiansen (He/Him) • 4:44 PM

Hello [redacted]

Thanks for connecting. Is [redacted] looking to add team members currently? Let me know if a tech wiz with years of experience of Mac and Apple support is of interest. Thanks.

MAY 29, 2014

[redacted] (He/Him) • 2:05 PM

Thanks Rod!

I think we're full-up now but I do have a few clients that are looking to hire Help Desk staff and IT support. Can you send me your email address so I can introduce you?

Thanks!

about.me/rodc



[redacted]

May 30, 2014 at 13:50

Introductions

To: [redacted] & 1 more

Details

Hi [redacted] I'd like you to meet Rod Christiansen.(CC'd) Say hi everyone.

Rod currently works for a managed service provider (MacinHome) We've subcontracted with them on several projects and they always do good work.

Have a great weekend!

[redacted]

[redacted]
Corporate Sales Manager

[redacted]

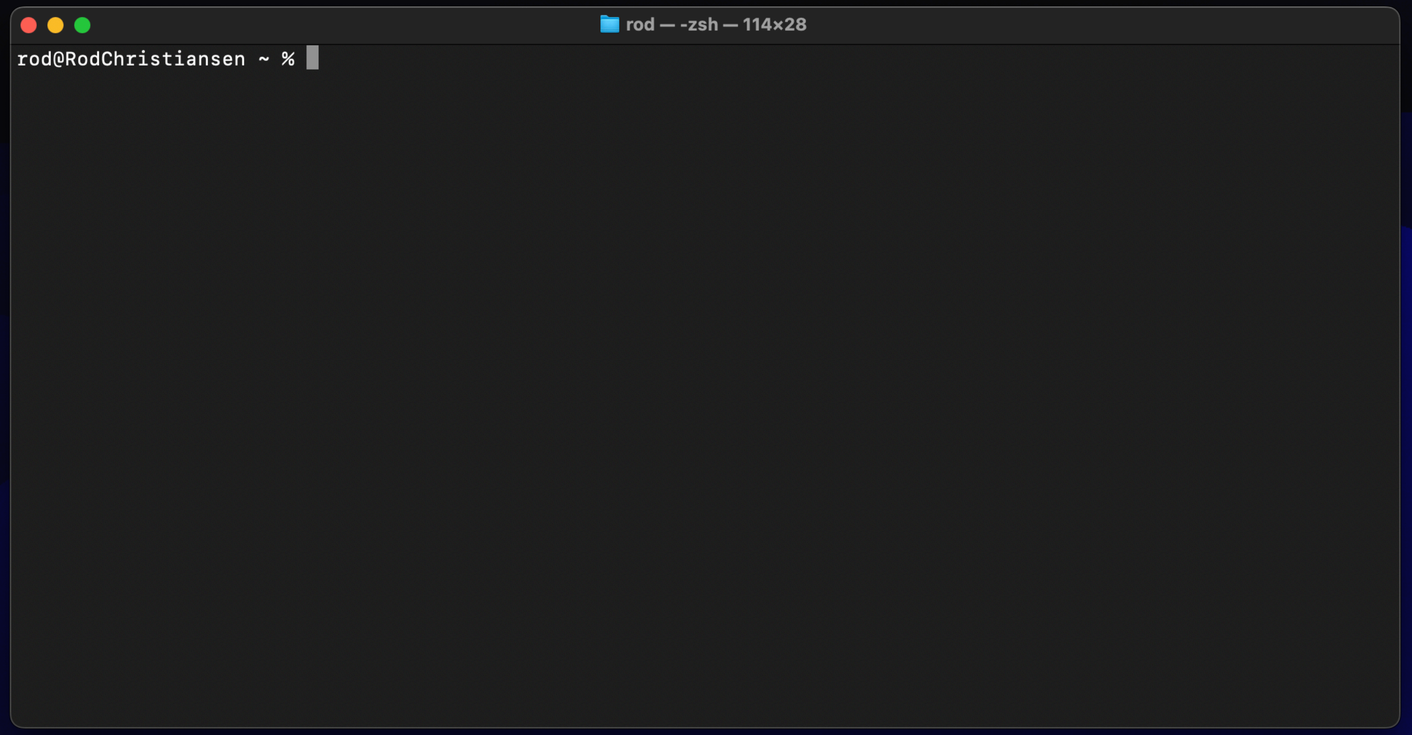
about.me/rodc





about.me/rodc

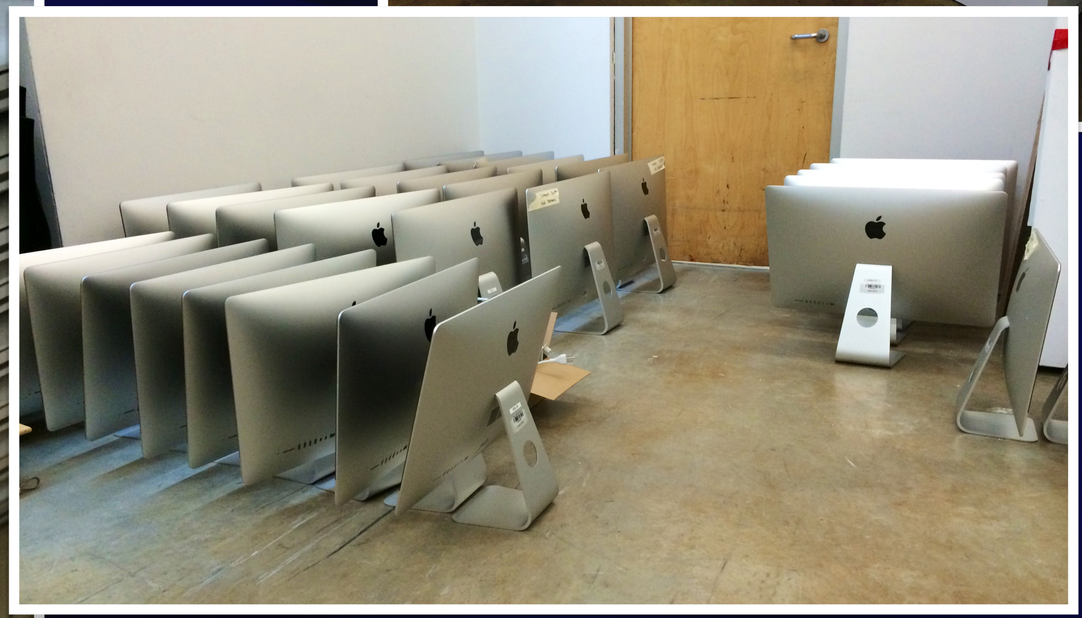





Mac Pro


iMac
iMac






MacBook Pro
MacBook Pro



2015

Learning
the ropes

NetBoot

Imaging

ARD commands

Munki



2015

Learning the ropes



2016

Profiles

Deprecated imaging!

Outset scripts

Munki Enroll scripts and robust inventory

2015

Learning the ropes



2016

Profiles

Deprecated imaging!

Outset scripts

Munki Enroll scripts and robust inventory

2017

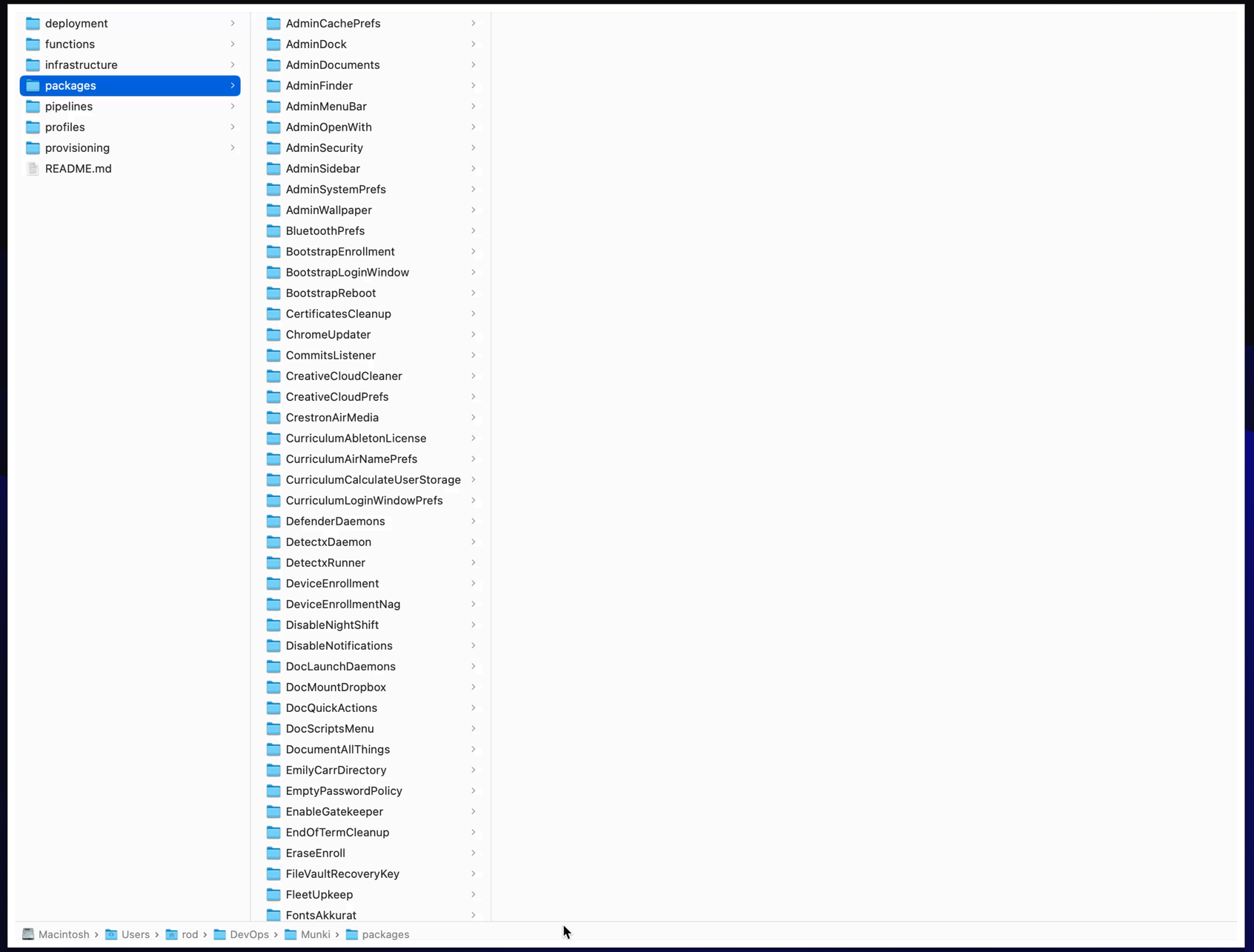
Early Provisioning Bootstrapping Workflow

- munki pkgs
- mobile configs

2017

Early
Provisioning
Bootstrapping
Workflow

- munki pkgs
- mobile configs



2016	2017
Profiles	
Deprecated imaging!	Early Provisioning Bootstrapping Workflow
Outset scripts	- munki pkgs - mobile configs
Munki Enroll scripts and robust inventory	"Hey, we should look at this MDM thing"


2016
2017
2018
2019
Profiles
Deprecated imaging!
Outset scripts
Munki Enroll scripts and robust inventory
Early Provisioning Bootstrapping Workflow
- munki pkgs
- mobile configs
MunkiReport
Fully Provisioning Bootstrapping Workflow
- apps
- printers
- prefs
"Hey, we should look at this MDM thing"
"So we MDM?"
Oh we gotta MDM

2016	2017	2018	2019
Profiles	Early Provisioning Bootstrapping Workflow	Fully Provisioning Bootstrapping Workflow	**MicroMDM** in AWS
Deprecated imaging!	- munki pkgs - mobile configs	- apps - printers - prefs	
Outset scripts	MunkiReport		MunkiReport in AWS
Munki Enroll scripts and robust inventory	"Hey, we should look at this MDM thing"	"So we MDM?"	Oh we gotta MDM



2019

Munki



Middleware
flask app



MicroMDM
in AWS



Devices



Install, update, remove mobileconfig profiles

Install and remove Managed Apps (VPP)

https://github.com/munkimdm/munkimdm

Munki



Keynote.plist

~/DevOps/Munki/deployment/pkgsinfo/apps/apple/Keynote.plist main (functions)

```
    <string>Keynote</string>
    <key>notes</key>
    <string>VPP App Store app</string>
    <key>postinstall_script</key>
    <string>#!/bin/bash
DEVICE=$(ioreg -d2 -c IOPlatformExpertDevice | awk -F\" '/IOPlatformUUID/{print $(NF-1)}')
SERIAL=$(ioreg -d2 -c IOPlatformExpertDevice | awk -F\" '/IOPlatformSerialNumber/{print $(NF-1)}')
curl --request POST \
  --url https://micromdm.ecuad.ca/api/InstallApplication \
  --header 'authorization: Basic                                \
  --header 'content-type: application/json' \
  --data '{
    "request_type":"InstallApplication",
    "udid" : "'"$DEVICE"'",
    "serial" : "'"$SERIAL"'",
    "itunes_store_id" : "409183694"
}'</string>
    <key>receipts</key>
    <array>
        <dict>
            <key>packageid</key>
            <string>com.apple.iWork.Keynote</string>
            <key>version</key>
            <string>14.4</string>
        </dict>
    </array>
    <key>unattended_install</key>
    <true/>
    <key>unattended_uninstall</key>
    <true/>
    <key>uninstall_method</key>
    <string>uninstall_script</string>
    <key>uninstall_script</key>
    <string>#!/bin/bash
rm -rf /Applications/Keynote.app
DEVICE=$(ioreg -d2 -c IOPlatformExpertDevice | awk -F\" '/IOPlatformUUID/{print $(NF-1)}')
SERIAL=$(ioreg -d2 -c IOPlatformExpertDevice | awk -F\" '/IOPlatformSerialNumber/{print $(NF-1)}')
curl --request POST \
  --url https://micromdm.ecuad.ca/api/DisassociateApplication
  --header 'authorization: Basic                                \
  --header 'content-type: application/json' \
  --data '{
    "request_type":"DisassociateApplication",
    "udid" : "'"$DEVICE"'",
    "serial" : "'"$SERIAL"'",
    "itunes_store_id" : "409183694"
}'</string>
    <key>uninstallable</key>
    <true/>
    <key>version</key>
    <string>14.4</string>
</dict>
</plist>
```

2019

MicroMDM
in AWS



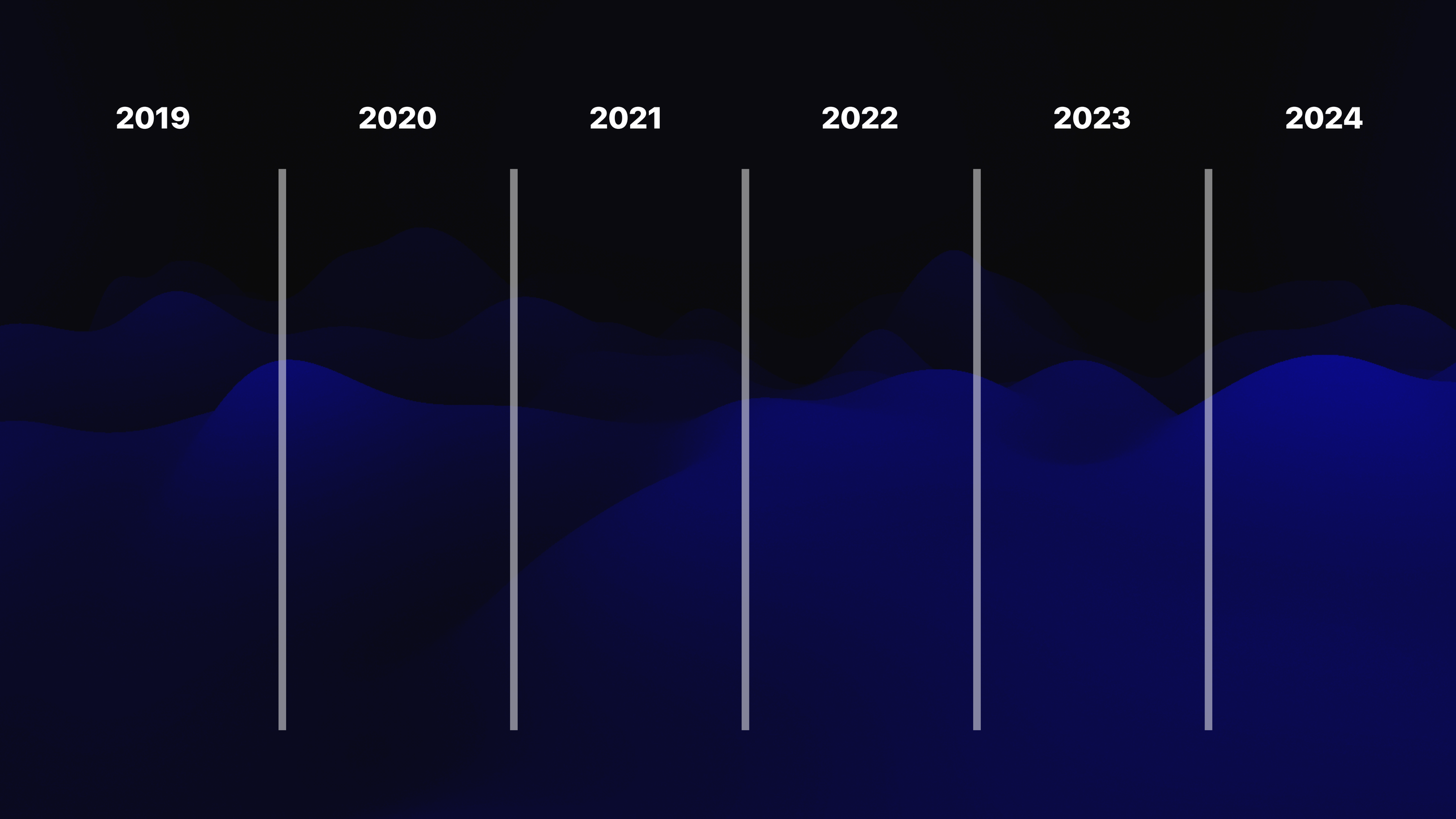
2020

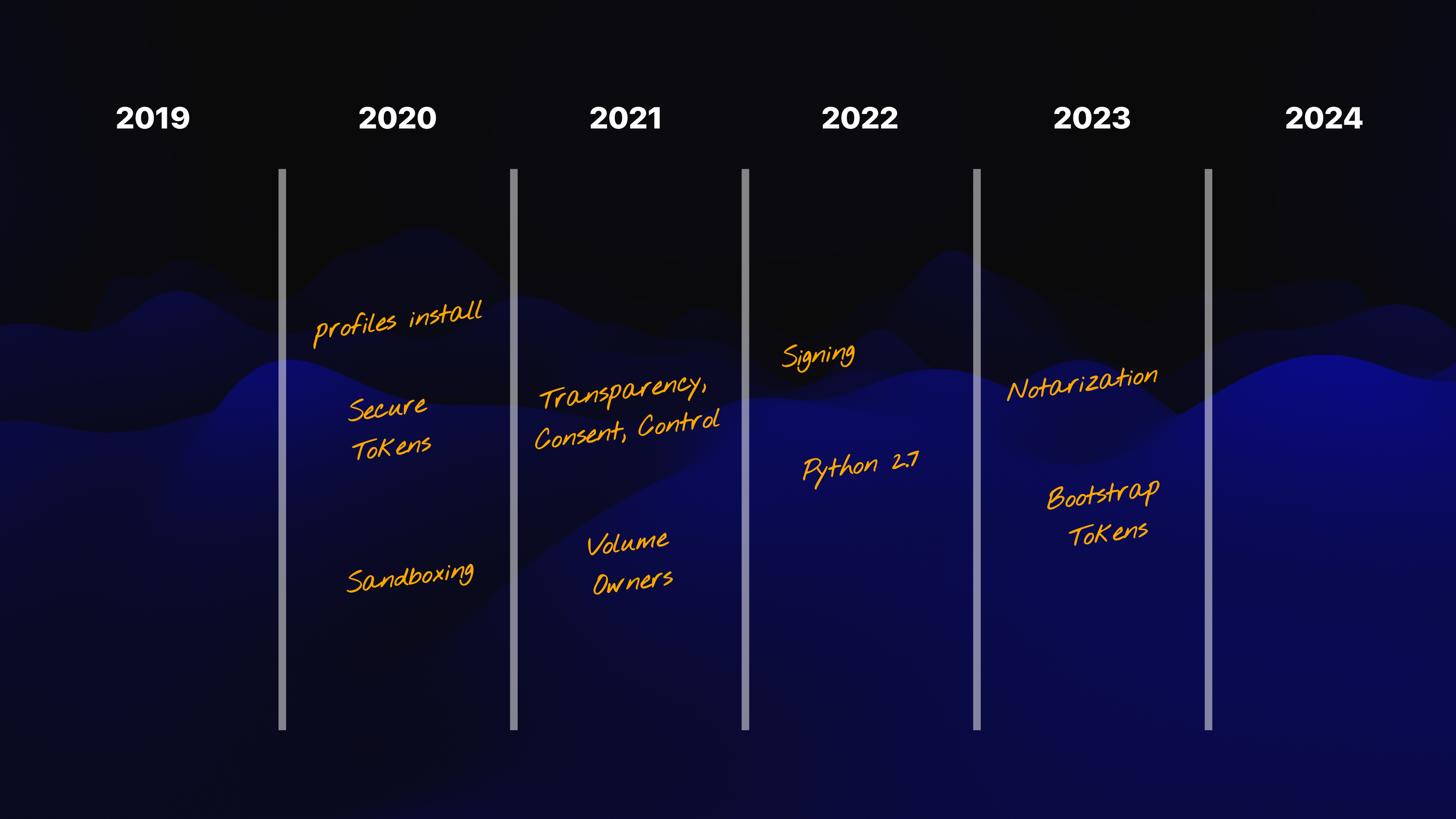
Munki in
the AWS

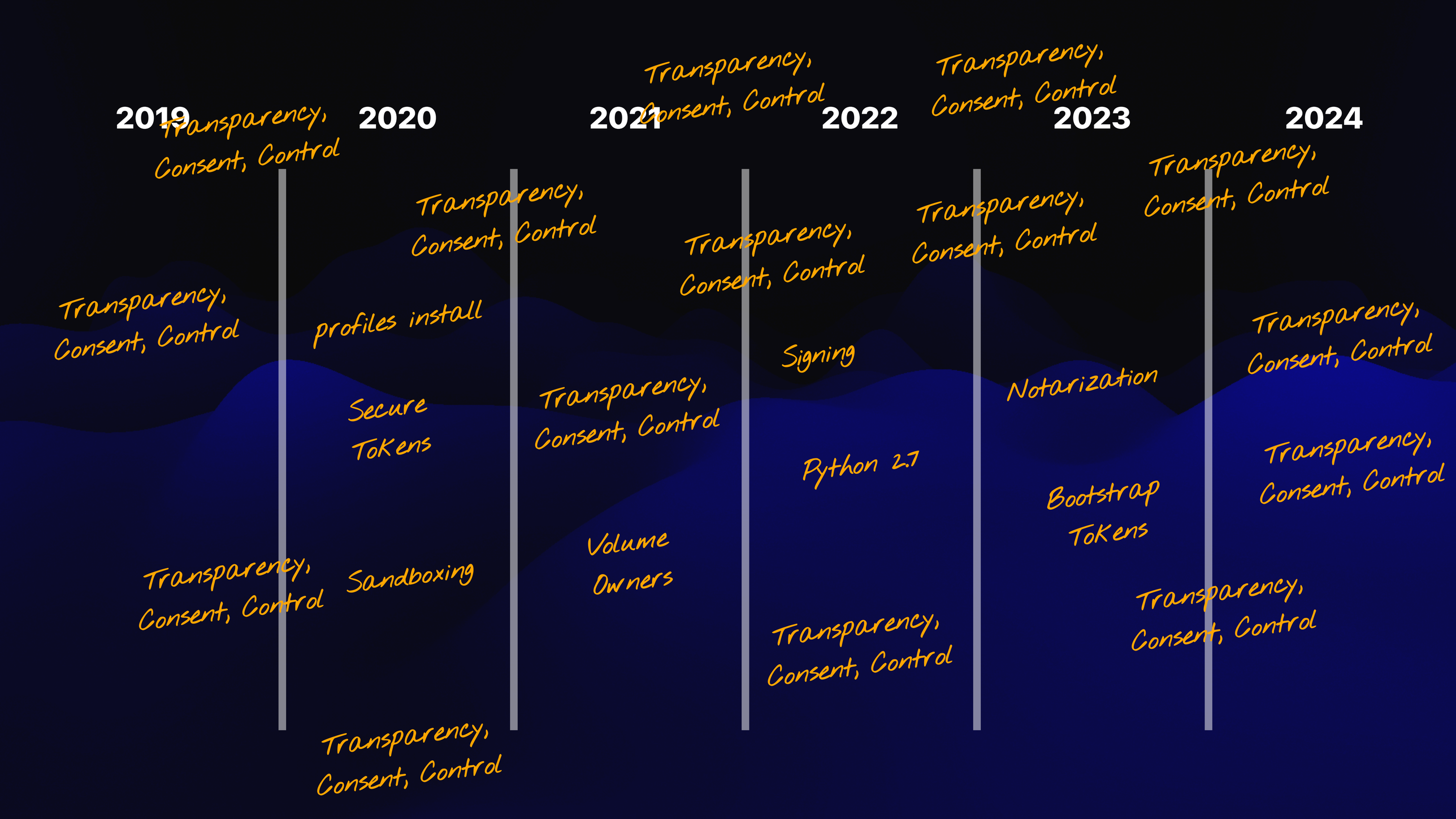


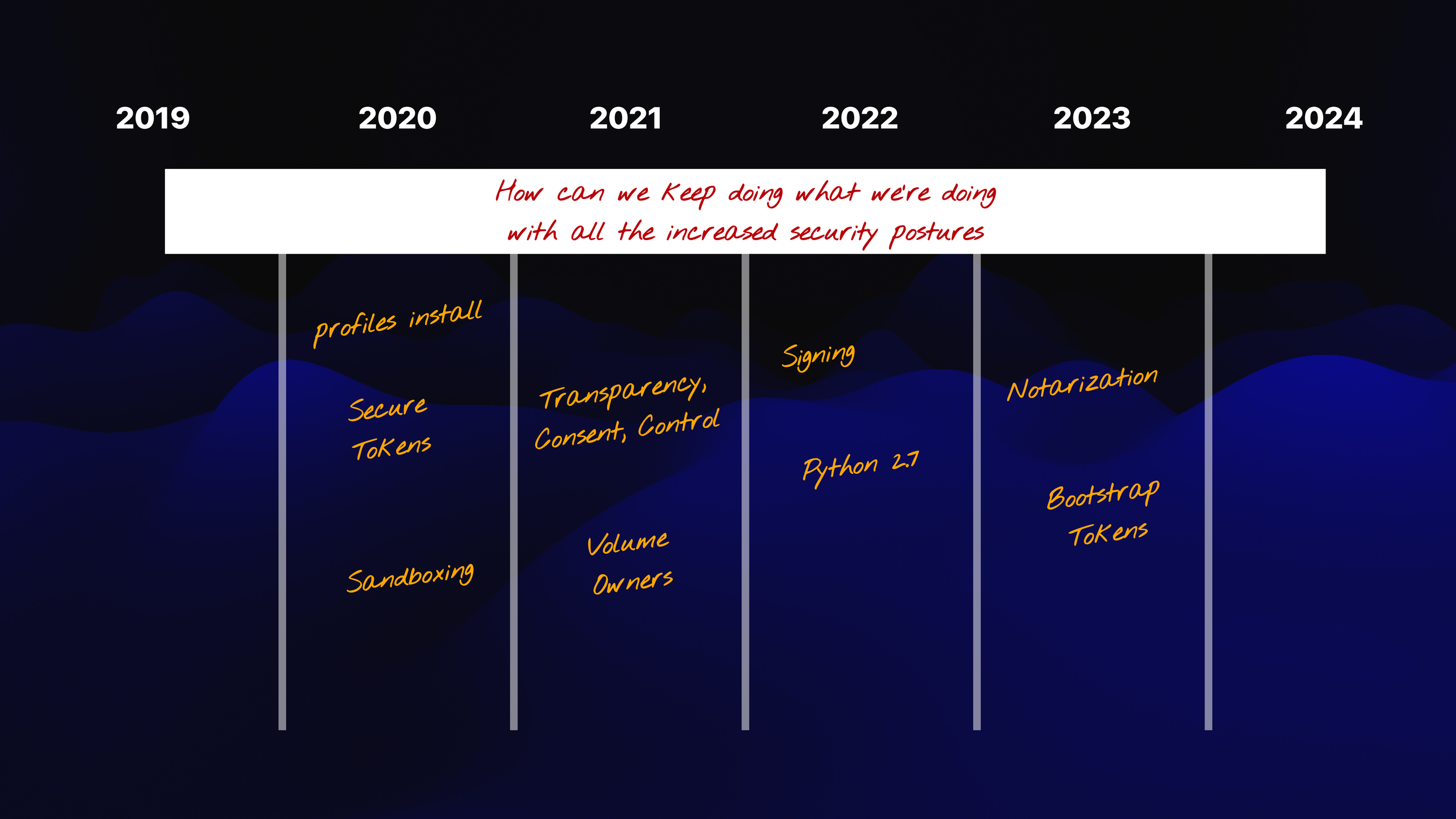


Terraform


2019
2020
2021
2022
2023
2024


2019
2020
2021
2022
2023
2024
profiles install
Secure Tokens
Sandboxing
Transparency, Consent, Control
Volume Owners
Signing
Python 2.7
Notarization
Bootstrap Tokens


2019
Transparency, Consent, Control
Transparency, Consent, Control
Transparency, Consent, Control
2020
Transparency, Consent, Control
profiles install
Secure Tokens
Sandboxing
Transparency, Consent, Control
2021
Transparency, Consent, Control
Transparency, Consent, Control
Transparency, Consent, Control
Volume Owners
2022
Transparency, Consent, Control
Signing
Python 2.7
Transparency, Consent, Control
2023
Transparency, Consent, Control
Notarization
Bootstrap Tokens
2024
Transparency, Consent, Control
Transparency, Consent, Control
Transparency, Consent, Control
Transparency, Consent, Control


2019
2020
2021
2022
2023
2024
How can we keep doing what we're doing with all the increased security postures
profiles install
Secure Tokens
Sandboxing
Transparency, Consent, Control
Volume Owners
Signing
Python 2.7
Notarization
Bootstrap Tokens

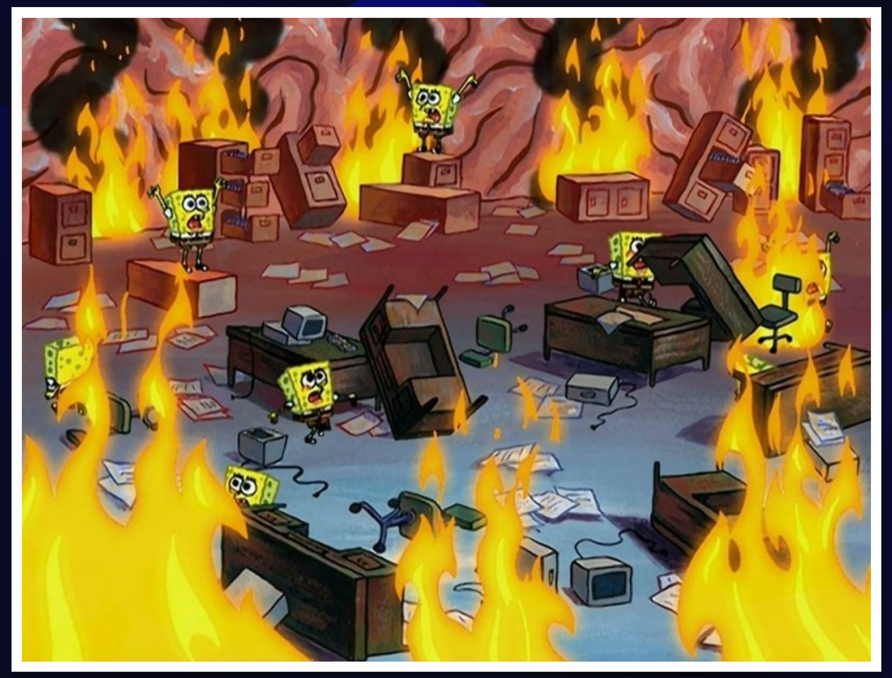
Nov 2022

Transition to Devices Admin

Macs + **Windows**



Windows



Nothing in source control

Still monolith imaging

Still tied to on prem GPO

Zero off site device connection (except if on VPN)

No MDM

Two aging local on prem servers

Dependant on 3rd party vendor for imaging and software deployment

Black box. No idea who did what, when, where, how.

Near zero documentation and what existed was on....Word files. Word. Files. The horror.



Mar 2023

University Migrated to Microsoft

2022



Devices Admin

Now managing Windows endpoints

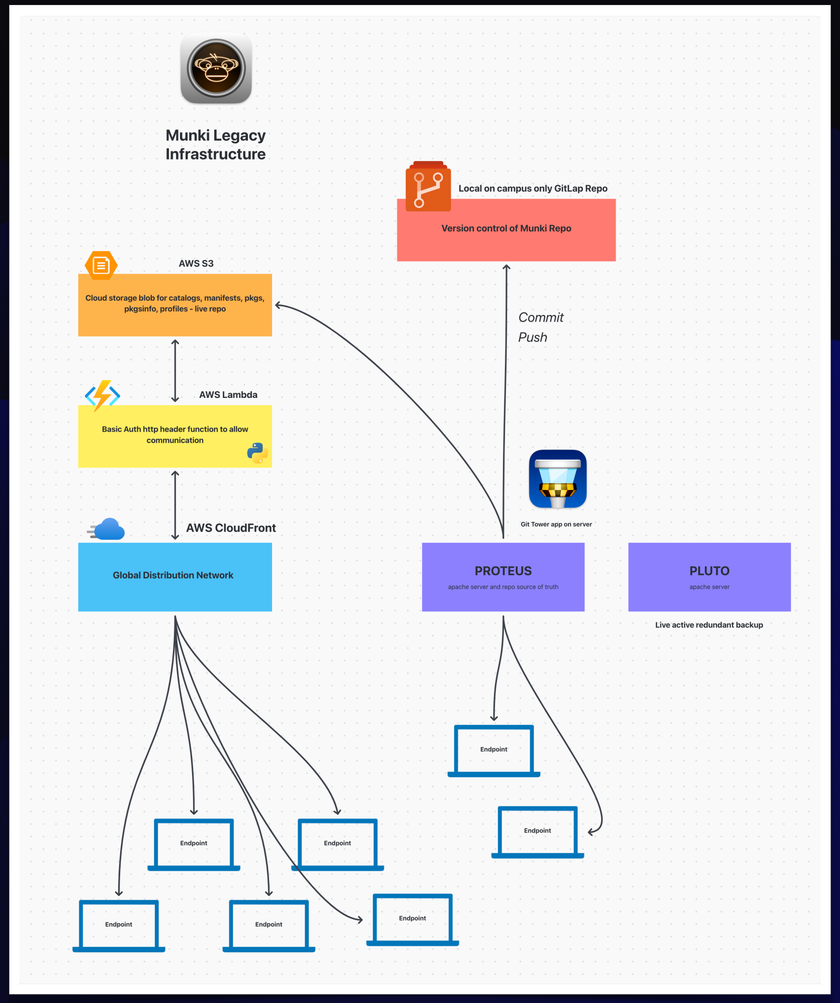
2023



University Migrated to Microsoft

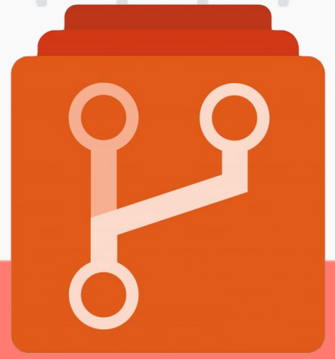
2024

2016 → 2024


Munki Legacy Infrastructure
Local on campus only GitLap Repo
Version control of Munki Repo
AWS S3
Cloud storage blob for catalogs, manifests, pkgs, pkgsinfo, profiles - live repo
Commit
Push
AWS Lambda
Basic Auth http header function to allow communication
Git Tower app on server
AWS CloudFront
Global Distribution Network
PROTEUS
apache server and repo source of truth
PLUTO
apache server
Live active redundant backup
Endpoint
Endpoint
Endpoint
Endpoint
Endpoint
Endpoint
Endpoint

egacy
cture

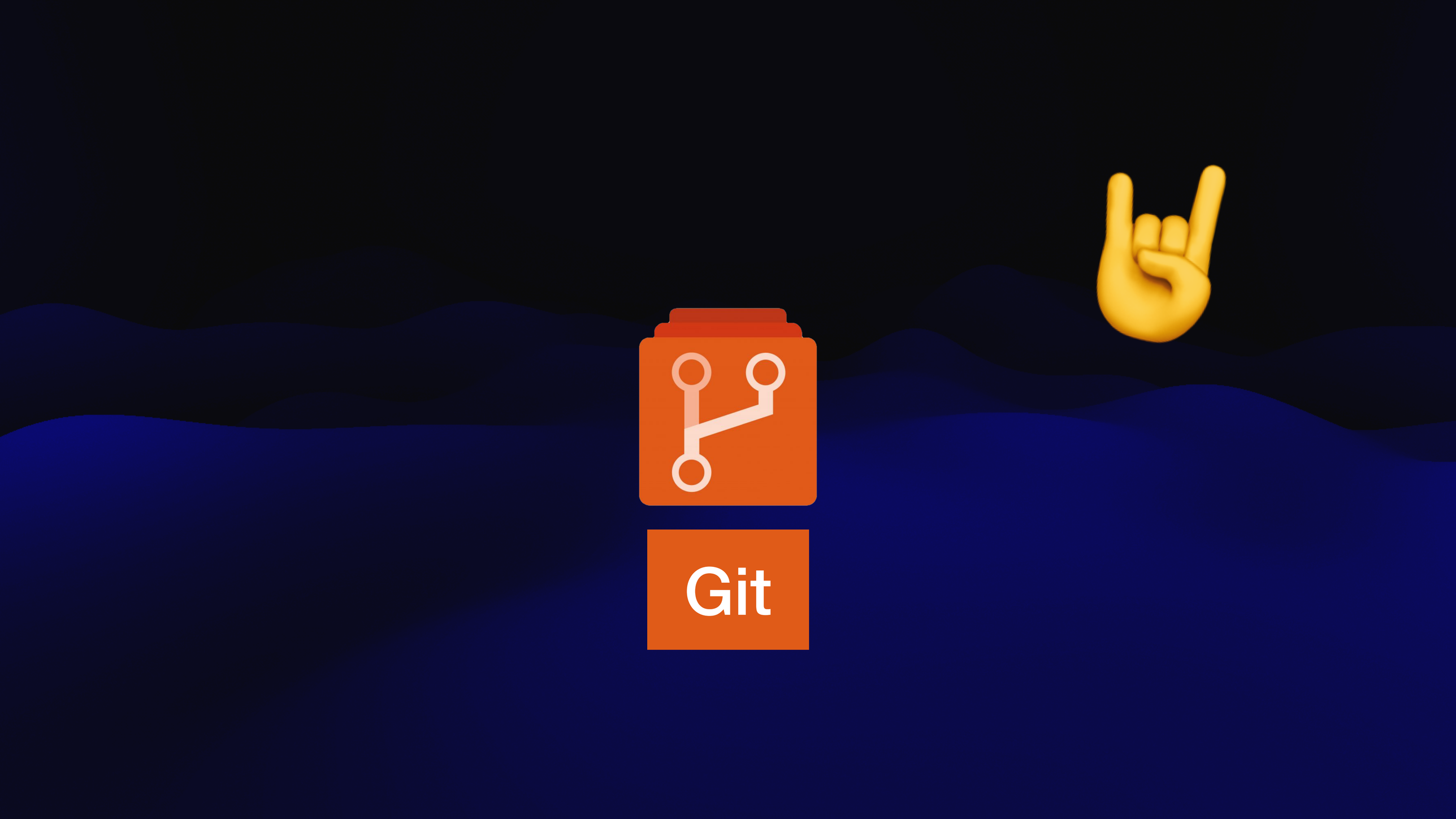
Repo & Documentation



Local on campus only GitLap Repo

Version control of Munki Repo

ests, pkgs

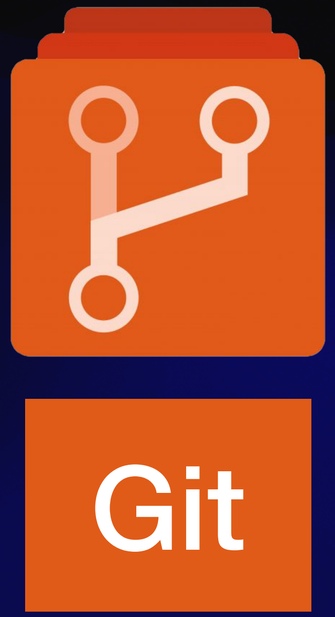

Git

DevOps Benefits and Advantages

Built-in Audit Trails

Documentation is built in to the process

Access Controls



Collaboration

Disaster Recovery

Change Management

Rollback Capabilities

Scalability

egacy
cture

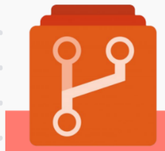


Local on campus only GitLap Repo

Version control of Munki Repo

ests_pkgs

Munki Legacy Infrastructure



Local on campus only GitLap Repo

Version control of Munki Repo



AWS S3

Cloud storage blob for catalogs, manifests, pkgs, pkgsinfo, profiles - live repo

Commit

Push



AWS Lambda

Basic Auth http header function to allow communication





Git Tower app on server



AWS CloudFront

Global Distribution Network



PROTEUS

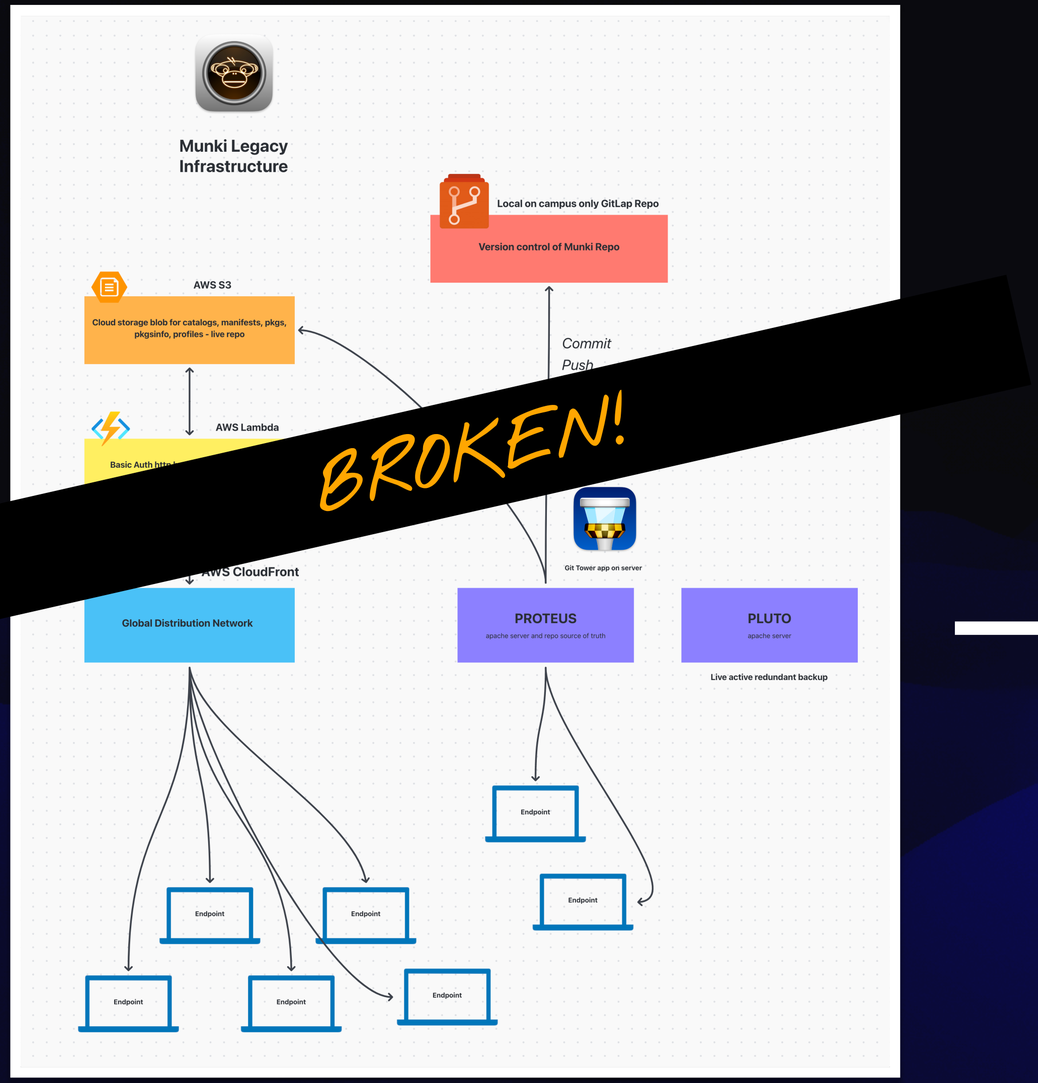
apache server and repo source of truth



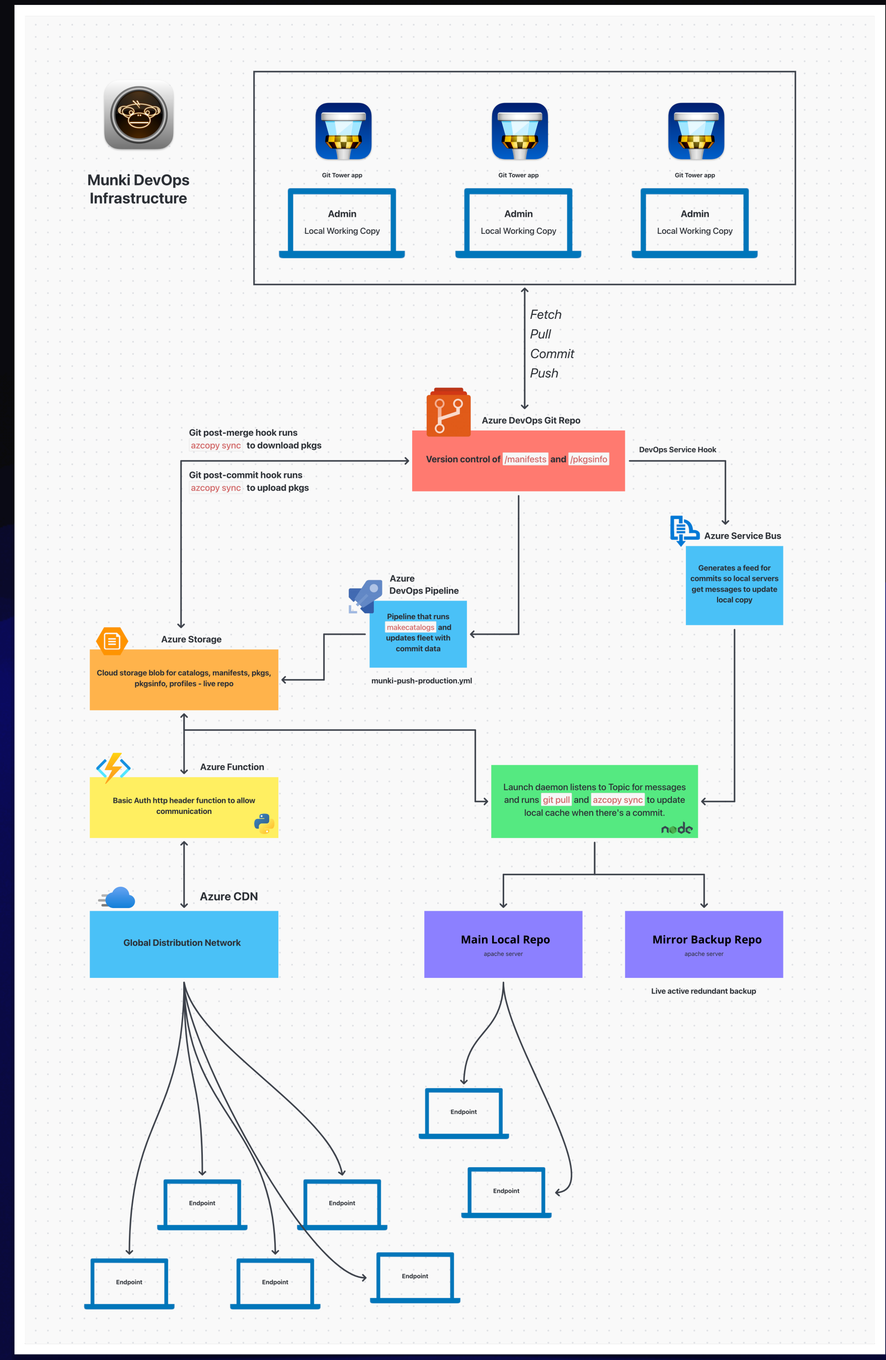
PLUTO

apache server

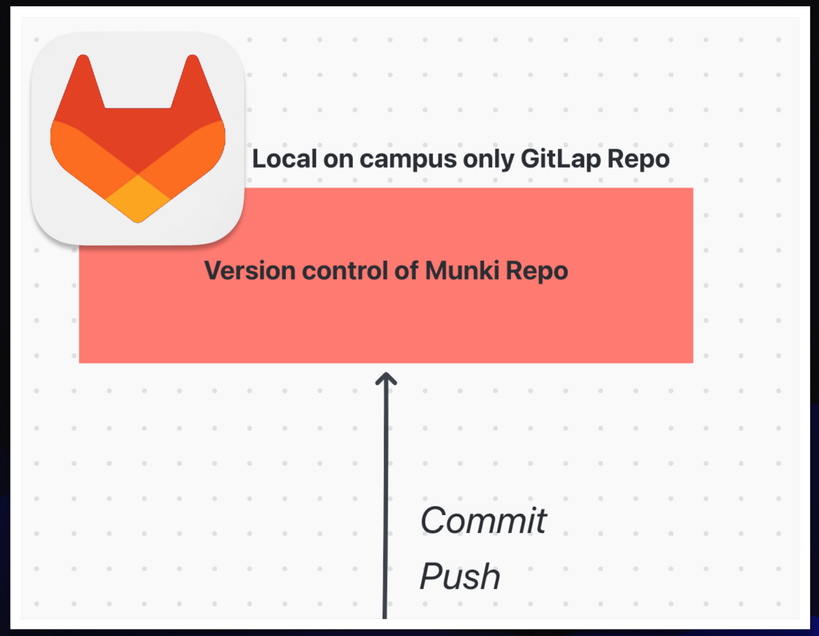
Live active redundant backup


Munki Legacy Infrastructure
Local on campus only GitLap Repo
Version control of Munki Repo
AWS S3
Cloud storage blob for catalogs, manifests, pkgs, pkgsinfo, profiles - live repo
Commit
Push
AWS Lambda
Basic Auth http
BROKEN!
Git Tower app on server
AWS CloudFront
Global Distribution Network
PROTEUS
apache server and repo source of truth
PLUTO
apache server
Live active redundant backup
Endpoint
Endpoint
Endpoint
Endpoint
Endpoint
Endpoint
Endpoint

2024


Munki DevOps Infrastructure
Git Tower app
Admin
Local Working Copy
Git Tower app
Admin
Local Working Copy
Git Tower app
Admin
Local Working Copy
Fetch
Pull
Commit
Push
Azure DevOps Git Repo
Git post-merge hook runs
azcopy sync to download pkgs
Git post-commit hook runs
azcopy sync to upload pkgs
Version control of /manifests and /pkgsinfo
DevOps Service Hook
Azure Service Bus
Generates a feed for commits so local servers get messages to update local copy
Azure DevOps Pipeline
Pipeline that runs makecatalogs and updates fleet with commit data
munki-push-production.yml
Azure Storage
Cloud storage blob for catalogs, manifests, pkgs, pkgsinfo, profiles - live repo
Azure Function
Basic Auth http header function to allow communication
Launch daemon listens to Topic for messages and runs git pull and azcopy sync to update local cache when there's a commit.
node
Azure CDN
Global Distribution Network
Main Local Repo
apache server
Mirror Backup Repo
apache server
Live active redundant backup
Endpoint
Endpoint
Endpoint
Endpoint
Endpoint
Endpoint
Endpoint

Local GitLab VM



Running on centOS

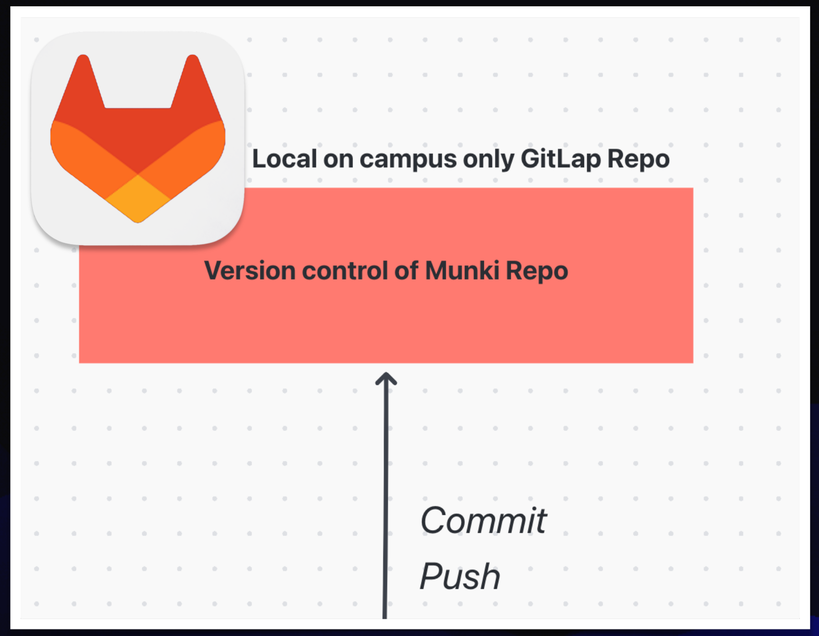
Managed by our server team

Constant patching and vulnerabilities.

First migrated it to Azure as a legacy VM + DB cloud infra -- costly and similar setup

Trying to figure out GitLab Runners gave me a headache.

Cloud GitLab VM

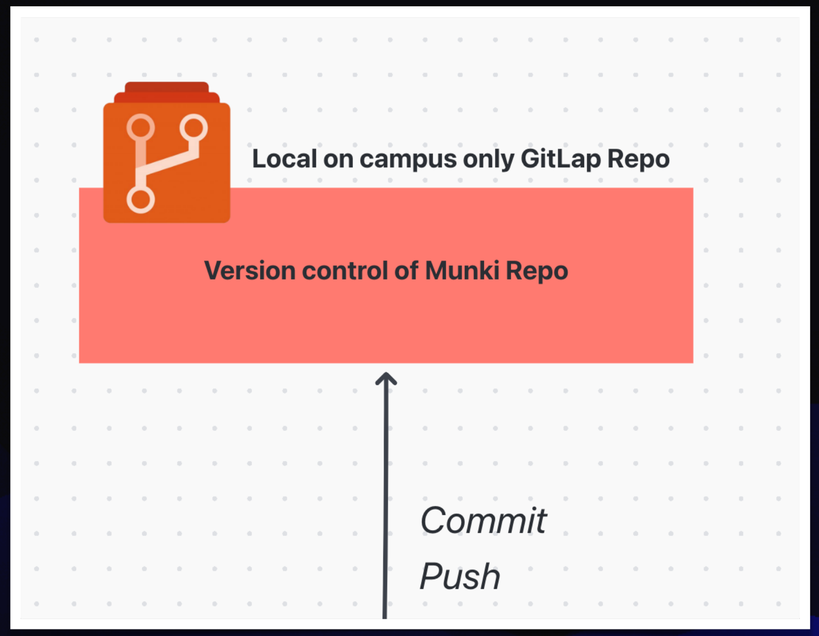


Figuring out runners

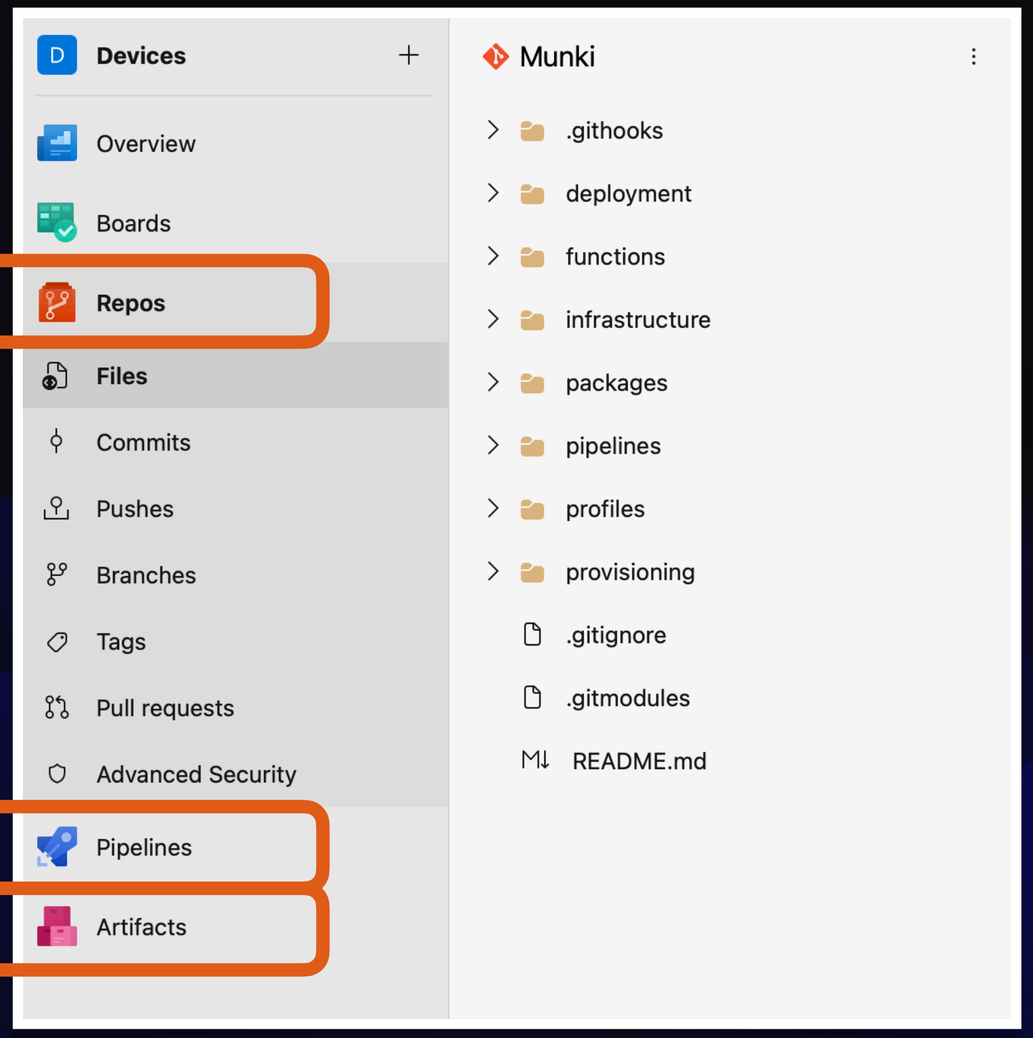
Kubernetes clusters



Where do we move to?



Azure DevOps



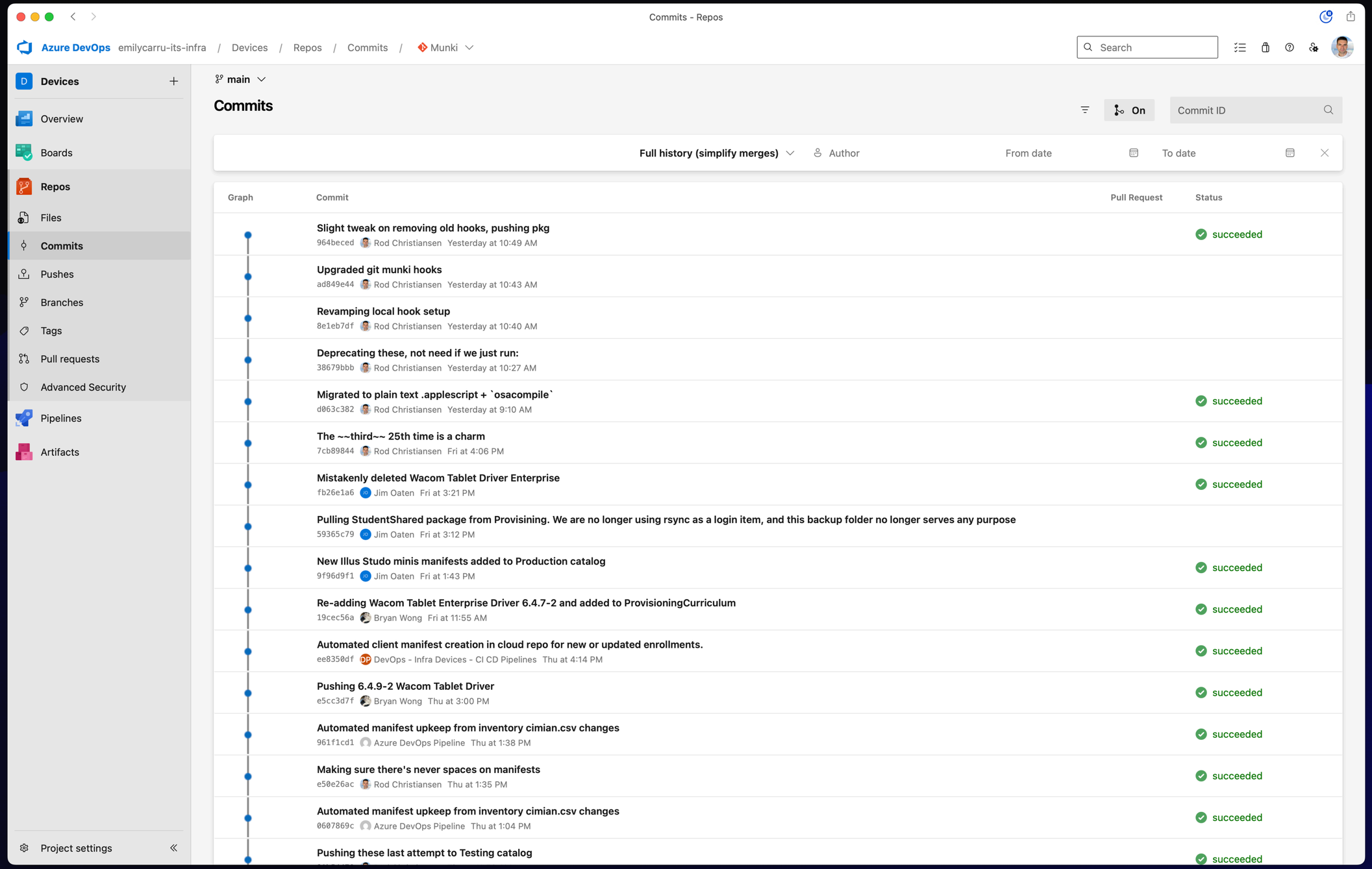
Azure DevOps

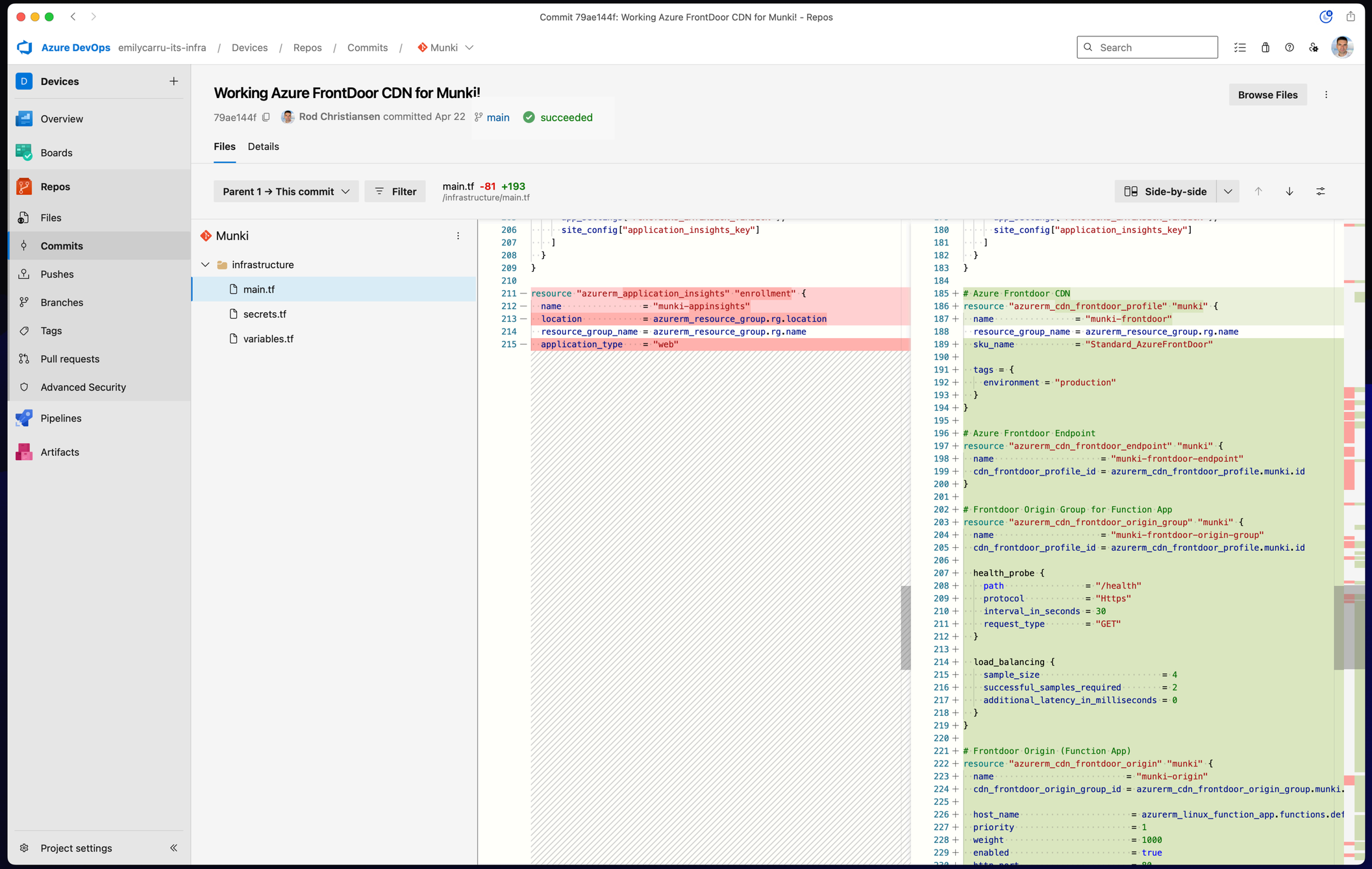
Simple!

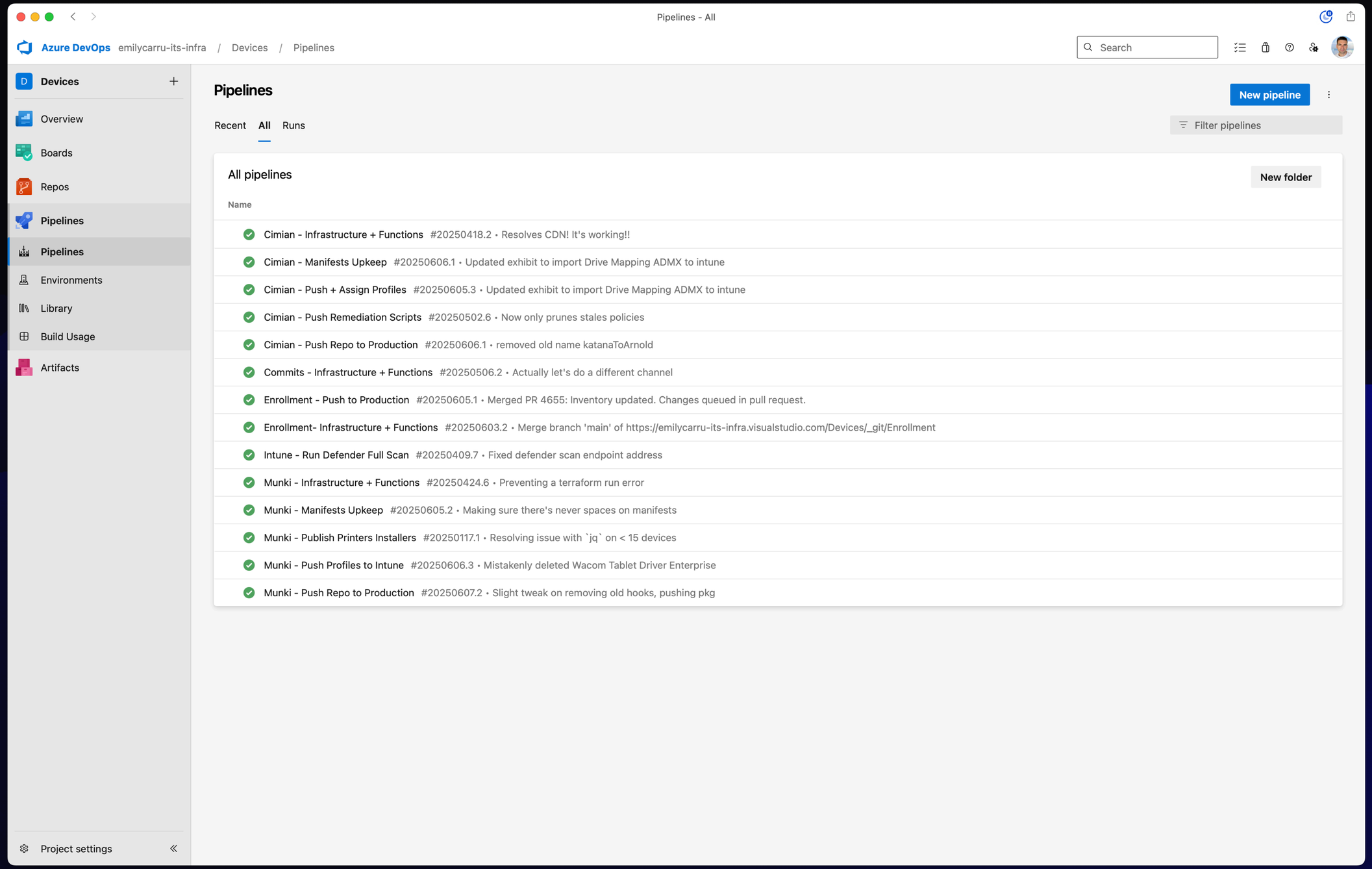
Everything I needed out of the box: Repos, Pipelines, Artifacts, Boards.

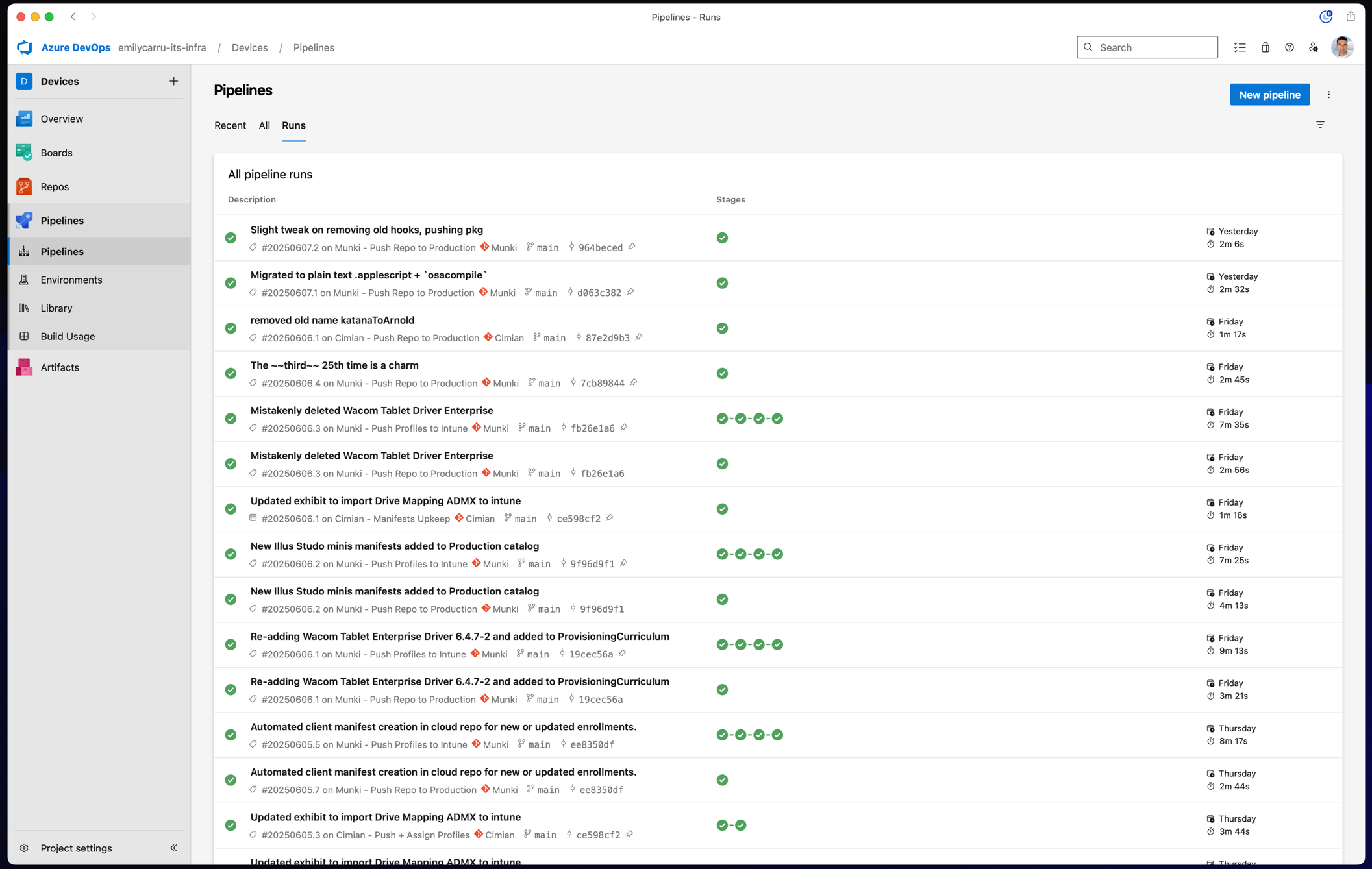
Entra SSO sign ins -- easy internal link sharing and permission mgmt

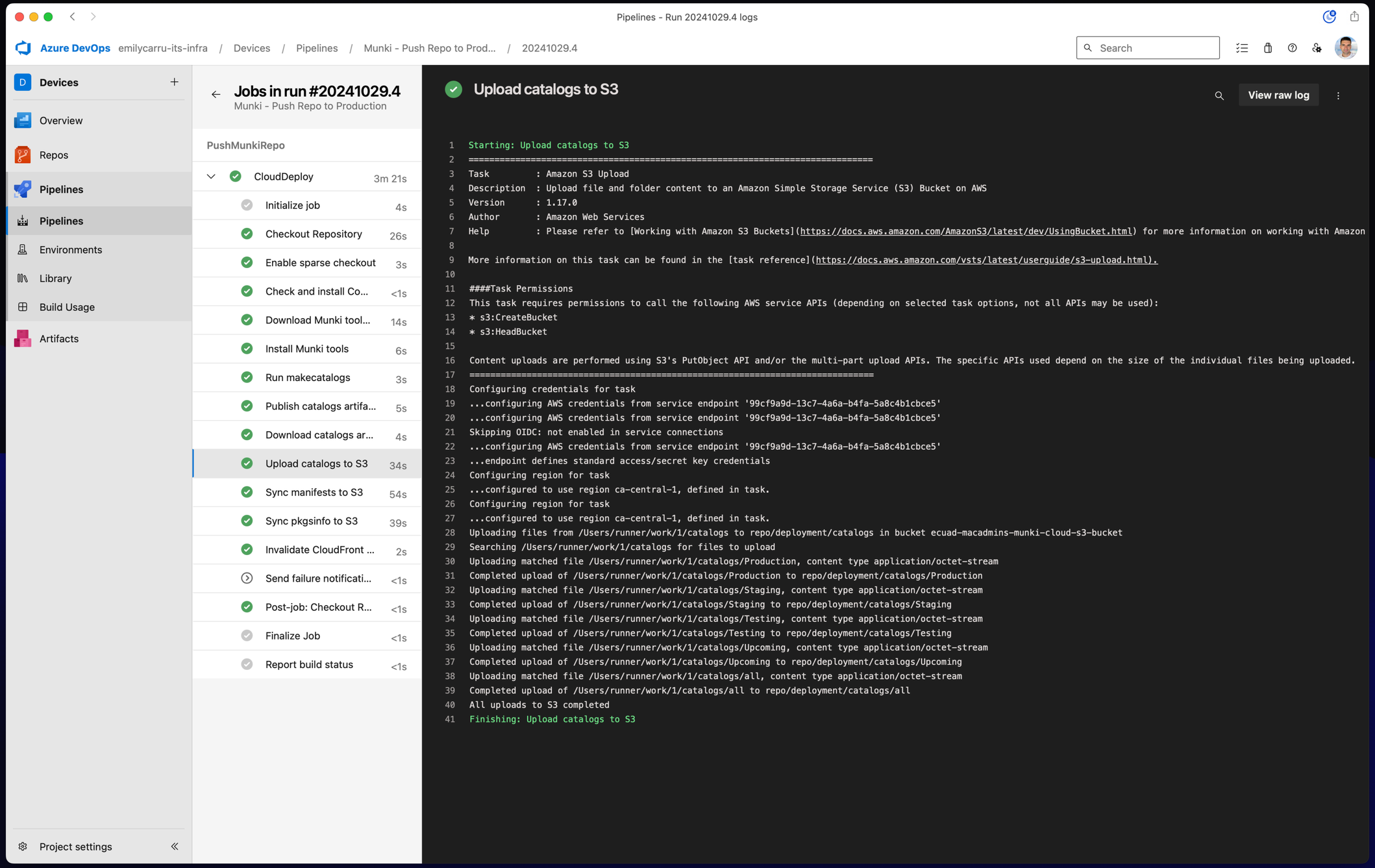
Robust `az devops` and Graph APIs

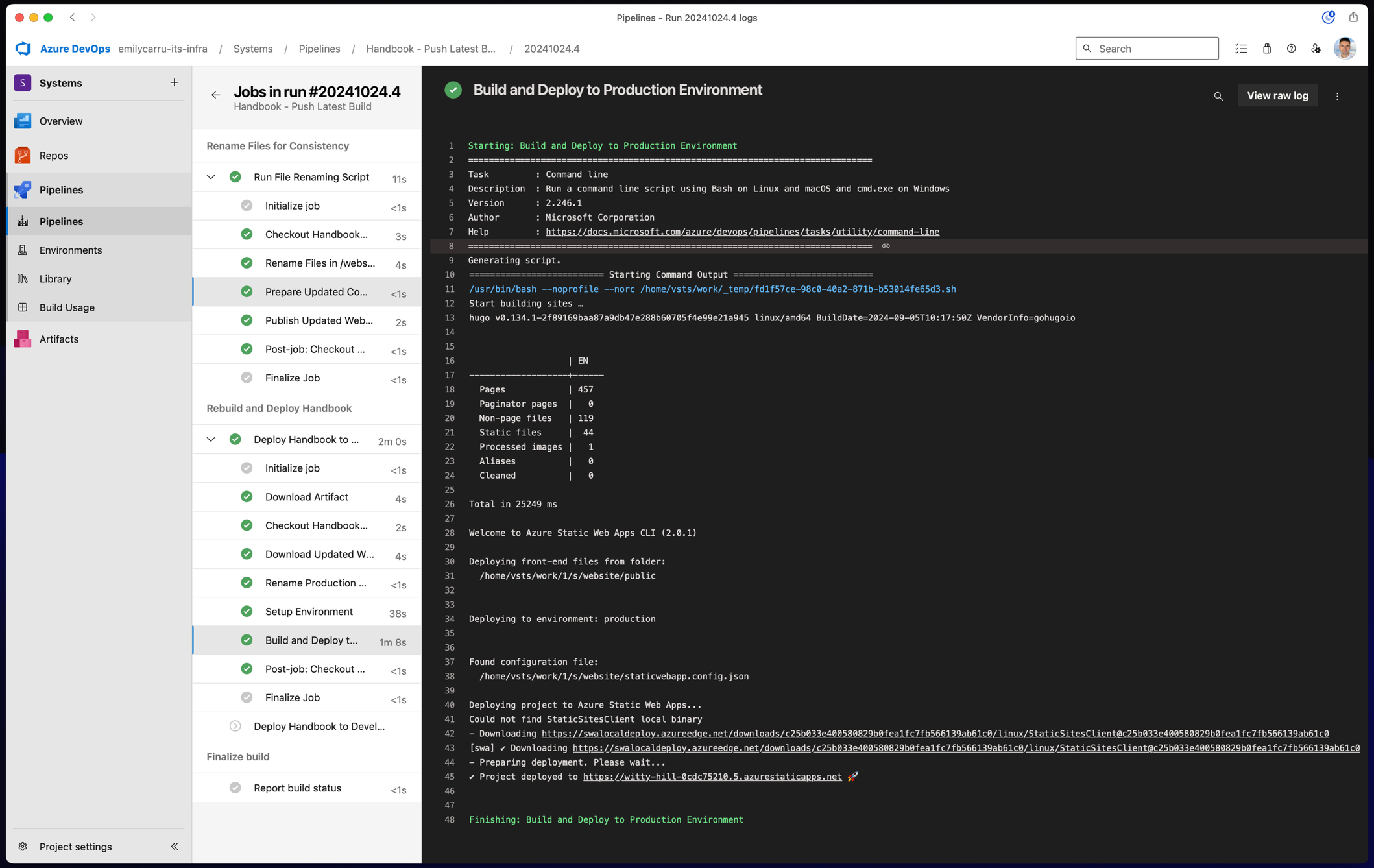

Commits - Repos
Azure DevOps emilycarru-its-infra / Devices / Repos / Commits / Munki
Search
Devices
Overview
Boards
Repos
Files
Commits
Pushes
Branches
Tags
Pull requests
Advanced Security
Pipelines
Artifacts
Project settings
main
Commits
On
Commit ID
Full history (simplify merges)
Author
From date
To date
Graph
Commit
Pull Request
Status
Slight tweak on removing old hooks, pushing pkg
964beced Rod Christiansen Yesterday at 10:49 AM
succeeded
Upgraded git munki hooks
ad849e44 Rod Christiansen Yesterday at 10:43 AM
Revamping local hook setup
8e1eb7df Rod Christiansen Yesterday at 10:40 AM
Deprecating these, not need if we just run:
38679bbb Rod Christiansen Yesterday at 10:27 AM
Migrated to plain text .applescript + `osacompile`
d063c382 Rod Christiansen Yesterday at 9:10 AM
succeeded
The ~~third~~ 25th time is a charm
7cb89844 Rod Christiansen Fri at 4:06 PM
succeeded
Mistakenly deleted Wacom Tablet Driver Enterprise
fb26e1a6 Jim Oaten Fri at 3:21 PM
succeeded
Pulling StudentShared package from Provisining. We are no longer using rsync as a login item, and this backup folder no longer serves any purpose
59365c79 Jim Oaten Fri at 3:12 PM
New Illus Studo minis manifests added to Production catalog
9f96d9f1 Jim Oaten Fri at 1:43 PM
succeeded
Re-adding Wacom Tablet Enterprise Driver 6.4.7-2 and added to ProvisioningCurriculum
19cec56a Bryan Wong Fri at 11:55 AM
succeeded
Automated client manifest creation in cloud repo for new or updated enrollments.
ee8350df DevOps - Infra Devices - CI CD Pipelines Thu at 4:14 PM
succeeded
Pushing 6.4.9-2 Wacom Tablet Driver
e5cc3d7f Bryan Wong Thu at 3:00 PM
succeeded
Automated manifest upkeep from inventory cimian.csv changes
961f1cd1 Azure DevOps Pipeline Thu at 1:38 PM
succeeded
Making sure there's never spaces on manifests
e50e26ac Rod Christiansen Thu at 1:35 PM
succeeded
Automated manifest upkeep from inventory cimian.csv changes
0607869c Azure DevOps Pipeline Thu at 1:04 PM
succeeded
Pushing these last attempt to Testing catalog
succeeded


Commit 79ae144f: Working Azure FrontDoor CDN for Munki! - Repos
Azure DevOps emilycarru-its-infra / Devices / Repos / Commits / Munki
Search
Devices
Overview
Boards
Repos
Files
Commits
Pushes
Branches
Tags
Pull requests
Advanced Security
Pipelines
Artifacts
Project settings
Working Azure FrontDoor CDN for Munki!
79ae144f Rod Christiansen committed Apr 22 main succeeded
Browse Files
Files Details
Parent 1 → This commit
Filter
main.tf -81 +193
/infrastructure/main.tf
Side-by-side
Munki
infrastructure
main.tf
secrets.tf
variables.tf
site_config["application_insights_key"]
resource "azurerm_application_insights" "enrollment" {
name = "munki-appinsights"
location = azurerm_resource_group.rg.location
resource_group_name = azurerm_resource_group.rg.name
application_type = "web"
site_config["application_insights_key"]
Azure Frontdoor CDN
resource "azurerm_cdn_frontdoor_profile" "munki" {
name = "munki-frontdoor"
resource_group_name = azurerm_resource_group.rg.name
sku_name = "Standard_AzureFrontDoor"
tags = {
environment = "production"
}
Azure Frontdoor Endpoint
resource "azurerm_cdn_frontdoor_endpoint" "munki" {
name = "munki-frontdoor-endpoint"
cdn_frontdoor_profile_id = azurerm_cdn_frontdoor_profile.munki.id
Frontdoor Origin Group for Function App
resource "azurerm_cdn_frontdoor_origin_group" "munki" {
name = "munki-frontdoor-origin-group"
cdn_frontdoor_profile_id = azurerm_cdn_frontdoor_profile.munki.id
health_probe {
path = "/health"
protocol = "Https"
interval_in_seconds = 30
request_type = "GET"
load_balancing {
sample_size = 4
successful_samples_required = 2
additional_latency_in_milliseconds = 0
Frontdoor Origin (Function App)
resource "azurerm_cdn_frontdoor_origin" "munki" {
name = "munki-origin"
cdn_frontdoor_origin_group_id = azurerm_cdn_frontdoor_origin_group.munki.
host_name = azurerm_linux_function_app.functions.def
priority = 1
weight = 1000
enabled = true


Pipelines - All
Azure DevOps emilycarru-its-infra / Devices / Pipelines
Search
Devices
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Library
Build Usage
Artifacts
Project settings
Pipelines
New pipeline
Recent
All
Runs
Filter pipelines
All pipelines
New folder
Name
Cimian - Infrastructure + Functions #20250418.2 • Resolves CDN! It's working!!
Cimian - Manifests Upkeep #20250606.1 • Updated exhibit to import Drive Mapping ADMX to intune
Cimian - Push + Assign Profiles #20250605.3 • Updated exhibit to import Drive Mapping ADMX to intune
Cimian - Push Remediation Scripts #20250502.6 • Now only prunes stales policies
Cimian - Push Repo to Production #20250606.1 • removed old name katanaToArnold
Commits - Infrastructure + Functions #20250506.2 • Actually let's do a different channel
Enrollment - Push to Production #20250605.1 • Merged PR 4655: Inventory updated. Changes queued in pull request.
Enrollment- Infrastructure + Functions #20250603.2 • Merge branch 'main' of https://emilycarru-its-infra.visualstudio.com/Devices/_git/Enrollment
Intune - Run Defender Full Scan #20250409.7 • Fixed defender scan endpoint address
Munki - Infrastructure + Functions #20250424.6 • Preventing a terraform run error
Munki - Manifests Upkeep #20250605.2 • Making sure there's never spaces on manifests
Munki - Publish Printers Installers #20250117.1 • Resolving issue with `jq` on < 15 devices
Munki - Push Profiles to Intune #20250606.3 • Mistakenly deleted Wacom Tablet Driver Enterprise
Munki - Push Repo to Production #20250607.2 • Slight tweak on removing old hooks, pushing pkg


Pipelines - Runs
Azure DevOps emilycarru-its-infra / Devices / Pipelines
Search
Devices
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Library
Build Usage
Artifacts
Project settings
Pipelines
New pipeline
Recent All Runs
All pipeline runs
Description
Stages
Slight tweak on removing old hooks, pushing pkg
#20250607.2 on Munki - Push Repo to Production Munki main 964beced
Yesterday
2m 6s
Migrated to plain text .applescript + `osacompile`
#20250607.1 on Munki - Push Repo to Production Munki main d063c382
Yesterday
2m 32s
removed old name katanaToArnold
#20250606.1 on Cimian - Push Repo to Production Cimian main 87e2d9b3
Friday
1m 17s
The ~~third~~ 25th time is a charm
#20250606.4 on Munki - Push Repo to Production Munki main 7cb89844
Friday
2m 45s
Mistakenly deleted Wacom Tablet Driver Enterprise
#20250606.3 on Munki - Push Profiles to Intune Munki main fb26e1a6
Friday
7m 35s
Mistakenly deleted Wacom Tablet Driver Enterprise
#20250606.3 on Munki - Push Repo to Production Munki main fb26e1a6
Friday
2m 56s
Updated exhibit to import Drive Mapping ADMX to intune
#20250606.1 on Cimian - Manifests Upkeep Cimian main ce598cf2
Friday
1m 16s
New Illus Studo minis manifests added to Production catalog
#20250606.2 on Munki - Push Profiles to Intune Munki main 9f96d9f1
Friday
7m 25s
New Illus Studo minis manifests added to Production catalog
#20250606.2 on Munki - Push Repo to Production Munki main 9f96d9f1
Friday
4m 13s
Re-adding Wacom Tablet Enterprise Driver 6.4.7-2 and added to ProvisioningCurriculum
#20250606.1 on Munki - Push Profiles to Intune Munki main 19cec56a
Friday
9m 13s
Re-adding Wacom Tablet Enterprise Driver 6.4.7-2 and added to ProvisioningCurriculum
#20250606.1 on Munki - Push Repo to Production Munki main 19cec56a
Friday
3m 21s
Automated client manifest creation in cloud repo for new or updated enrollments.
#20250605.5 on Munki - Push Profiles to Intune Munki main ee8350df
Thursday
8m 17s
Automated client manifest creation in cloud repo for new or updated enrollments.
#20250605.7 on Munki - Push Repo to Production Munki main ee8350df
Thursday
2m 44s
Updated exhibit to import Drive Mapping ADMX to intune
#20250605.3 on Cimian - Push + Assign Profiles Cimian main ce598cf2
Thursday
3m 44s
Updated exhibit to import Drive Mapping ADMX to intune
Thursday


Pipelines - Run 20241029.4 logs
Azure DevOps emilycarru-its-infra / Devices / Pipelines / Munki - Push Repo to Prod... / 20241029.4
Search
Devices
Overview
Repos
Pipelines
Pipelines
Environments
Library
Build Usage
Artifacts
Project settings
Jobs in run #20241029.4
Munki - Push Repo to Production
PushMunkiRepo
CloudDeploy 3m 21s
Initialize job 4s
Checkout Repository 26s
Enable sparse checkout 3s
Check and install Co... <1s
Download Munki tool... 14s
Install Munki tools 6s
Run makecatalogs 3s
Publish catalogs artifa... 5s
Download catalogs ar... 4s
Upload catalogs to S3 34s
Sync manifests to S3 54s
Sync pkgsinfo to S3 39s
Invalidate CloudFront ... 2s
Send failure notificati... <1s
Post-job: Checkout R... <1s
Finalize Job <1s
Report build status <1s
Upload catalogs to S3
View raw log
1 Starting: Upload catalogs to S3
2 ==
3 Task : Amazon S3 Upload
4 Description : Upload file and folder content to an Amazon Simple Storage Service (S3) Bucket on AWS
5 Version : 1.17.0
6 Author : Amazon Web Services
7 Help : Please refer to [Working with Amazon S3 Buckets](https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingBucket.html) for more information on working with Amazon
8
9 More information on this task can be found in the [task reference](https://docs.aws.amazon.com/vsts/latest/userguide/s3-upload.html).
10
11 ###Task Permissions
12 This task requires permissions to call the following AWS service APIs (depending on selected task options, not all APIs may be used):
13 * s3:CreateBucket
14 * s3:HeadBucket
15
16 Content uploads are performed using S3's PutObject API and/or the multi-part upload APIs. The specific APIs used depend on the size of the individual files being uploaded.
17 ==
18 Configuring credentials for task
19 ...configuring AWS credentials from service endpoint '99cf9a9d-13c7-4a6a-b4fa-5a8c4b1cbce5'
20 ...configuring AWS credentials from service endpoint '99cf9a9d-13c7-4a6a-b4fa-5a8c4b1cbce5'
21 Skipping OIDC: not enabled in service connections
22 ...configuring AWS credentials from service endpoint '99cf9a9d-13c7-4a6a-b4fa-5a8c4b1cbce5'
23 ...endpoint defines standard access/secret key credentials
24 Configuring region for task
25 ...configured to use region ca-central-1, defined in task.
26 Configuring region for task
27 ...configured to use region ca-central-1, defined in task.
28 Uploading files from /Users/runner/work/1/catalogs to repo/deployment/catalogs in bucket ecuad-macadmins-munki-cloud-s3-bucket
29 Searching /Users/runner/work/1/catalogs for files to upload
30 Uploading matched file /Users/runner/work/1/catalogs/Production, content type application/octet-stream
31 Completed upload of /Users/runner/work/1/catalogs/Production to repo/deployment/catalogs/Production
32 Uploading matched file /Users/runner/work/1/catalogs/Staging, content type application/octet-stream
33 Completed upload of /Users/runner/work/1/catalogs/Staging to repo/deployment/catalogs/Staging
34 Uploading matched file /Users/runner/work/1/catalogs/Testing, content type application/octet-stream
35 Completed upload of /Users/runner/work/1/catalogs/Testing to repo/deployment/catalogs/Testing
36 Uploading matched file /Users/runner/work/1/catalogs/Upcoming, content type application/octet-stream
37 Completed upload of /Users/runner/work/1/catalogs/Upcoming to repo/deployment/catalogs/Upcoming
38 Uploading matched file /Users/runner/work/1/catalogs/all, content type application/octet-stream
39 Completed upload of /Users/runner/work/1/catalogs/all to repo/deployment/catalogs/all
40 All uploads to S3 completed
41 Finishing: Upload catalogs to S3


Pipelines - Run 20241024.4 logs
Azure DevOps emilycarru-its-infra / Systems / Pipelines / Handbook - Push Latest B... / 20241024.4
Search
Systems
Overview
Repos
Pipelines
Pipelines
Environments
Library
Build Usage
Artifacts
Project settings
Jobs in run #20241024.4
Handbook - Push Latest Build
Rename Files for Consistency
Run File Renaming Script 11s
Initialize job <1s
Checkout Handbook... 3s
Rename Files in /webs... 4s
Prepare Updated Co... <1s
Publish Updated Web... 2s
Post-job: Checkout ... <1s
Finalize Job <1s
Rebuild and Deploy Handbook
Deploy Handbook to ... 2m 0s
Initialize job <1s
Download Artifact 4s
Checkout Handbook... 2s
Download Updated W... 4s
Rename Production ... <1s
Setup Environment 38s
Build and Deploy t... 1m 8s
Post-job: Checkout ... <1s
Finalize Job <1s
Deploy Handbook to Devel...
Finalize build
Report build status <1s
Build and Deploy to Production Environment
View raw log
Starting: Build and Deploy to Production Environment
Task : Command line
Description : Run a command line script using Bash on Linux and macOS and cmd.exe on Windows
Version : 2.246.1
Author : Microsoft Corporation
Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/command-line
Generating script.
========== Starting Command Output ==========
/usr/bin/bash --noprofile --norc /home/vsts/work/_temp/fd1f57ce-98c0-40a2-871b-b53014fe65d3.sh
Start building sites …
hugo v0.134.1-2f89169baa87a9db47e288b60705f4e99e21a945 linux/amd64 BuildDate=2024-09-05T10:17:50Z VendorInfo=gohugoio
| EN
Pages | 457
Paginator pages | 0
Non-page files | 119
Static files | 44
Processed images | 1
Aliases | 0
Cleaned | 0
Total in 25249 ms
Welcome to Azure Static Web Apps CLI (2.0.1)
Deploying front-end files from folder:
/home/vsts/work/1/s/website/public
Deploying to environment: production
Found configuration file:
/home/vsts/work/1/s/website/staticwebapp.config.json
Deploying project to Azure Static Web Apps...
Could not find StaticSitesClient local binary
- Downloading https://swalocaldeploy.azureedge.net/downloads/c25b033e400580829b0fea1fc7fb566139ab61c0/linux/StaticSitesClient@c25b033e400580829b0fea1fc7fb566139ab61c0
[swa] ✔ Downloading https://swalocaldeploy.azureedge.net/downloads/c25b033e400580829b0fea1fc7fb566139ab61c0/linux/StaticSitesClient@c25b033e400580829b0fea1fc7fb566139ab61c0
- Preparing deployment. Please wait...
✔ Project deployed to https://witty-hill-0cdc75210.5.azurestaticapps.net 🚀
Finishing: Build and Deploy to Production Environment

Devices

Overview

Boards

Repos

Pipelines

Artifacts

Project settings

munkitools 6.6.3-4704

Overview Versions Upstream Versions

Promote Delete Get Link

Get this package with the Azure DevOps Extension for Azure CLI

Connect to Feed then

```
az artifacts universal download \
--organization "https://dev.azure.com/emilycarru-its-infra/" \
--project "f6d33719-4062-45ea-af9e-4e4f881c1409" \
--scope project \
--feed "Munki" \
--name "munkitools" \
--version "6.6.3-4704" \
--path .
```

Description

Munki package version 6.6.3-4704

Provenance

Build

Munki - Update Artifact(s) / 20241105.4

Nov 5, 2024 (2024-11-05 12:34:21 PM)

Code

7783fadb in main

Munki

Development

Source

This feed

Stats

0 downloads of this version

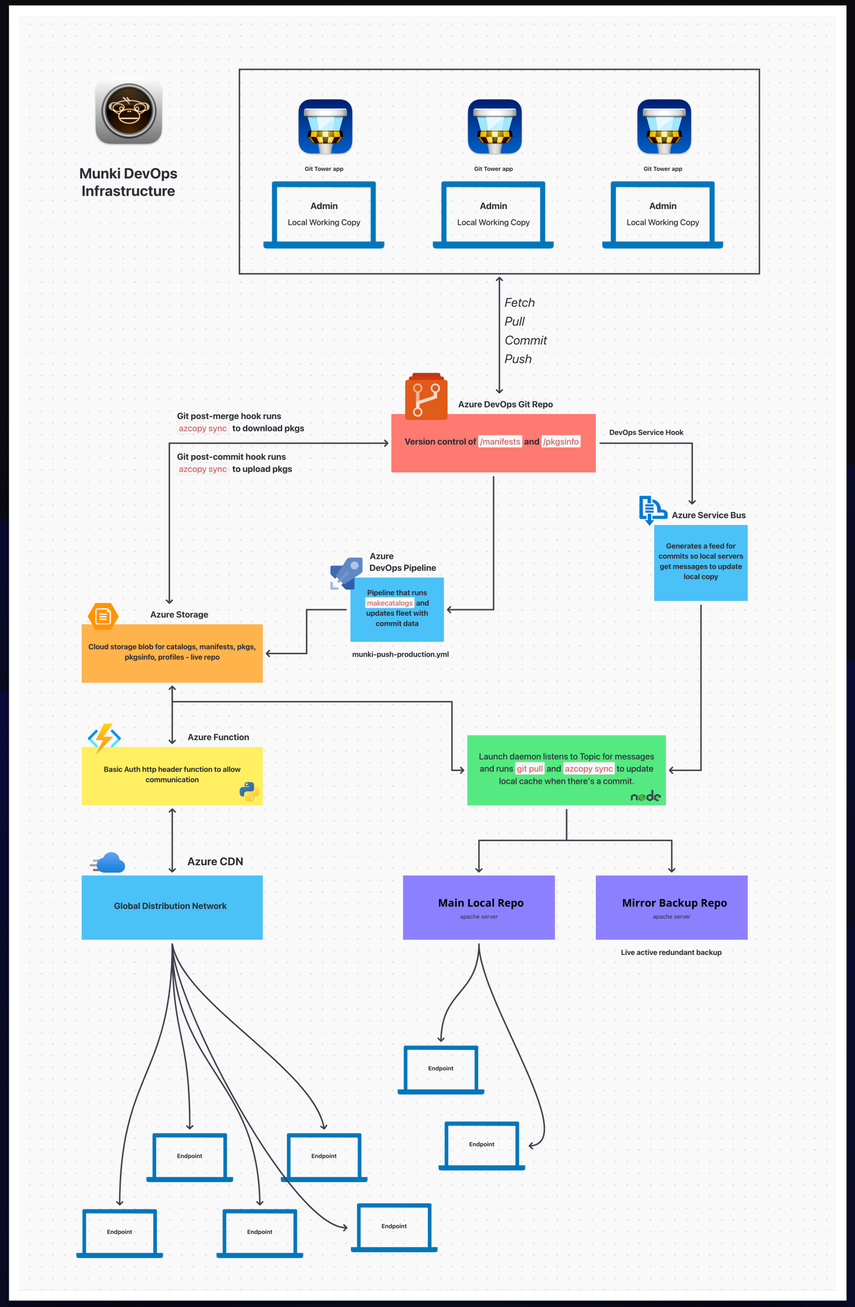
1K+ total downloads

0 users of this version

1 total user

Views

Local



Fully DevOps Munki

Admins local working copies

Git Repos

Git Hooks

CI/CD Pipeline

Cloud serverless infra

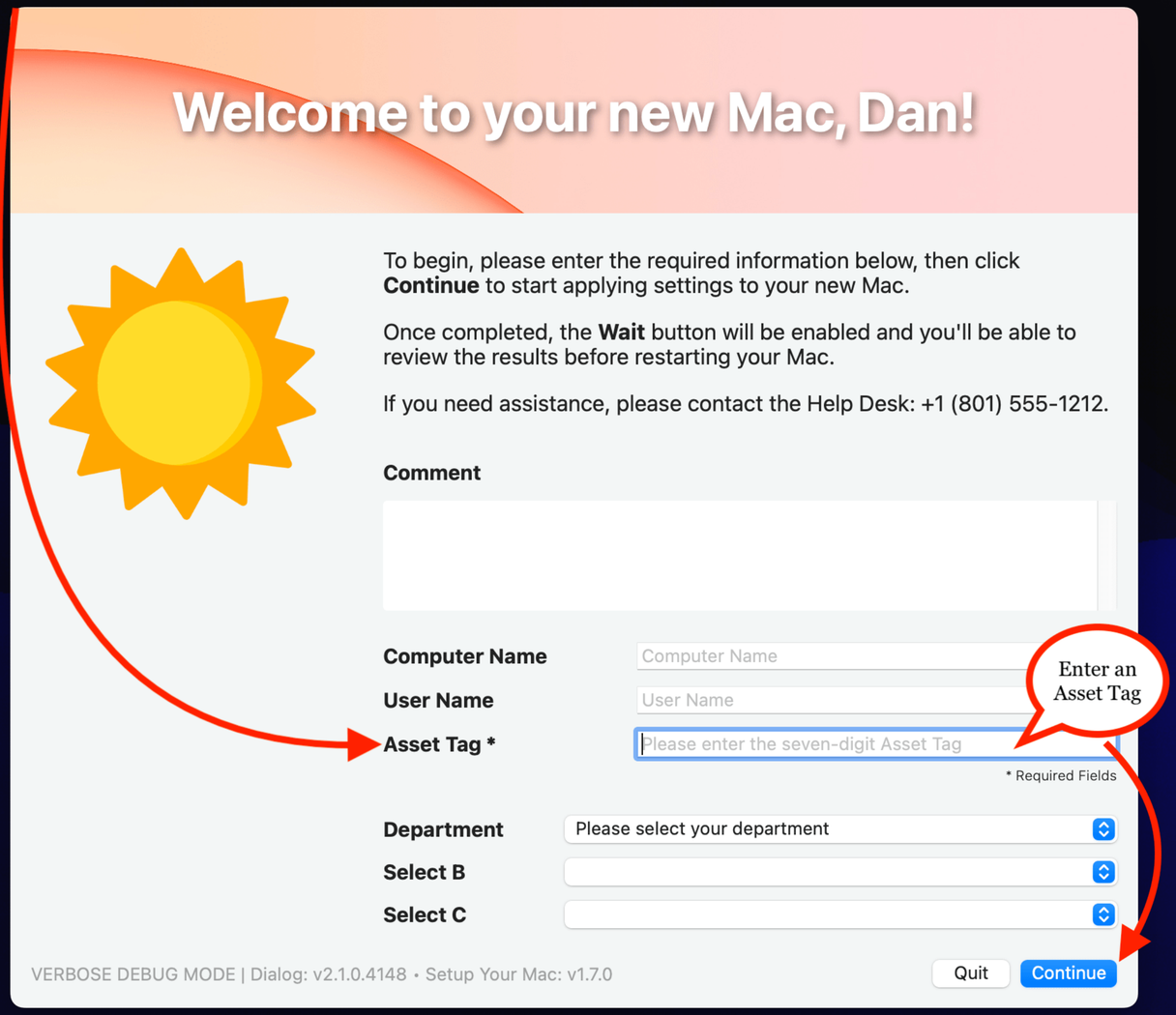
Updating local cache

Dynamic Repo URL

Before that though...



First part of the migration



Inventory

Asking end user for device info?!

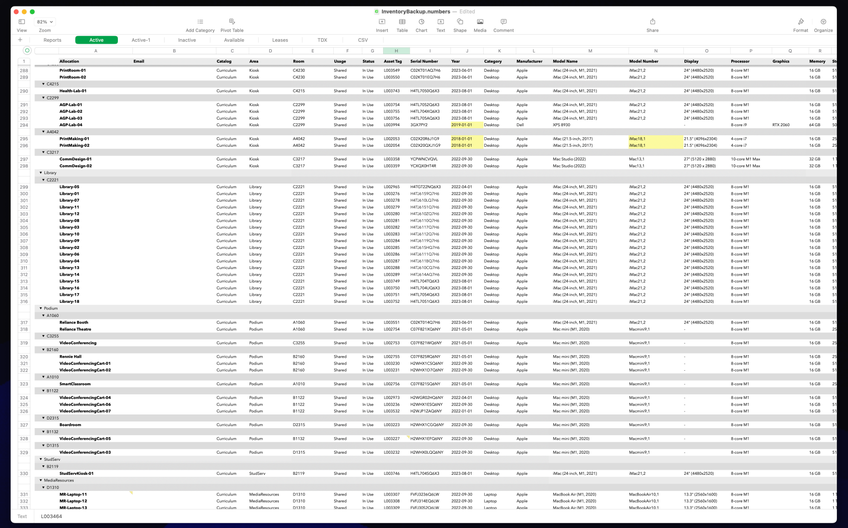
InventoryBackup.numbers — Edited

View | Zoom 82% | Add Category | Pivot Table | Insert | Table | Chart | Text | Shape | Media | Comment | Share | Format | Organize

Reports | Active | Active-1 | Inactive | Available | Leases | TDX | CSV

	Allocation	Email	Catalog	Area	Room	Usage	Status	Asset Tag	Serial Number	Year	Category	Manufacturer	Model Name	Model Number	Display	Processor	Graphics	Memory	St
288	**PrintRoom-01**		Curriculum	Kiosk	C4230	Shared	In Use	L003549	C02KT01AQ7H6	2023-06-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
289	**PrintRoom-02**		Curriculum	Kiosk	C4230	Shared	In Use	L003550	C02KT01EQ7H6	2023-06-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
	▼ C4215																		
290	**Health-Lab-01**		Curriculum	Kiosk	C4215	Shared	In Use	L003743	H4TL7050Q6X3	2023-08-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
	▼ C2299																		
291	**AGP-Lab-01**		Curriculum	Kiosk	C2299	Shared	In Use	L003754	H4TL7052Q6X3	2023-08-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
292	**AGP-Lab-02**		Curriculum	Kiosk	C2299	Shared	In Use	L003755	H4TL704XQ6X3	2023-08-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
293	**AGP-Lab-03**		Curriculum	Kiosk	C2299	Shared	In Use	L003756	H4TL705AQ6X3	2023-08-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
294	**AGP-Lab-04**		Curriculum	Kiosk	C2299	Shared	In Use	L003994	3GX7PY2	2019-01-01	Desktop	Dell	XPS 8930		-	8-core i9	RTX 2060	64 GB	50
	▼ A4042																		
295	**PrintMaking-01**		Curriculum	Kiosk	A4042	Shared	In Use	L002053	C02X20R6J1G9	2018-01-01	Desktop	Apple	iMac (21.5-inch, 2017)	iMac18,1	21.5" (4096x2304)	4-core i7		16 GB	25
296	**PrintMaking-02**		Curriculum	Kiosk	A4042	Shared	In Use	L002054	C02X20QXJ1G9	2018-01-01	Desktop	Apple	iMac (21.5-inch, 2017)	iMac18,1	21.5" (4096x2304)	4-core i7		16 GB	25
	▼ C3217																		
297	**CommDesign-01**		Curriculum	Kiosk	C3217	Shared	In Use	L003358	YCPWNCVQVL	2022-09-30	Desktop	Apple	Mac Studio (2022)	Mac13,1	27" (5120 x 2880)	10-core M1 Max		32 GB	1 T
298	**CommDesign-02**		Curriculum	Kiosk	C3217	Shared	In Use	L003359	YCXQX0HT4R	2022-09-30	Desktop	Apple	Mac Studio (2022)	Mac13,1	27" (5120 x 2880)	10-core M1 Max		32 GB	1 T
	▼ Library																		
	▼ C2221																		
299	**Library-05**		Curriculum	Library	C2221	Shared	In Use	L002965	H4TGT22NQ6X3	2022-04-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
300	**Library-01**		Curriculum	Library	C2221	Shared	In Use	L003276	H4TJ6159Q7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
301	**Library-07**		Curriculum	Library	C2221	Shared	In Use	L003278	H4TJ610LQ7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
302	**Library-11**		Curriculum	Library	C2221	Shared	In Use	L003279	H4TJ6151Q7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
303	**Library-12**		Curriculum	Library	C2221	Shared	In Use	L003280	H4TJ610ZQ7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
304	**Library-08**		Curriculum	Library	C2221	Shared	In Use	L003281	H4TJ6110Q7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
305	**Library-03**		Curriculum	Library	C2221	Shared	In Use	L003282	H4TJ6117Q7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
306	**Library-10**		Curriculum	Library	C2221	Shared	In Use	L003283	H4TJ6112Q7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
307	**Library-09**		Curriculum	Library	C2221	Shared	In Use	L003284	H4TJ6119Q7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
308	**Library-02**		Curriculum	Library	C2221	Shared	In Use	L003285	H4TJ615HQ7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
309	**Library-06**		Curriculum	Library	C2221	Shared	In Use	L003286	H4TJ6111Q7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
310	**Library-04**		Curriculum	Library	C2221	Shared	In Use	L003287	H4TJ611BQ7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
311	**Library-13**		Curriculum	Library	C2221	Shared	In Use	L003288	H4TJ610CQ7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
312	**Library-14**		Curriculum	Library	C2221	Shared	In Use	L003289	H4TJ614AQ7H6	2022-09-30	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
313	**Library-15**		Curriculum	Library	C2221	Shared	In Use	L003749	H4TL704TQ6X3	2023-08-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
314	**Library-16**		Curriculum	Library	C2221	Shared	In Use	L003750	H4TL704UQ6X3	2023-08-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
315	**Library-17**		Curriculum	Library	C2221	Shared	In Use	L003751	H4TL7054Q6X3	2023-08-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
316	**Library-18**		Curriculum	Library	C2221	Shared	In Use	L003752	H4TL7051Q6X3	2023-08-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
	▼ Podium																		
	▼ A1060																		
317	**Reliance Booth**		Curriculum	Podium	A1060	Shared	In Use	L003551	C02KT014Q7H6	2023-06-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
318	**Reliance Theatre**		Curriculum	Podium	A1060	Shared	In Use	L002754	C07F821XQ6NY	2021-05-01	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	25
	▼ C3255																		
319	**VideoConferencing**		Curriculum	Podium	C3255	Shared	In Use	L002753	C07F821WQ6NY	2021-05-01	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	25
	▼ B2160																		
320	**Rennie Hall**		Curriculum	Podium	B2160	Shared	In Use	L002755	C07F825RQ6NY	2021-05-01	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	25
321	**VideoConferencingCart-01**		Curriculum	Podium	B2160	Shared	In Use	L003230	H2WHX1CSQ6NY	2022-09-30	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	51
322	**VideoConferencingCart-02**		Curriculum	Podium	B2160	Shared	In Use	L003231	H2WHX1D7Q6NY	2022-09-30	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	51
	▼ A1010																		
323	**SmartClassroom**		Curriculum	Podium	A1010	Shared	In Use	L002756	C07F821SQ6NY	2021-05-01	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	25
	▼ B1122																		
324	**VideoConferencingCart-04**		Curriculum	Podium	B1122	Shared	In Use	L002973	H2WGR02HQ6NY	2022-04-01	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	51
325	**VideoConferencingCart-06**		Curriculum	Podium	B1122	Shared	In Use	L003236	H2WHX1ESQ6NY	2022-09-30	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	51
326	**VideoConferencingCart-07**		Curriculum	Podium	B1122	Shared	In Use	L003532	H2WJP1ZAQ6NY	2022-01-01	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	51
	▼ D2315																		
327	**Boardroom**		Curriculum	Podium	D2315	Shared	In Use	L003223	H2WHX1CGQ6NY	2022-09-30	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	51
	▼ B1132																		
328	**VideoConferencingCart-05**		Curriculum	Podium	B1132	Shared	In Use	L003227	H2WHX1EFQ6NY	2022-09-30	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	51
	▼ D1315																		
329	**VideoConferencingCart-03**		Curriculum	Podium	D1315	Shared	In Use	L003232	H2WHX0LQQ6NY	2022-09-30	Desktop	Apple	Mac mini (M1, 2020)	Macmini9,1	-	8-core M1		16 GB	51
	▼ StudServ																		
	▼ B2119																		
330	**StudServKiosk-01**		Curriculum	StudServ	B2119	Shared	In Use	L003746	H4TL704SQ6X3	2023-08-01	Desktop	Apple	iMac (24-inch, M1, 2021)	iMac21,2	24" (4480x2520)	8-core M1		16 GB	51
	▼ MediaResources																		
	▼ D1310																		
331	**MR-Laptop-11**		Curriculum	MediaResources	D1310	Shared	In Use	L003307	FVFJ3236Q6LW	2022-09-30	Laptop	Apple	MacBook Air (M1, 2020)	MacBookAir10,1	13.3" (2560x1600)	8-core M1		16 GB	1 T
332	**MR-Laptop-12**		Curriculum	MediaResources	D1310	Shared	In Use	L003308	FVFJ314EQ6LW	2022-09-30	Laptop	Apple	MacBook Air (M1, 2020)	MacBookAir10,1	13.3" (2560x1600)	8-core M1		16 GB	1 T
333	**MR-Laptop-13**		Curriculum	MediaResources	D1310	Shared	In Use	L003309	FVFJ305Q2Q6LW	2022-09-30	Laptop	Apple	MacBook Air (M1, 2020)	MacBookAir10,1	13.3" (2560x1600)	8-core M1		16 GB	1 T

Text | L003464



Inventory

Single Local Mac

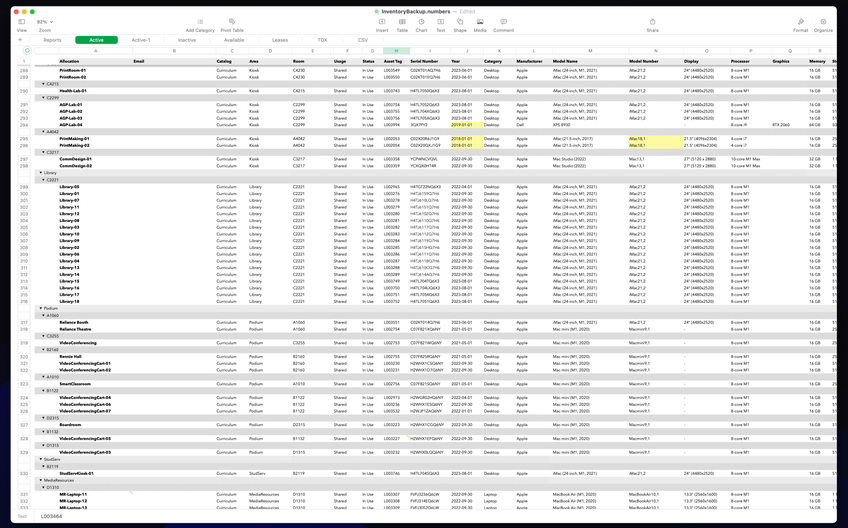
Exporting with local osascript

VNC'ing into that one Mac

Numbers and iCloud sharing 🤗

http://munki.domain.com/enroll/computers.csv

Munki preflight



Inventory

Determines Munki's manifests

Asset tag and Serial

Catalog (Staff, Faculty, Curriculum, Kiosk)

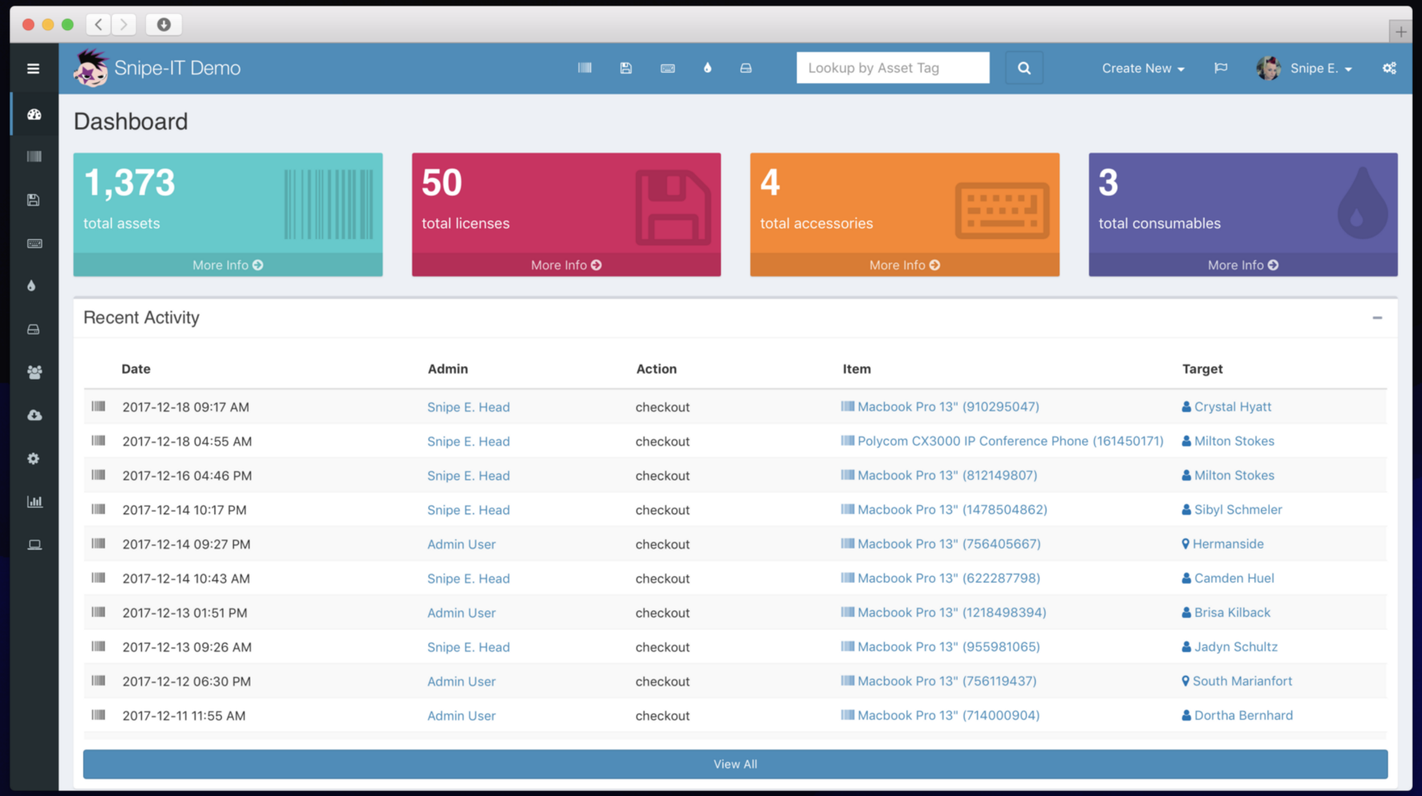
Area

Location

Username

http://munki.domain.com/enroll/computers.csv

Munki preflight



Inventory

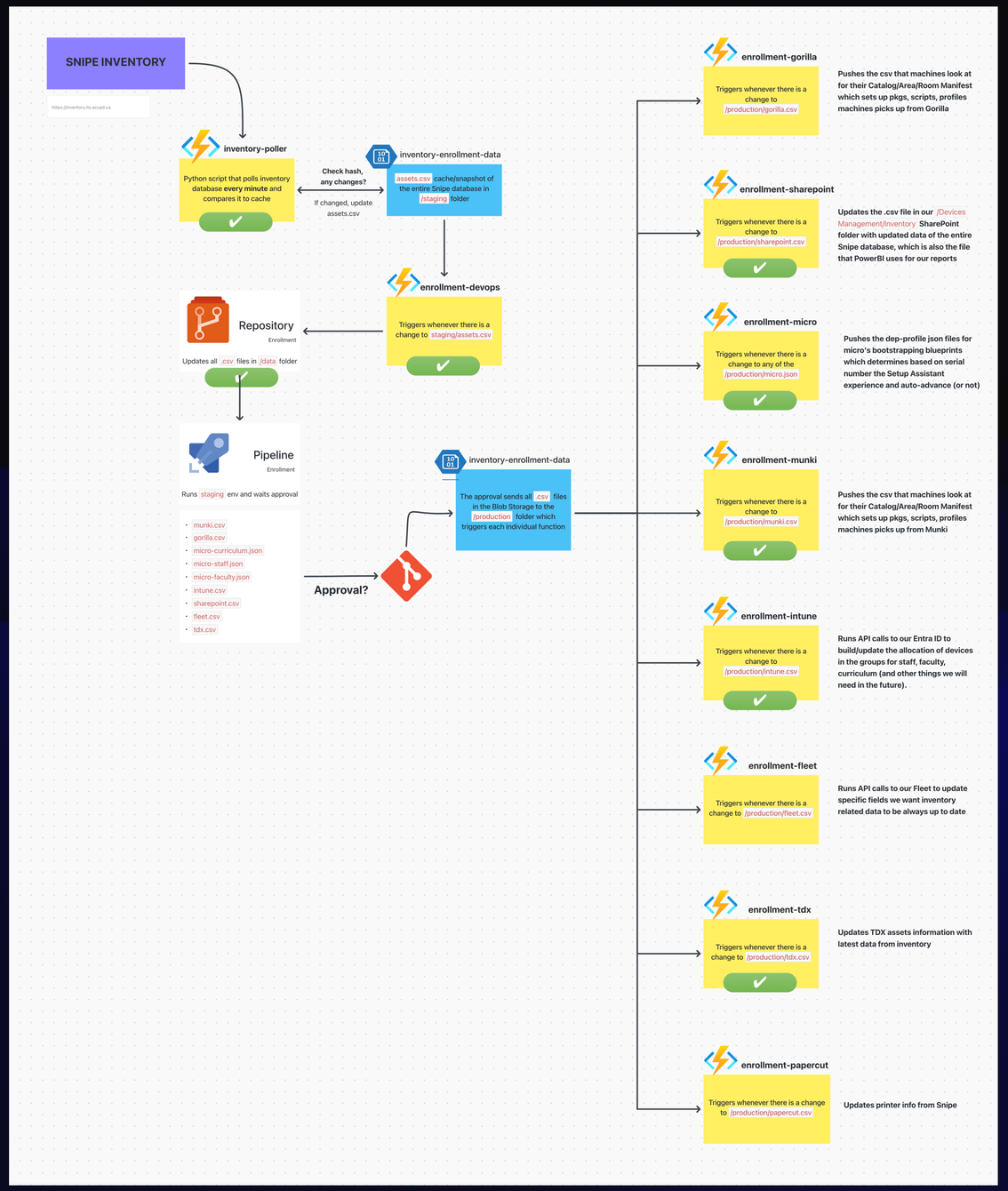
Terraformed in Azure

PaaS / Serverless

REST API

It's awesome!

https://registry.terraform.io/modules/rodchristiansen/snipeit/azurerm



Munki Manifests

Inventory poller every minute

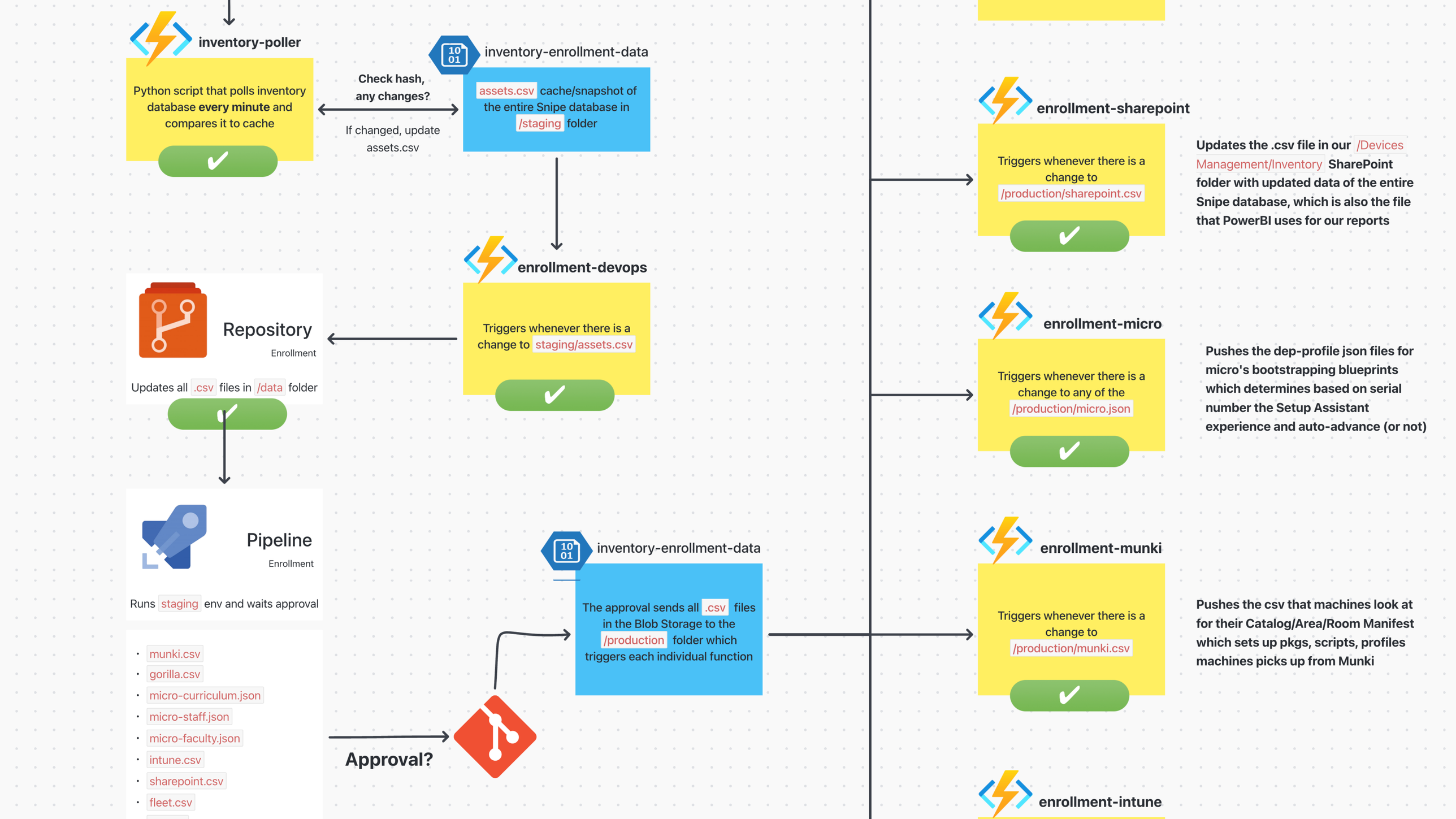
Caches export into a `staging/assets.csv`

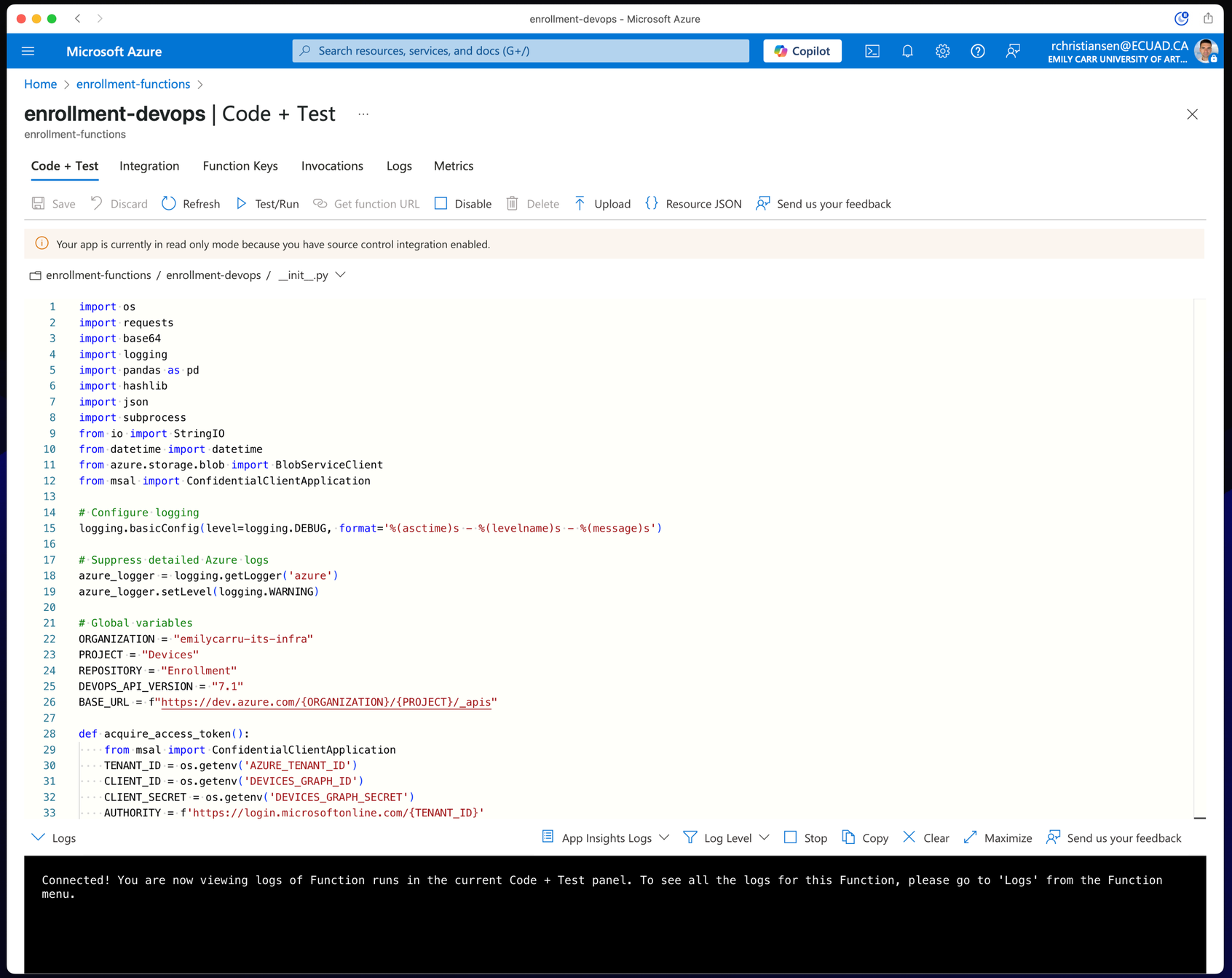
Compares hashes, if changed create a PR

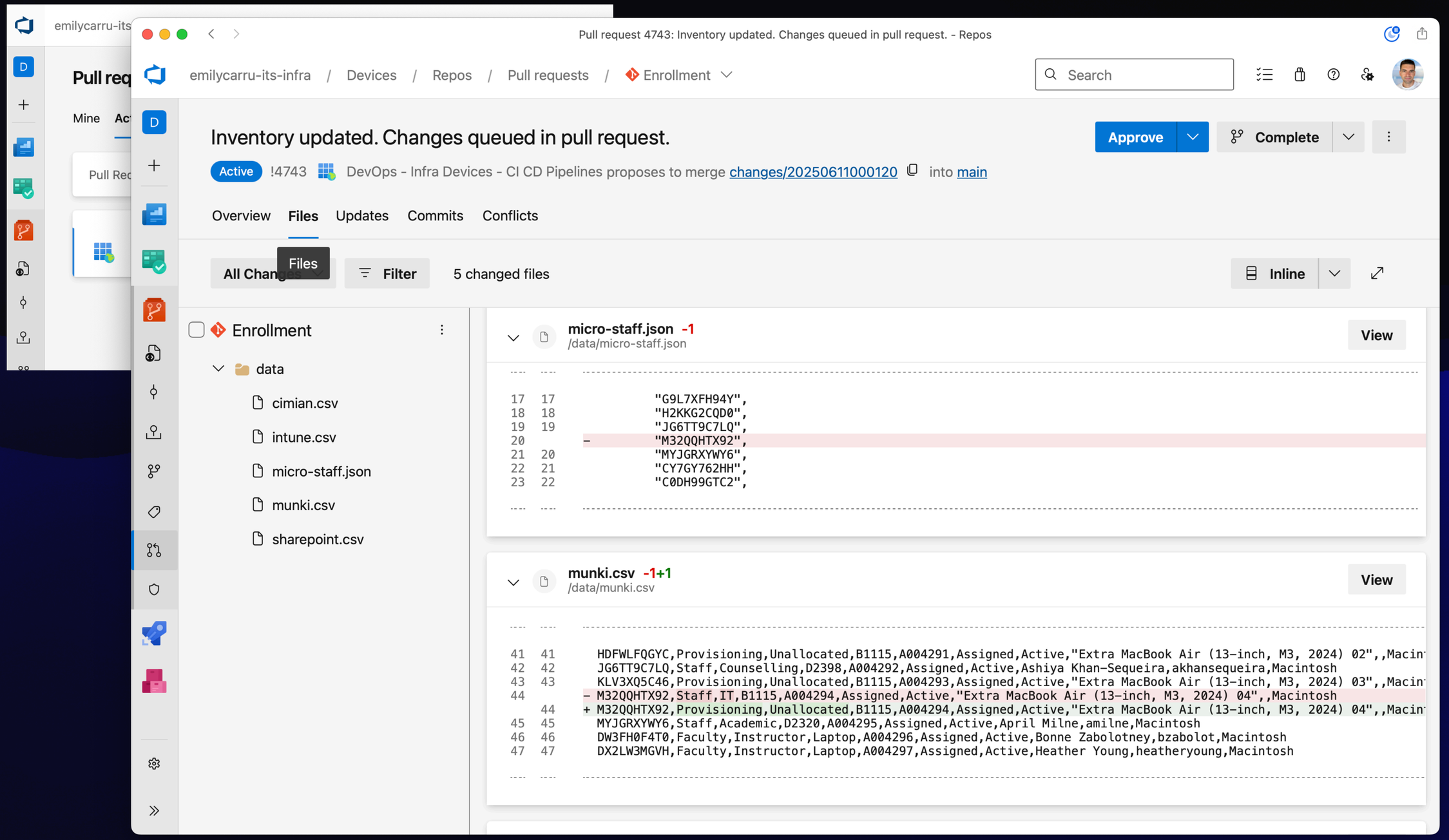
We review PR and approve/abandon

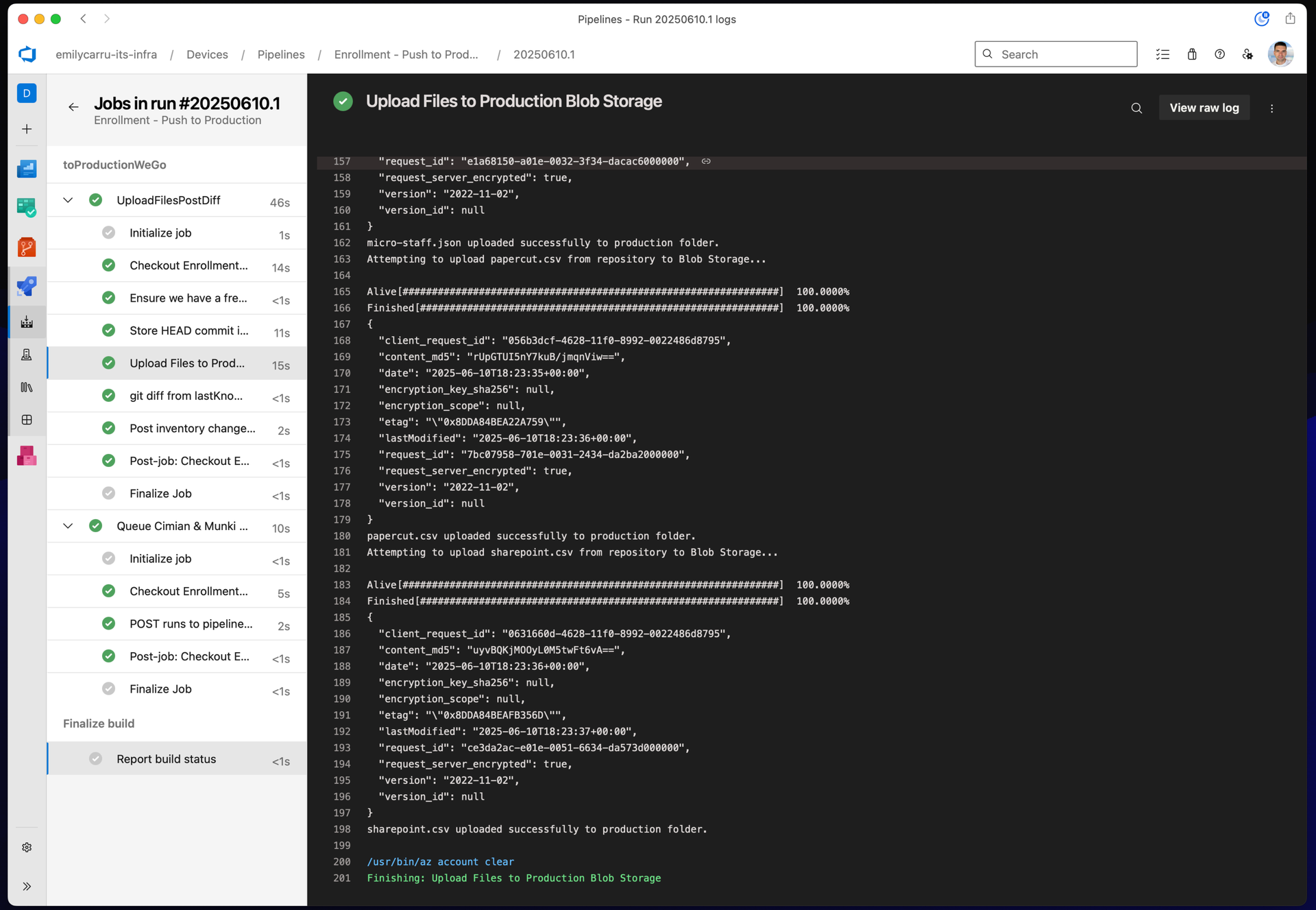
If approved, pipeline runs and creates csv files for each individual system that needs inventory data

Each system has a csv


inventory-poller
Python script that polls inventory database every minute and compares it to cache
Check hash, any changes?
If changed, update assets.csv
inventory-enrollment-data
assets.csv cache/snapshot of the entire Snipe database in /staging folder
enrollment-devops
Triggers whenever there is a change to staging/assets.csv
Repository
Enrollment
Updates all .csv files in /data folder
Pipeline
Enrollment
Runs staging env and waits approval
munki.csv
gorilla.csv
micro-curriculum.json
micro-staff.json
micro-faculty.json
intune.csv
sharepoint.csv
fleet.csv
Approval?
inventory-enrollment-data
The approval sends all .csv files in the Blob Storage to the /production folder which triggers each individual function
enrollment-sharepoint
Triggers whenever there is a change to /production/sharepoint.csv
Updates the .csv file in our /Devices Management/Inventory SharePoint folder with updated data of the entire Snipe database, which is also the file that PowerBI uses for our reports
enrollment-micro
Triggers whenever there is a change to any of the /production/micro.json
Pushes the dep-profile json files for micro's bootstrapping blueprints which determines based on serial number the Setup Assistant experience and auto-advance (or not)
enrollment-munki
Triggers whenever there is a change to /production/munki.csv
Pushes the csv that machines look at for their Catalog/Area/Room Manifest which sets up pkgs, scripts, profiles machines picks up from Munki
enrollment-intune


enrollment-devops - Microsoft Azure
Microsoft Azure
Search resources, services, and docs (G+/)
Copilot
rchristiansen@ECUAD.CA
EMILY CARR UNIVERSITY OF ART...
Home > enrollment-functions >
enrollment-devops | Code + Test
enrollment-functions
Code + Test
Integration
Function Keys
Invocations
Logs
Metrics
Save
Discard
Refresh
Test/Run
Get function URL
Disable
Delete
Upload
Resource JSON
Send us your feedback
Your app is currently in read only mode because you have source control integration enabled.
enrollment-functions / enrollment-devops / __init__.py
1 import os
2 import requests
3 import base64
4 import logging
5 import pandas as pd
6 import hashlib
7 import json
8 import subprocess
9 from io import StringIO
10 from datetime import datetime
11 from azure.storage.blob import BlobServiceClient
12 from msal import ConfidentialClientApplication
13
14 # Configure logging
15 logging.basicConfig(level=logging.DEBUG, format='%(asctime)s - %(levelname)s - %(message)s')
16
17 # Suppress detailed Azure logs
18 azure_logger = logging.getLogger('azure')
19 azure_logger.setLevel(logging.WARNING)
20
21 # Global variables
22 ORGANIZATION = "emilycarru-its-infra"
23 PROJECT = "Devices"
24 REPOSITORY = "Enrollment"
25 DEVOPS_API_VERSION = "7.1"
26 BASE_URL = f"https://dev.azure.com/{ORGANIZATION}/{PROJECT}/_apis"
27
28 def acquire_access_token():
29 from msal import ConfidentialClientApplication
30 TENANT_ID = os.getenv('AZURE_TENANT_ID')
31 CLIENT_ID = os.getenv('DEVICES_GRAPH_ID')
32 CLIENT_SECRET = os.getenv('DEVICES_GRAPH_SECRET')
33 AUTHORITY = f'https://login.microsoftonline.com/{TENANT_ID}'
Logs
App Insights Logs
Log Level
Stop
Copy
Clear
Maximize
Send us your feedback
Connected! You are now viewing logs of Function runs in the current Code + Test panel. To see all the logs for this Function, please go to 'Logs' from the Function menu.


Pull request 4743: Inventory updated. Changes queued in pull request. - Repos
emilycarru-its-infra / Devices / Repos / Pull requests / Enrollment
Search
Inventory updated. Changes queued in pull request.
Approve
Complete
Active !4743 DevOps - Infra Devices - CI CD Pipelines proposes to merge changes/20250611000120 into main
Overview Files Updates Commits Conflicts
Files
Filter
5 changed files
Inline
Enrollment
data
cimian.csv
intune.csv
micro-staff.json
munki.csv
sharepoint.csv
micro-staff.json -1
/data/micro-staff.json
View
"G9L7XFH94Y",
"H2KKG2CQD0",
"JG6TT9C7LQ",
"M32QQHTX92",
"MYJGRXYWY6",
"CY7GY762HH",
"C0DH99GTC2",
munki.csv -1+1
/data/munki.csv
View
HDFWLFQGYC,Provisioning,Unallocated,B1115,A004291,Assigned,Active,"Extra MacBook Air (13-inch, M3, 2024) 02",,Macin
JG6TT9C7LQ,Staff,Counselling,D2398,A004292,Assigned,Active,Ashiya Khan-Sequeira,akhansequeira,Macintosh
KLV3XQ5C46,Provisioning,Unallocated,B1115,A004293,Assigned,Active,"Extra MacBook Air (13-inch, M3, 2024) 03",,Macin
- M32QQHTX92,Staff,IT,B1115,A004294,Assigned,Active,"Extra MacBook Air (13-inch, M3, 2024) 04",,Macintosh
+ M32QQHTX92,Provisioning,Unallocated,B1115,A004294,Assigned,Active,"Extra MacBook Air (13-inch, M3, 2024) 04",,Macin
MYJGRXYWY6,Staff,Academic,D2320,A004295,Assigned,Active,April Milne,amilne,Macintosh
DW3FH0F4T0,Faculty,Instructor,Laptop,A004296,Assigned,Active,Bonne Zabolotney,bzabolot,Macintosh
DX2LW3MGVH,Faculty,Instructor,Laptop,A004297,Assigned,Active,Heather Young,heatheryoung,Macintosh


Pipelines - Run 20250610.1 logs
emilycarru-its-infra / Devices / Pipelines / Enrollment - Push to Prod... / 20250610.1
Search
Jobs in run #20250610.1
Enrollment - Push to Production
toProductionWeGo
UploadFilesPostDiff 46s
Initialize job 1s
Checkout Enrollment... 14s
Ensure we have a fre... <1s
Store HEAD commit i... 11s
Upload Files to Prod... 15s
git diff from lastKno... <1s
Post inventory change... 2s
Post-job: Checkout E... <1s
Finalize Job <1s
Queue Cimian & Munki ... 10s
Initialize job <1s
Checkout Enrollment... 5s
POST runs to pipeline... 2s
Post-job: Checkout E... <1s
Finalize Job <1s
Finalize build
Report build status <1s
Upload Files to Production Blob Storage
View raw log
"request_id": "e1a68150-a01e-0032-3f34-dacac6000000",
"request_server_encrypted": true,
"version": "2022-11-02",
"version_id": null
}
micro-staff.json uploaded successfully to production folder.
Attempting to upload papercut.csv from repository to Blob Storage...
Alive[##] 100.0000%
Finished[##] 100.0000%
{
"client_request_id": "056b3dcf-4628-11f0-8992-0022486d8795",
"content_md5": "rUpGTUI5nY7kuB/jmqnViw==",
"date": "2025-06-10T18:23:35+00:00",
"encryption_key_sha256": null,
"encryption_scope": null,
"etag": "\"0x8DDA84BEA22A759\"",
"lastModified": "2025-06-10T18:23:36+00:00",
"request_id": "7bc07958-701e-0031-2434-da2ba2000000",
"request_server_encrypted": true,
"version": "2022-11-02",
"version_id": null
}
papercut.csv uploaded successfully to production folder.
Attempting to upload sharepoint.csv from repository to Blob Storage...
Alive[##] 100.0000%
Finished[##] 100.0000%
{
"client_request_id": "0631660d-4628-11f0-8992-0022486d8795",
"content_md5": "uyvBQKjMOOyL0M5twFt6vA==",
"date": "2025-06-10T18:23:36+00:00",
"encryption_key_sha256": null,
"encryption_scope": null,
"etag": "\"0x8DDA84BEAFB356D\"",
"lastModified": "2025-06-10T18:23:37+00:00",
"request_id": "ce3da2ac-e01e-0051-6634-da573d000000",
"request_server_encrypted": true,
"version": "2022-11-02",
"version_id": null
}
sharepoint.csv uploaded successfully to production folder.
/usr/bin/az account clear
Finishing: Upload Files to Production Blob Storage

Info | Licenses | Components | Assets | History | Maintenances | Files | Additional Files | Upload

Asset Tag	A004656
Status	Active Deployed → Rod Christiansen
Asset Name	Rod Christiansen
Serial	DC2TNJ4PPQ
Checkout Date	2025-01-21 09:50 AM
Manufacturer	Apple
Category	Laptop
Model	MacBook Pro (16-inch, 2024)
Model No.	Mac16,5
BYOD	✕ No
Requestable	✕ No
Fleet	
Platform	Macintosh
Usage	Assigned
Catalog	Staff
Area	IT
Colour	Silver
Display	16.2-inch (3456x2234)
Memory	24 GB
Storage	2 TB
Chip	M4 Pro
CPU	



Edit Asset

Checkin Asset

Add Note

Audit

Clone Asset

Generate Label

Checkin and Delete

Checked Out To →

Rod Christiansen

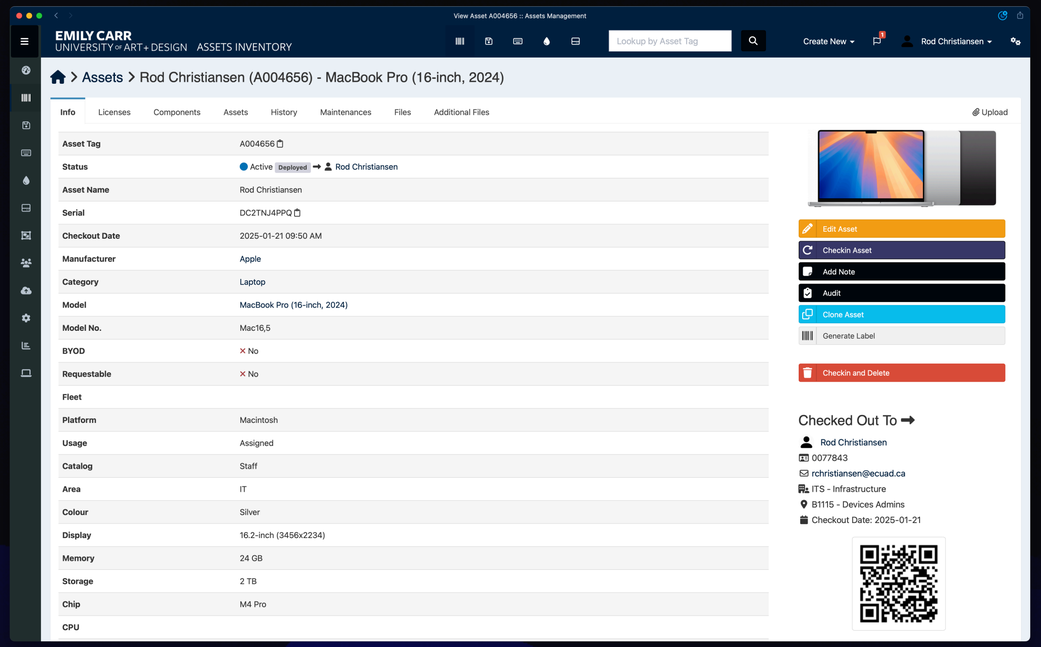
0077843

rchristiansen@ecuad.ca

ITS - Infrastructure

B1115 - Devices Admins

Checkout Date: 2025-01-21



Inventory

Determines what apps, prefs, printers, etc.

Fully git'ed

Forces accurate inventory

http://munki.domain.com/enroll/computers.csv

Munki preflight → Munki Manifest:

`Assigned/Staff/IT/B1115/RodChristiansen`

Pipeline

Enrollment

Runs `staging` env and waits approval

- `munki.csv`
- `gorilla.csv`
- `micro-curriculum.json`
- `micro-staff.json`
- `micro-faculty.json`
- `intune.csv`
- `sharepoint.csv`
- `fleet.csv`
- `tdx.csv`

Approval?

inventory-enrollment-data

The approval sends all `.csv` files in the Blob Storage to the `/production` folder which triggers each individual function

enrollment-munki

Triggers whenever there is a change to `/production/munki.csv`

✔

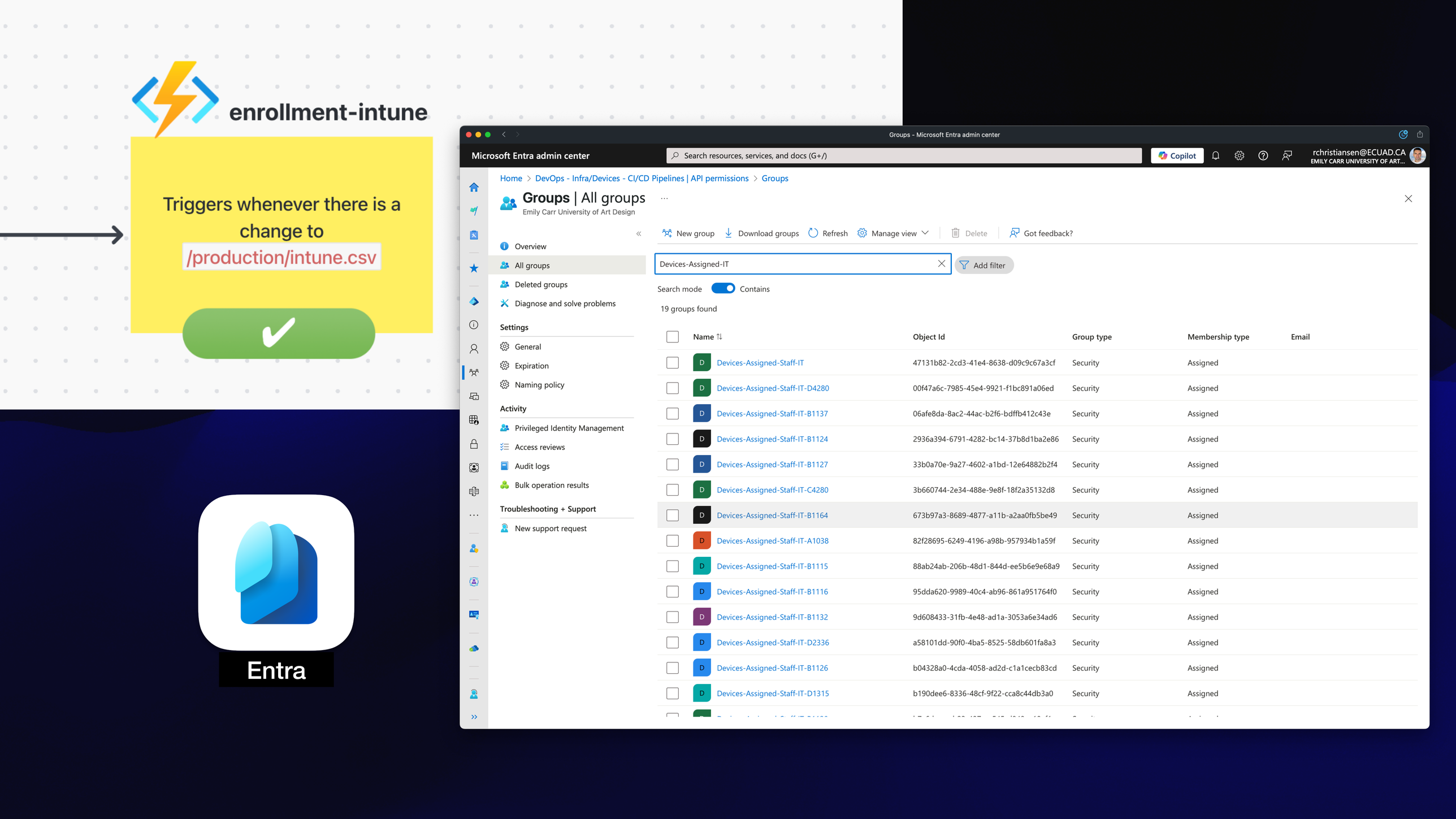
Pushes the csv that machines look at for their Catalog/Area/Room Manifest which sets up pkgs, scripts, profiles machines picks up from Munki

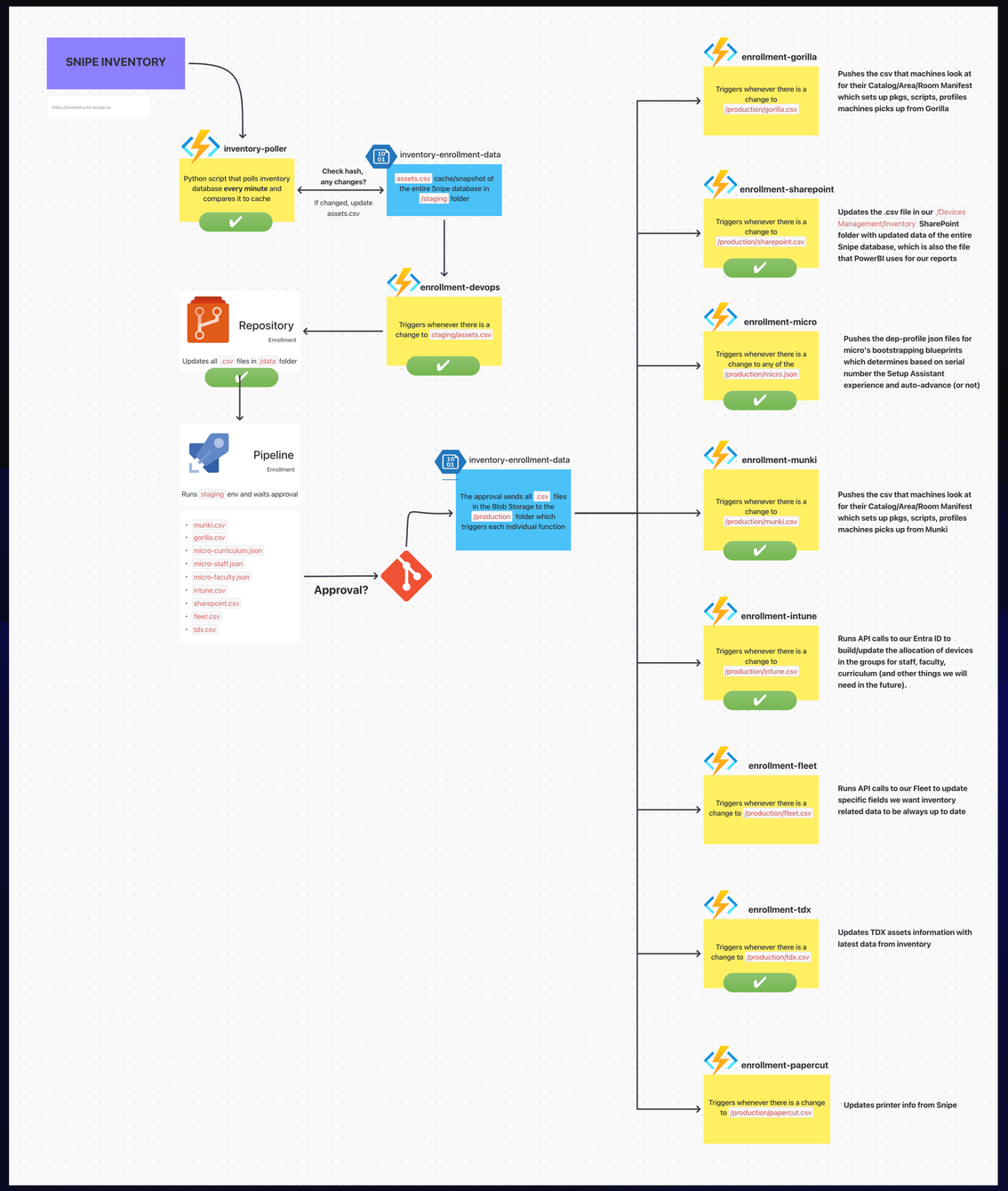
enrollment-intune

Triggers whenever there is a change to `/production/intune.csv`

✔

Runs API calls to our Entra ID to build/update the allocation of devices in the groups for staff, faculty, curriculum (and other things we will need in the future).


enrollment-intune
Triggers whenever there is a change to
/production/intune.csv
Entra
Groups - Microsoft Entra admin center
Microsoft Entra admin center
Search resources, services, and docs (G+/)
Copilot
rchristiansen@ECUAD.CA
EMILY CARR UNIVERSITY OF ART...
Home > DevOps - Infra/Devices - CI/CD Pipelines | API permissions > Groups
Groups | All groups
Emily Carr University of Art Design
Overview
All groups
Deleted groups
Diagnose and solve problems
Settings
General
Expiration
Naming policy
Activity
Privileged Identity Management
Access reviews
Audit logs
Bulk operation results
Troubleshooting + Support
New support request
New group
Download groups
Refresh
Manage view
Delete
Got feedback?
Devices-Assigned-IT
Add filter
Search mode
Contains
19 groups found
Name
Object Id
Group type
Membership type
Email
Devices-Assigned-Staff-IT 47131b82-2cd3-41e4-8638-d09c9c67a3cf Security Assigned
Devices-Assigned-Staff-IT-D4280 00f47a6c-7985-45e4-9921-f1bc891a06ed Security Assigned
Devices-Assigned-Staff-IT-B1137 06afe8da-8ac2-44ac-b2f6-bdffb412c43e Security Assigned
Devices-Assigned-Staff-IT-B1124 2936a394-6791-4282-bc14-37b8d1ba2e86 Security Assigned
Devices-Assigned-Staff-IT-B1127 33b0a70e-9a27-4602-a1bd-12e64882b2f4 Security Assigned
Devices-Assigned-Staff-IT-C4280 3b660744-2e34-488e-9e8f-18f2a35132d8 Security Assigned
Devices-Assigned-Staff-IT-B1164 673b97a3-8689-4877-a11b-a2aa0fb5be49 Security Assigned
Devices-Assigned-Staff-IT-A1038 82f28695-6249-4196-a98b-957934b1a59f Security Assigned
Devices-Assigned-Staff-IT-B1115 88ab24ab-206b-48d1-844d-ee5b6e9e68a9 Security Assigned
Devices-Assigned-Staff-IT-B1116 95dda620-9989-40c4-ab96-861a951764f0 Security Assigned
Devices-Assigned-Staff-IT-B1132 9d608433-31fb-4e48-ad1a-3053a6e34ad6 Security Assigned
Devices-Assigned-Staff-IT-D2336 a58101dd-90f0-4ba5-8525-58db601fa8a3 Security Assigned
Devices-Assigned-Staff-IT-B1126 b04328a0-4cda-4058-ad2d-c1a1cecb83cd Security Assigned
Devices-Assigned-Staff-IT-D1315 b190dee6-8336-48cf-9f22-cca8c44db3a0 Security Assigned



Munki Manifests

Inventory poller every minute

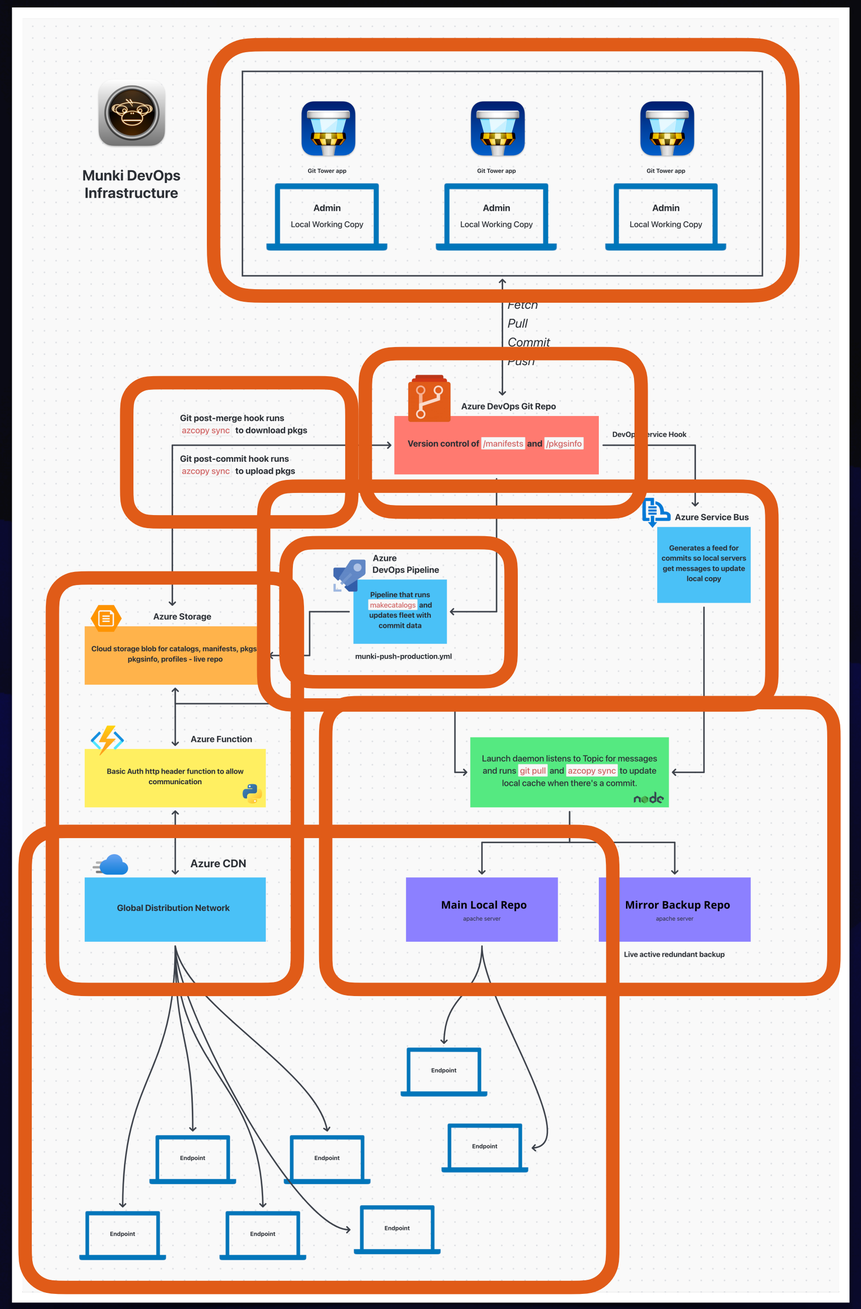
Caches export into a `staging/assets.csv`

Compares hashes, if changed create a PR

We review PR and approve/abandon

If approved, pipeline runs and creates csv files for each individual system that needs inventory data

Each system has a csv



Fully DevOps Munki

Each admin has the entire repo locally

Git repo source controls manifests and pkgsinfo

Git hooks - the key for syncing pkgs (and icons).

Git Pull run downloads. Git Push runs uploads.

Catalogs built by CI/CD pipeline on commits to main

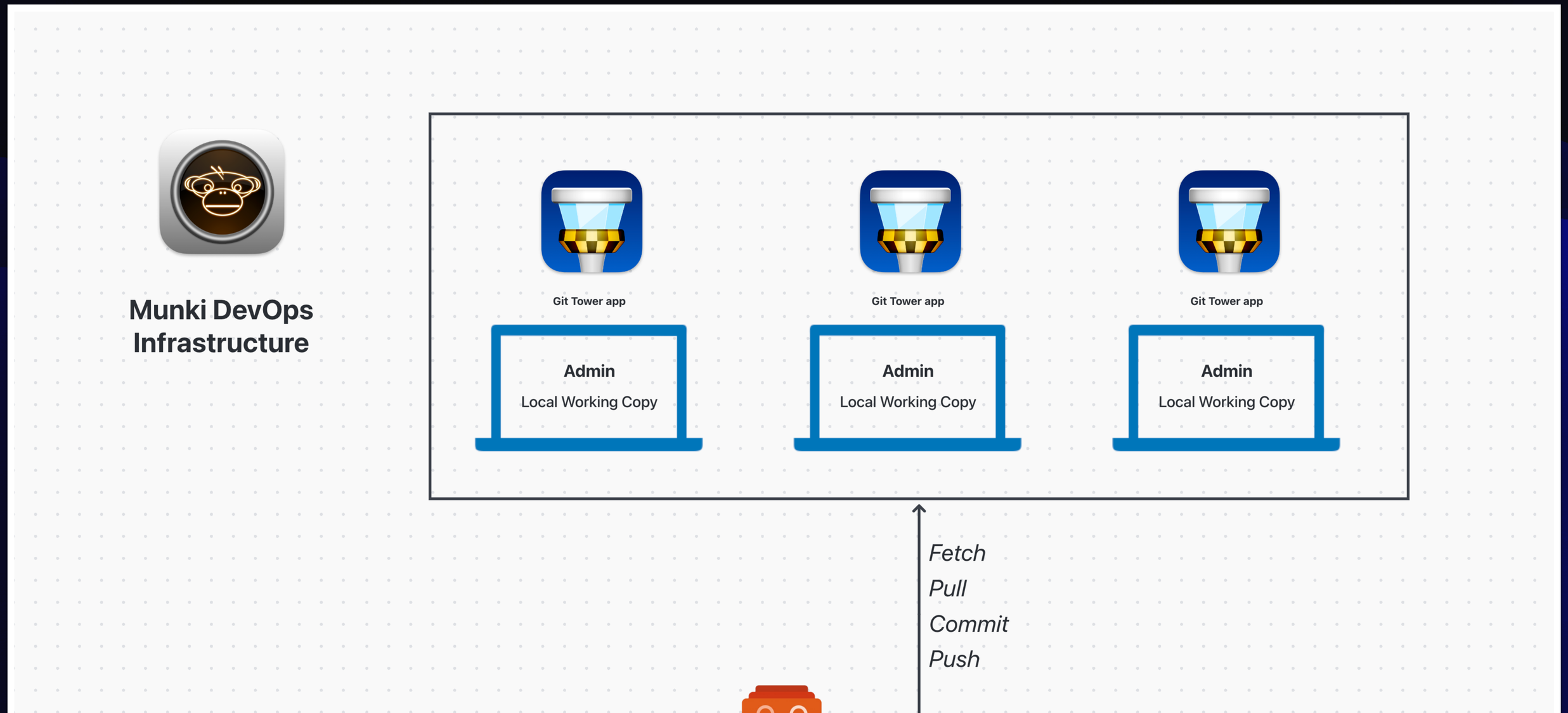
Pipeline pushes repo data to cloud bucket and purges/ invalidates CDN cache

A pipeline successful run messages Service Bus

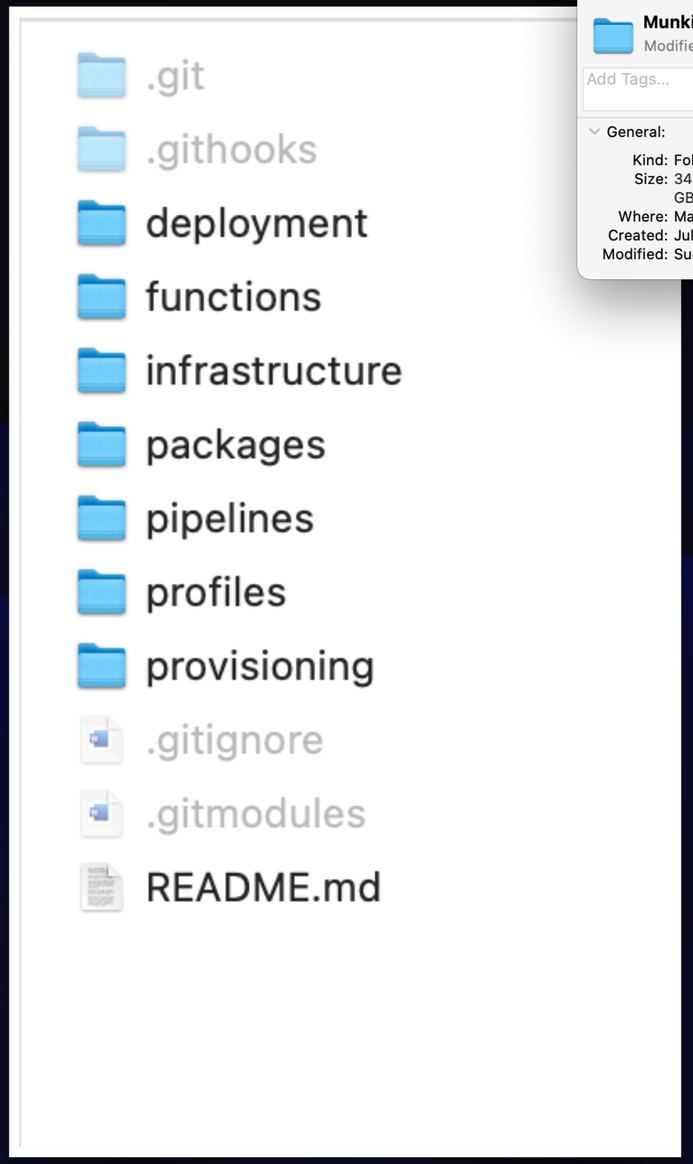
Local caching servers git pull and download pkgs

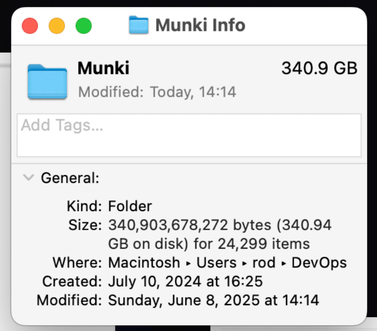
Seamless switching between on/off prem with preflight

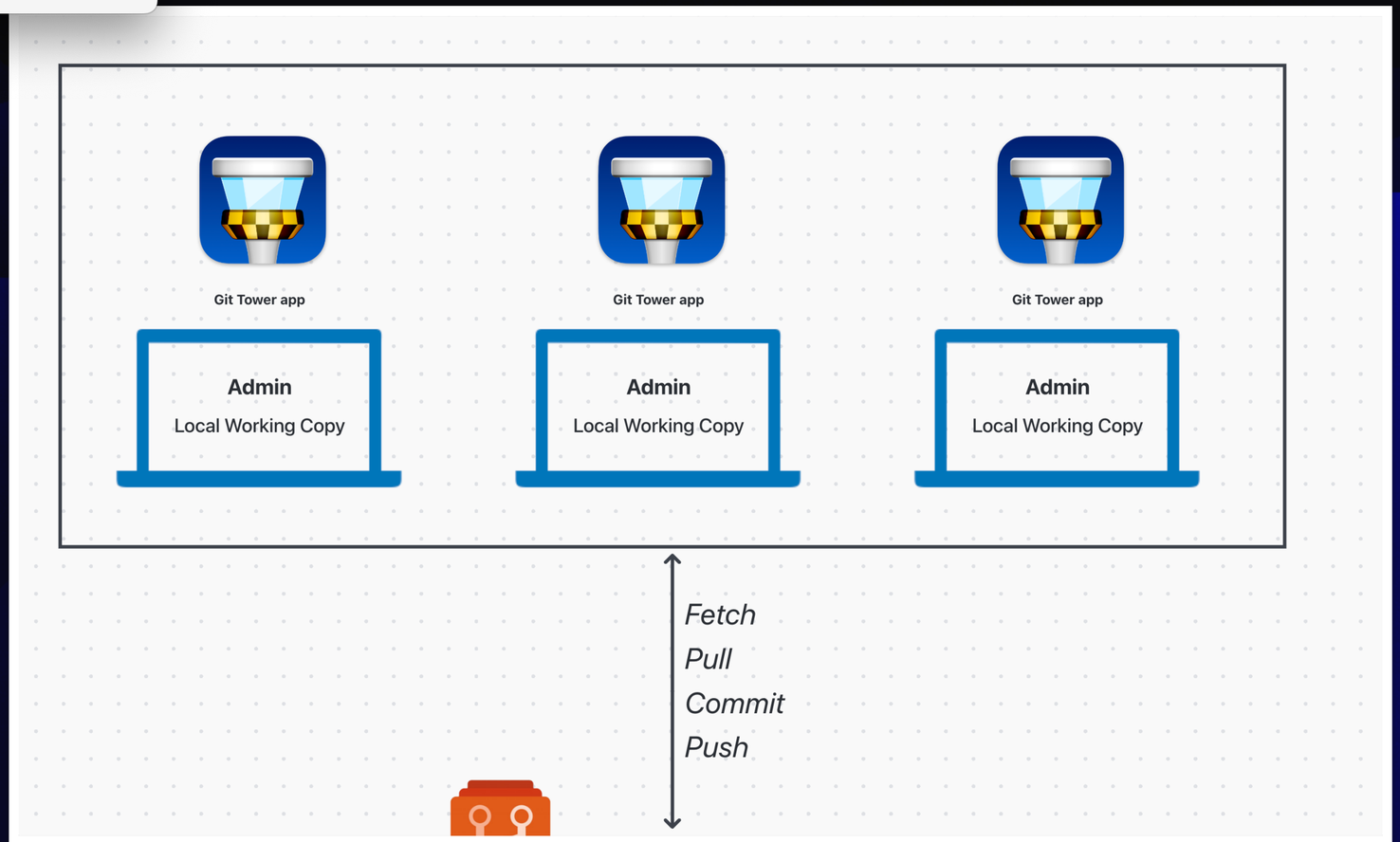
Each admin has the entire repo locally

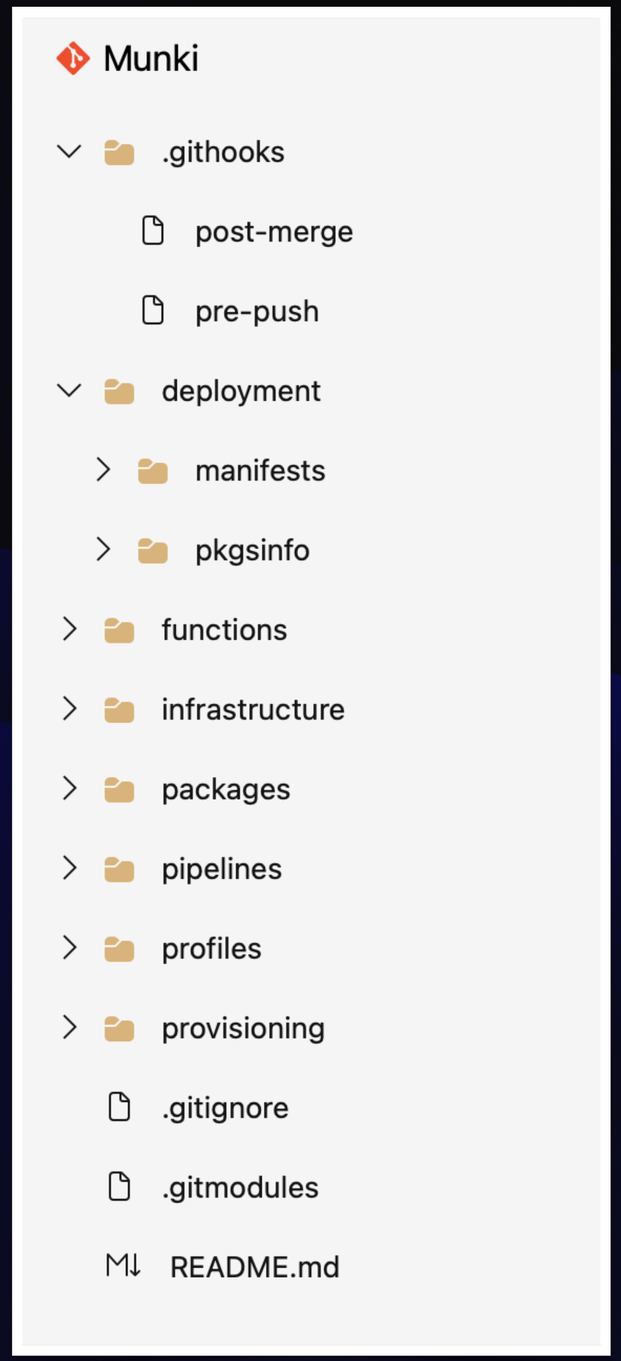


Each admin has the entire repo locally









Repo

`./.githooks` - triggers pkgs download/upload

`./deployment/manifests`

`./deployment/pkgsinfo`

`./functions` - func source code (such as Basic Auth)

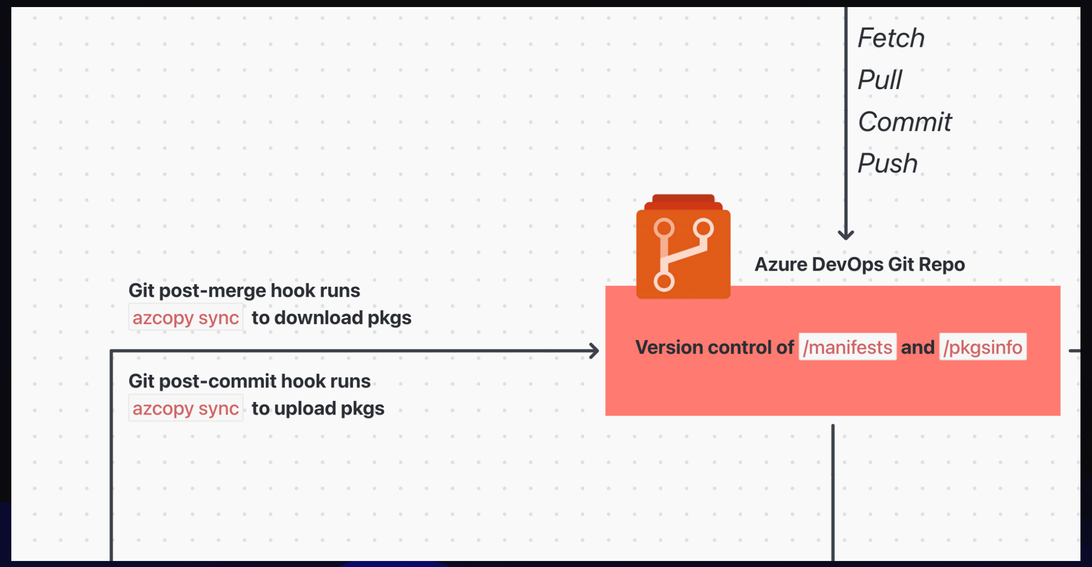
`./infrastructure` - terraform plan for munki infra

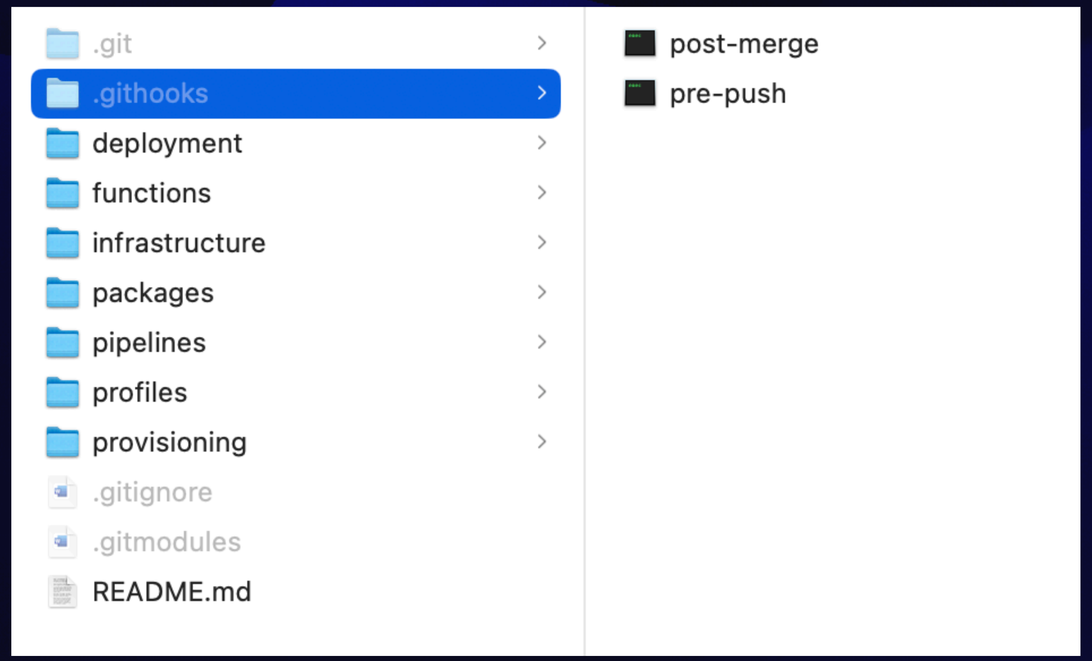
`./packages` - source of all munki pkgs projects

`./pipelines` - all munki repo related CI/CD YAML files

`./profiles` - source of all .mobileconfig we deploy

`./provisioning` - bootstrapping munki workflows





Git Hooks

Ties together upload/download of manfiests, pkginfo, and pkgs at the same time.

Git Pull runs `post-merge` hook to download

Git Push runs `pre-push` hook to upload

Keep your hooks in `./.githooks` and set

```
git config core.hooksPath .githooks
```

No middleware repo plugin - sync headache

```
#!/usr/bin/env zsh
#
# ──────────────────────────────────────────────────────────────────────────────
#  post-merge hook  —  refreshes local Munki cache after every merge / pull
#
#  • Fires automatically after:
#       - `git pull` (even fast-forward)         - `git merge`
#       - `git checkout <branch>`                - rebases that move HEAD
#
#  • Mirrors four folder sets down from S3:
#       1. /deployment/icons     (exact mirror, --delete)
#       2. /deployment/catalogs  (exact mirror, --delete)
#       3. /deployment/pkgs      (exact mirror, --delete)
#
#  • Unified include/exclude rule-set starts with `--exclude *` so hashed
#    filenames never match and are skipped.
#
#  • Daily rotating log kept in .git/logs (7-day retention).
# ──────────────────────────────────────────────────────────────────────────────

set -euo pipefail
set -f
setopt null_glob

BASE="$HOME/DevOps/Munki"
BUCKET='s3://ecuad-macadmins-munki-cloud-s3-bucket/repo'
AWS=$(command -v aws)

# ── logging ──────────────────────────────────────────────
LOGDIR="$BASE/.git/logs"
mkdir -p "$LOGDIR"
TODAY=$(date +%F)
LOGFILE="$LOGDIR/hook-post-merge-download-$TODAY.log"
: >"$LOGFILE"
find "$LOGDIR" -name 'hook-post-merge-download-*.log' -mtime +7 -exec rm {} \;

log() {
  local ts=$(date '+%Y-%m-%d %H:%M:%S')
  printf '%s %s\n' "$ts" "$*" | tee -a "$LOGFILE"
  logger -t 'ca.ecuad.macadmins.munki.post-merge' "$*"
}

next_index() {
  local max=0
  for f in "$LOGDIR"/hook-post-merge-download-"$TODAY"-*; do
    [[ -e $f ]] || continue
    local n=${f##*-}; n=${n%.log}
    (( n > max )) && max=$n
  done
  echo $((max+1))
}

# ── branch gate ──────────────────────────────────────────
current_branch=$(git symbolic-ref --quiet --short HEAD || true)
if [[ "$current_branch" != "main" ]]; then
  log "Skipping downloads — we are in branch '$current_branch', not main."
  mv "$LOGFILE" "${LOGFILE%.*}-$(next_index).log"
  exit 0
fi

log "On branch 'main' — running downloads"

# ── open Console for live view ───────────────────────────
open -a Console "$LOGFILE" || true
trap 'killall Console 2>/dev/null || true' EXIT

# ── include / exclude rules ──────────────────────────────
typeset -a RULES=(
  --exclude '*'
  --include '*.yaml' --include '*.yml' --include '*.json' --include '*.plist'
  --include '*.sh' --include '*.py'
  --include '*.pkg' --include '*.dmg' --include '*.zip' --include '*.tar.gz'
  --include '*.png' --include '*.icns' --include '*.svg'
  --exclude '.DS_Store' --exclude '**/.DS_Store' --exclude '*.git/*'
)

typeset -a PAYLOAD_RULES=( --exclude '*' --include '*/payload/**' )

# ── downloads ────────────────────────────────────────────
log '▶ downloading deployment/icons'
"$AWS" s3 sync "$BUCKET/deployment/icons/"    "$BASE/deployment/icons"    \
  "${RULES[@]}" --delete | tee -a "$LOGFILE"

log '▶ downloading deployment/catalogs'
"$AWS" s3 sync "$BUCKET/deployment/catalogs/" "$BASE/deployment/catalogs" \
  "${RULES[@]}" --delete | tee -a "$LOGFILE"

log '▶ downloading deployment/pkgs'
"$AWS" s3 sync "$BUCKET/deployment/pkgs/"     "$BASE/deployment/pkgs"     \
  "${RULES[@]}" --delete | tee -a "$LOGFILE"

log 'Git post-merge hook complete'
mv "$LOGFILE" "${LOGFILE%.*}-$(next_index).log"
```

post-merge

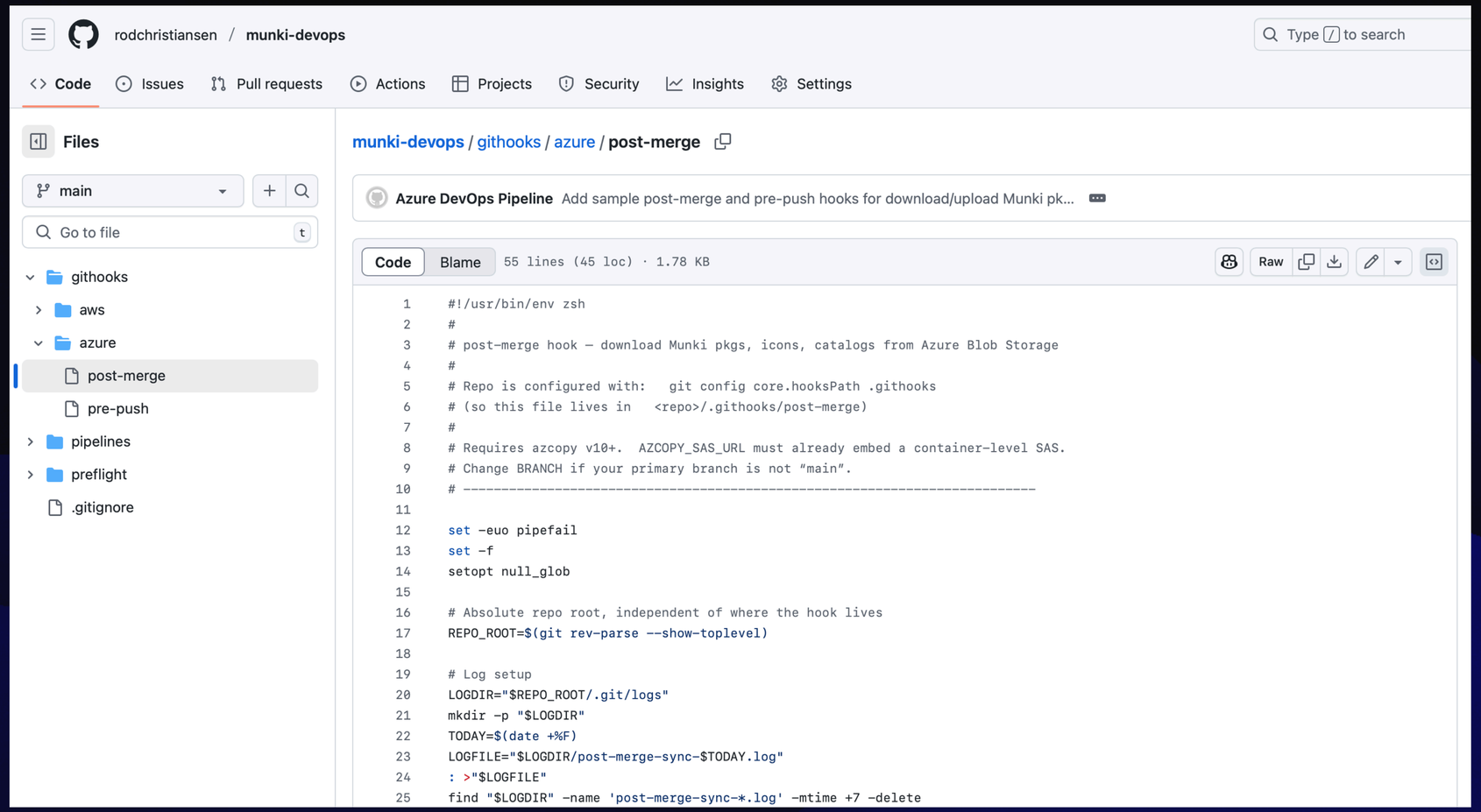
Triggers after every pull/merge/checkout on main

Syncs icons, catalogs, and pkgs from cloud bucket to local

Strict error handling plus daily rotating log in .git/logs

Skips when not on main

Idempotent, repeat runs are harmless



Azure hook sample:

https://github.com/rodchristiansen/munki-devops/blob/main/githooks/azure/post-merge

AWS hook sample:

https://github.com/rodchristiansen/munki-devops/blob/main/githooks/aws/post-merge

```
#!/usr/bin/env zsh
#
# ──────────────────────────────────────────────────────────────────────────
#  pre-push hook  —  safe bucket upload for Munki repo
#
#  • Only runs the bucket upload when pushing *main*.
#      — Any other branch → skip upload, allow push.
#  • If on main, aborts when behind origin/main (unless it can fast-forward).
#  • Daily log rotation, Console.app live view, unified include/exclude rules.
# ──────────────────────────────────────────────────────────────────────────
set -euo pipefail
set -f
setopt null_glob

BASE="$HOME/DevOps/Munki"
BUCKET='s3://ecuad-macadmins-munki-cloud-s3-bucket/repo'
AWS=$(command -v aws)

# ── logging ──────────────────────────────────────────────────────────────
LOGDIR="$BASE/.git/logs"
mkdir -p "$LOGDIR"
TODAY=$(date +%F)
LOGFILE="$LOGDIR/hook-pre-push-upload-$TODAY.log"
: >"$LOGFILE"
find "$LOGDIR" -name 'hook-pre-push-upload-*.log' -mtime +7 -exec rm {} \;

open -a Console "$LOGFILE" || true
trap 'killall Console 2>/dev/null || true' EXIT

log() {
  local ts=$(date '+%Y-%m-%d %H:%M:%S')
  printf '%s %s\n' "$ts" "$*" | tee -a "$LOGFILE"
  logger -t 'ca.ecuad.macadmins.munki.cloud-repo-upload' "$*"
}

next_index() {
  local max=0
  for f in "$LOGDIR"/hook-pre-push-upload-"$TODAY"-*; do
    [[ -e $f ]] || continue
    local n=${f##*-}; n=${n%.log}
    (( n > max )) && max=$n
  done
  echo $((max+1))
}

# ── branch gating ────────────────────────────────────────────────────────
current_branch=$(git symbolic-ref --quiet --short HEAD || true)

if [[ "$current_branch" != "main" ]]; then
  log "Skipping uploads — we are in branch '$current_branch', not main."
  mv "$LOGFILE" "${LOGFILE%.*}-$(next_index).log"
  exit 0          # allow push to proceed with no upload
fi

# ── safety guard (main only) ─────────────────────────────────────────────
git fetch --quiet origin main

behind=$(git rev-list --count HEAD..origin/main || echo 0)

if (( behind > 0 )); then
  log "Local main is $behind commit(s) behind origin/main — running git pull first"

  if git pull --ff-only origin main; then
    log "Git Pull with fast-forward successful — now at $(git rev-parse --short
HEAD)"
  else
    log "Push cancelled: local main is missing commits from origin/main and needs a
manual merge or rebase."
    exit 1
  fi
fi

log "On branch 'main' — running uploads"

# ── include / exclude rules ──────────────────────────────────────────────
typeset -a RULES=(
  --exclude '*'
  --include '*.pkg'  --include '*.dmg'  --include '*.zip'  --include '*.tar.gz'
  --include '*.png'  --include '*.icns' --include '*.svg'
  --include '*.yaml' --include '*.yml'  --include '*.json' --include '*.plist'
  --exclude '.DS_Store' --exclude '**/.DS_Store' --exclude '*.git/*'
)

# ── /deployment/icons ────────────────────────────────────────────────────
log '▶ uploading deployment/icons'
"$AWS" s3 sync "$BASE/deployment/icons" "$BUCKET/deployment/icons/" \
  "${RULES[@]}" --delete | tee -a "$LOGFILE"

# ── /deployment/pkgs ─────────────────────────────────────────────────────
log '▶ uploading deployment/pkgs'
"$AWS" s3 sync "$BASE/deployment/pkgs" "$BUCKET/deployment/pkgs/" \
  "${RULES[@]}" --delete | tee -a "$LOGFILE"

log '✔ pre-push completed'
mv "$LOGFILE" "${LOGFILE%.*}-$(next_index).log"
```

pre-push

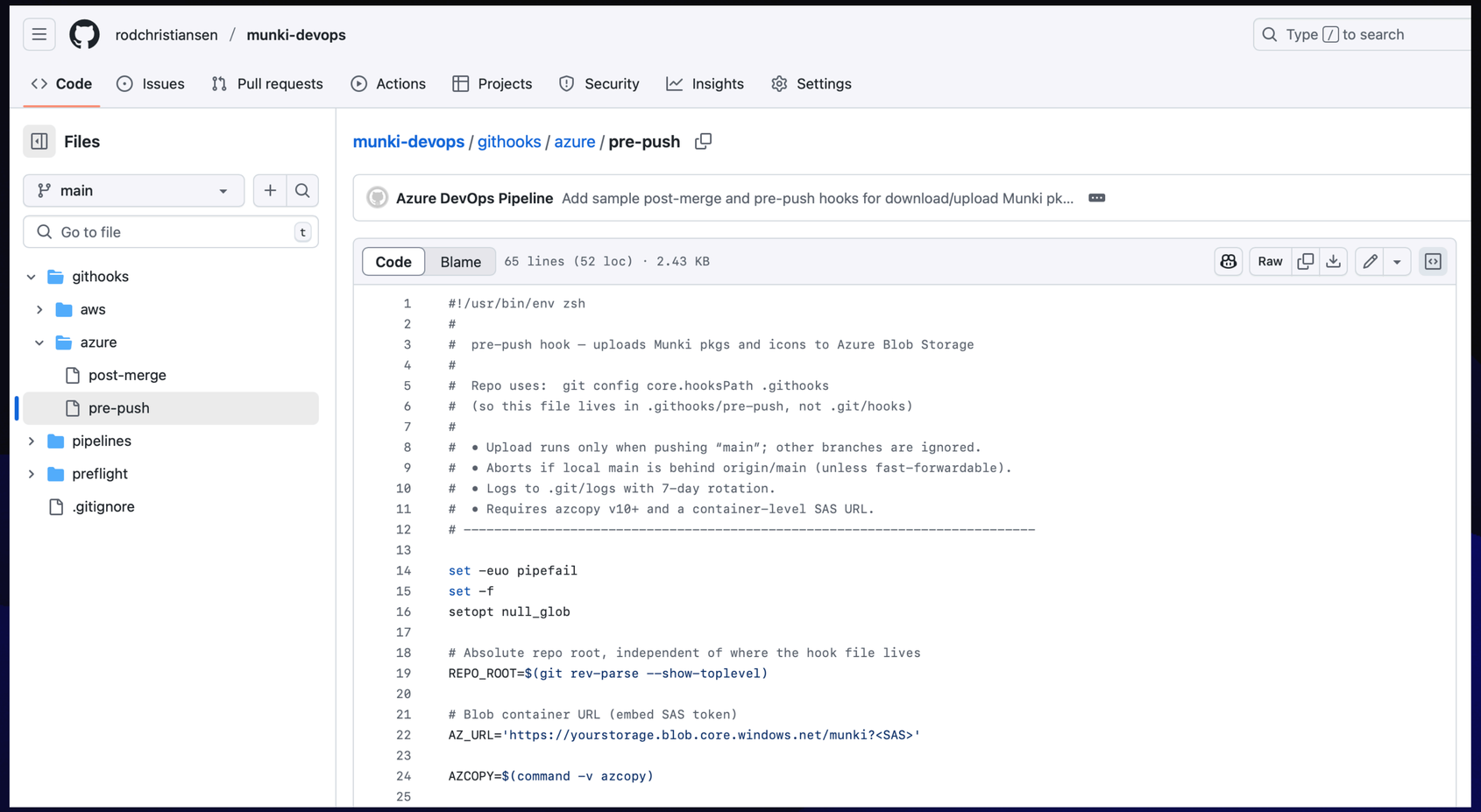
Triggers with git push, so you can collect multiple commits

Uploads icons and pkgs folder to cloud bucket

Runs only when pushing main

Fast-forwards if behind origin; otherwise aborts the push

Daily rotating log in .git/logs

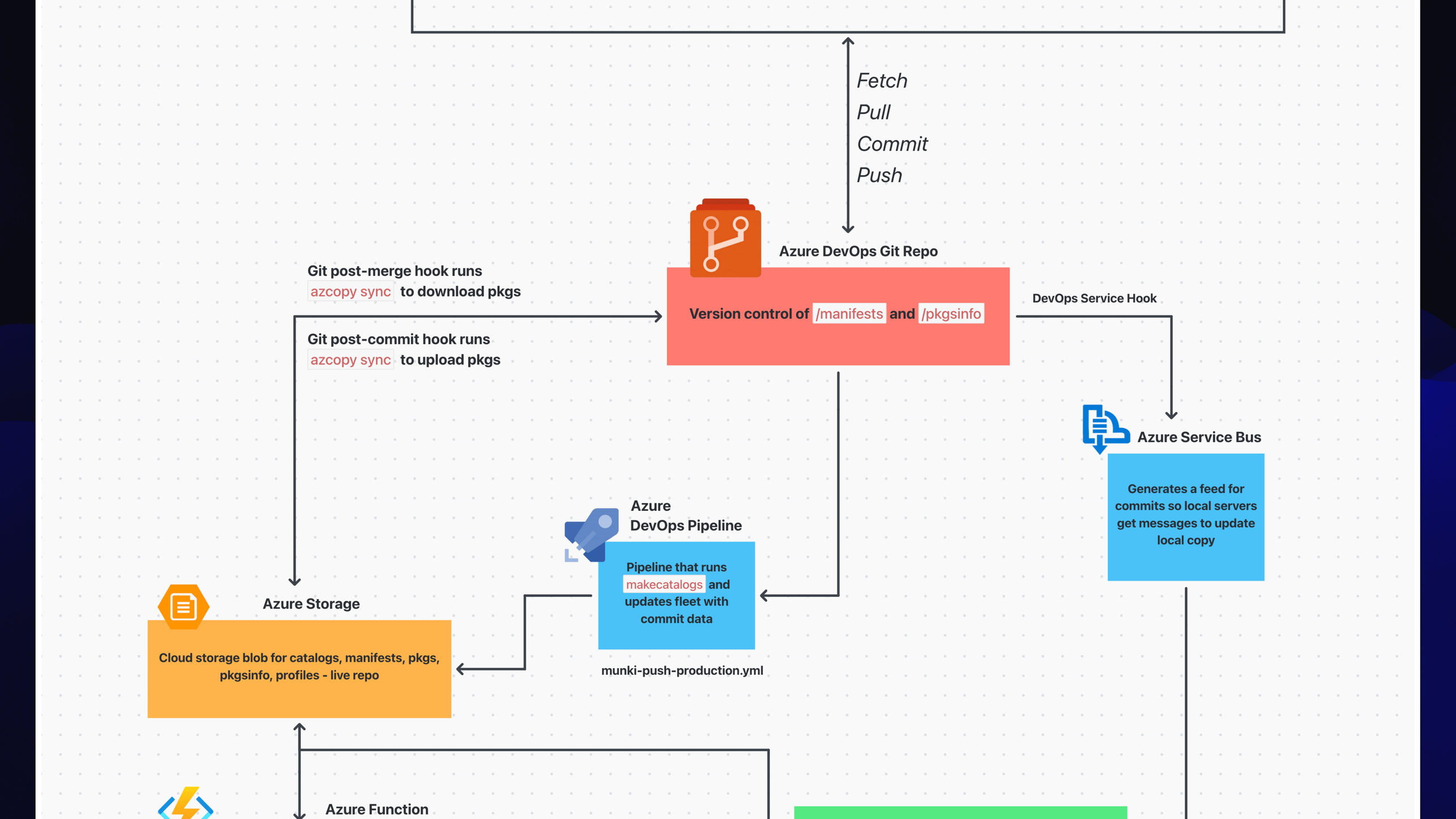


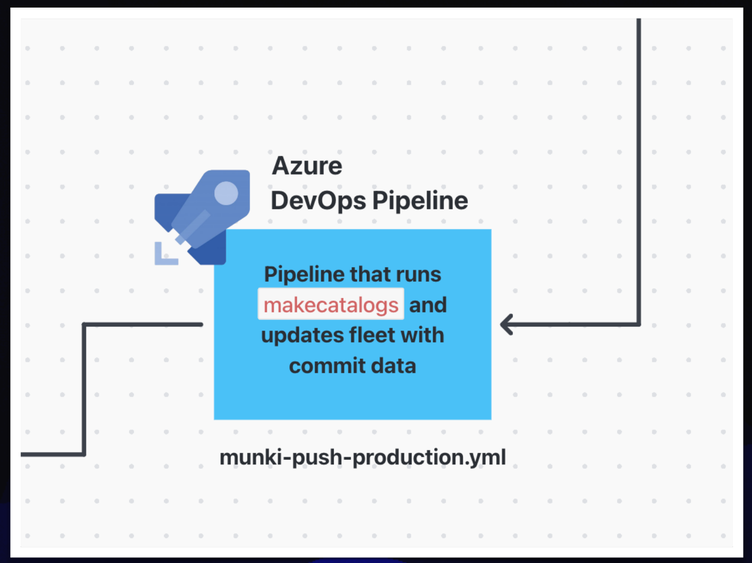
Azure hook sample:

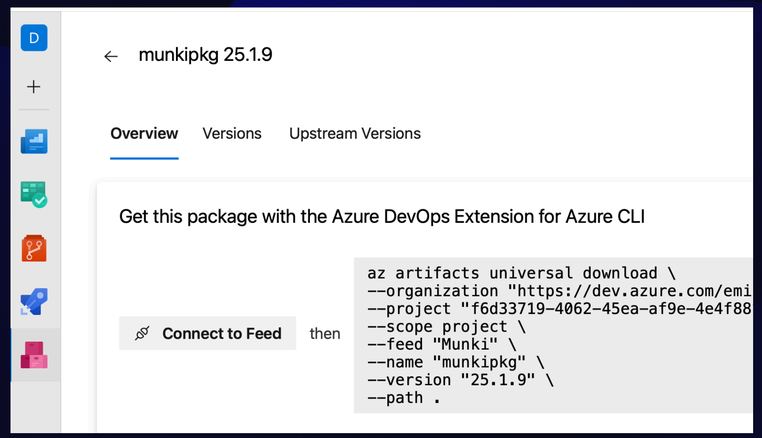
https://github.com/rodchristiansen/munki-devops/blob/main/githooks/azure/pre-push

AWS hook sample:

https://github.com/rodchristiansen/munki-devops/blob/main/githooks/aws/pre-push


Fetch
Pull
Commit
Push
Azure DevOps Git Repo
Git post-merge hook runs
azcopy sync to download pkgs
Git post-commit hook runs
azcopy sync to upload pkgs
Version control of /manifests and /pkgsinfo
DevOps Service Hook
Azure Service Bus
Generates a feed for commits so local servers get messages to update local copy
Azure
DevOps Pipeline
Pipeline that runs makecatalogs and updates fleet with commit data
munki-push-production.yml
Azure Storage
Cloud storage blob for catalogs, manifests, pkgs, pkgsinfo, profiles - live repo
Azure Function





Push to Prod Pipeline

Triggers when repo changes on main

Spins up a macOS hosted cloud vm agent

Installs Munki tools from an artifact with choices xml

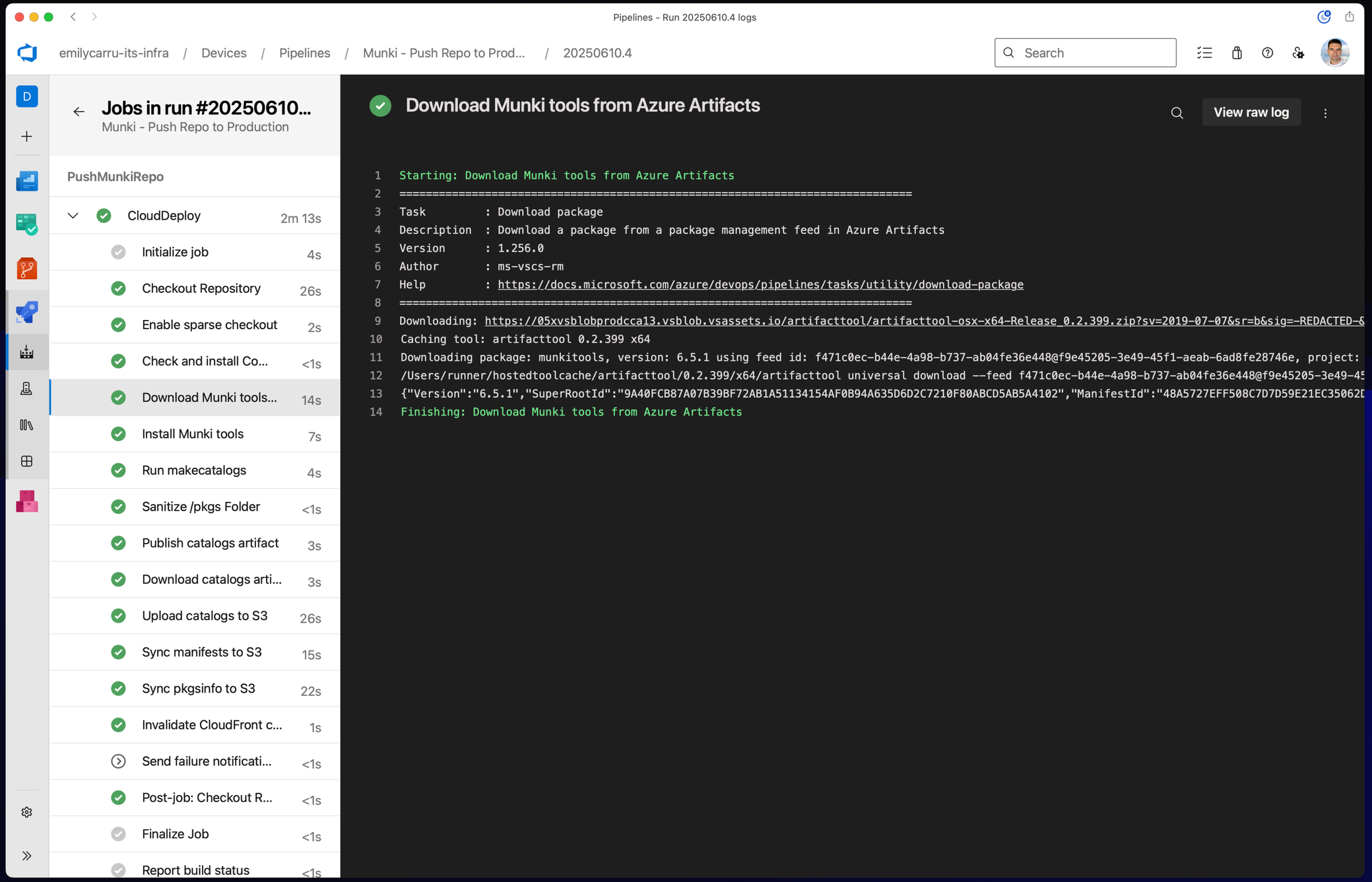
`makecatalogs` builds fresh catalogs

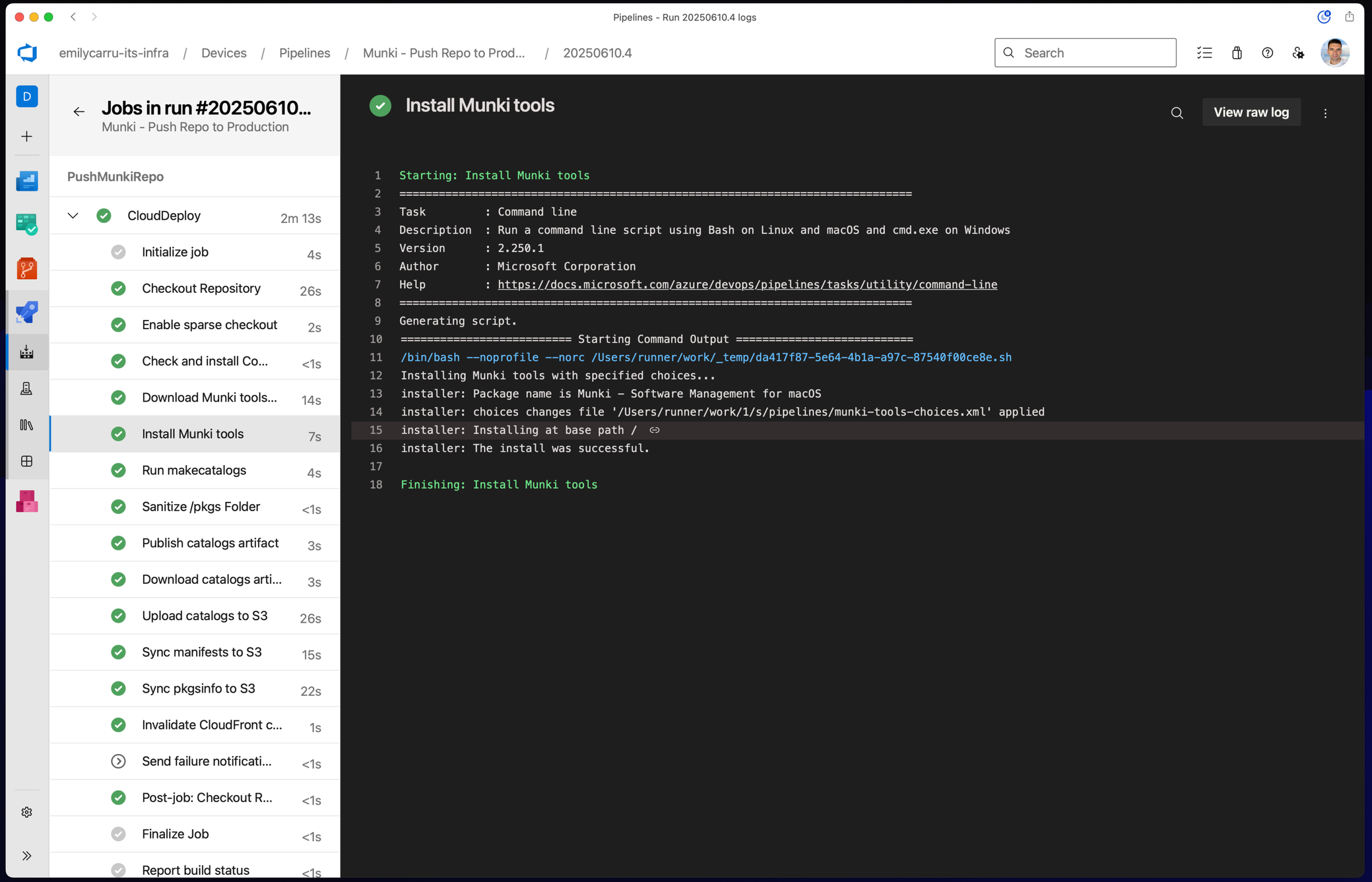
Catalogs pushed to bucket

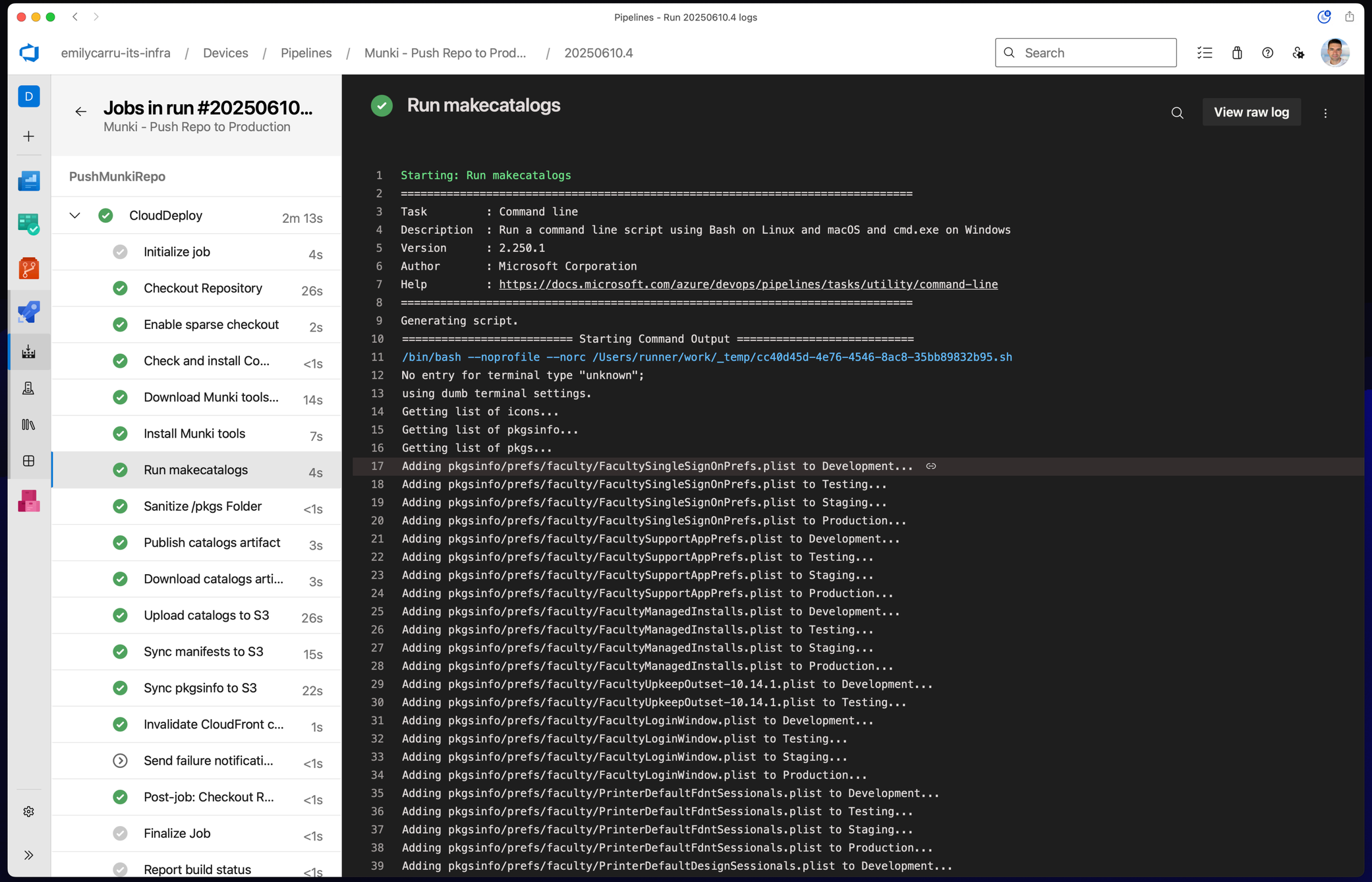
Munki manifests and pkgsinfo folders pushed to bucket

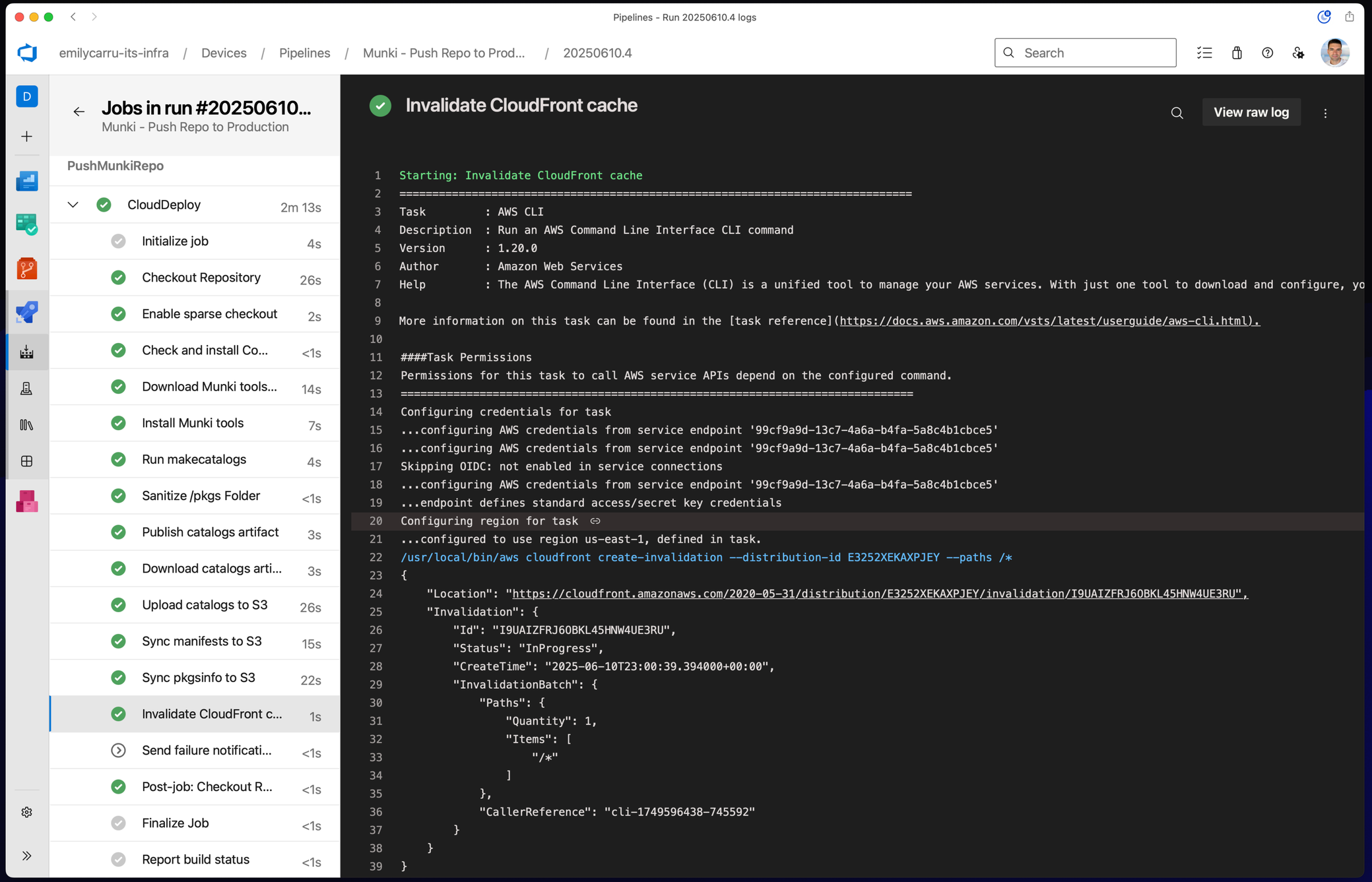
CDN cache purged/invalidated so clients fetch new data

Service Hook sends message to Service Bus


Pipelines - Run 20250610.4 logs
emilycarru-its-infra / Devices / Pipelines / Munki - Push Repo to Prod... / 20250610.4
Search
Jobs in run #20250610...
Munki - Push Repo to Production
PushMunkiRepo
CloudDeploy 2m 13s
Initialize job 4s
Checkout Repository 26s
Enable sparse checkout 2s
Check and install Co... <1s
Download Munki tools... 14s
Install Munki tools 7s
Run makecatalogs 4s
Sanitize /pkgs Folder <1s
Publish catalogs artifact 3s
Download catalogs arti... 3s
Upload catalogs to S3 26s
Sync manifests to S3 15s
Sync pkgsinfo to S3 22s
Invalidate CloudFront c... 1s
Send failure notificati... <1s
Post-job: Checkout R... <1s
Finalize Job <1s
Report build status <1s
Download Munki tools from Azure Artifacts
View raw log
1 Starting: Download Munki tools from Azure Artifacts
2 ==
3 Task : Download package
4 Description : Download a package from a package management feed in Azure Artifacts
5 Version : 1.256.0
6 Author : ms-vscs-rm
7 Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/download-package
8 ==
9 Downloading: https://05xvsblobprodcca13.vsblob.vsassets.io/artifacttool/artifacttool-osx-x64-Release_0.2.399.zip?sv=2019-07-07&sr=b&sig=-REDACTED-&
10 Caching tool: artifacttool 0.2.399 x64
11 Downloading package: munkitools, version: 6.5.1 using feed id: f471c0ec-b44e-4a98-b737-ab04fe36e448@f9e45205-3e49-45f1-aeab-6ad8fe28746e, project:
12 /Users/runner/hostedtoolcache/artifacttool/0.2.399/x64/artifacttool universal download --feed f471c0ec-b44e-4a98-b737-ab04fe36e448@f9e45205-3e49-45
13 {"Version":"6.5.1","SuperRootId":"9A40FCB87A07B39BF72AB1A51134154AF0B94A635D6D2C7210F80ABCD5AB5A4102","ManifestId":"48A5727EFF508C7D7D59E21EC35062D
14 Finishing: Download Munki tools from Azure Artifacts


Pipelines - Run 20250610.4 logs
emilycarru-its-infra / Devices / Pipelines / Munki - Push Repo to Prod... / 20250610.4
Search
Jobs in run #20250610...
Munki - Push Repo to Production
PushMunkiRepo
CloudDeploy 2m 13s
Initialize job 4s
Checkout Repository 26s
Enable sparse checkout 2s
Check and install Co... <1s
Download Munki tools... 14s
Install Munki tools 7s
Run makecatalogs 4s
Sanitize /pkgs Folder <1s
Publish catalogs artifact 3s
Download catalogs arti... 3s
Upload catalogs to S3 26s
Sync manifests to S3 15s
Sync pkgsinfo to S3 22s
Invalidate CloudFront c... 1s
Send failure notificati... <1s
Post-job: Checkout R... <1s
Finalize Job <1s
Report build status <1s
Install Munki tools
View raw log
1 Starting: Install Munki tools
2 ==
3 Task : Command line
4 Description : Run a command line script using Bash on Linux and macOS and cmd.exe on Windows
5 Version : 2.250.1
6 Author : Microsoft Corporation
7 Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/command-line
8 ==
9 Generating script.
10 ========== Starting Command Output ==========
11 /bin/bash --noprofile --norc /Users/runner/work/_temp/da417f87-5e64-4b1a-a97c-87540f00ce8e.sh
12 Installing Munki tools with specified choices...
13 installer: Package name is Munki - Software Management for macOS
14 installer: choices changes file '/Users/runner/work/1/s/pipelines/munki-tools-choices.xml' applied
15 installer: Installing at base path /
16 installer: The install was successful.
17
18 Finishing: Install Munki tools


Pipelines - Run 20250610.4 logs
emilycarru-its-infra / Devices / Pipelines / Munki - Push Repo to Prod... / 20250610.4
Search
Jobs in run #20250610...
Munki - Push Repo to Production
PushMunkiRepo
CloudDeploy 2m 13s
Initialize job 4s
Checkout Repository 26s
Enable sparse checkout 2s
Check and install Co... <1s
Download Munki tools... 14s
Install Munki tools 7s
Run makecatalogs 4s
Sanitize /pkgs Folder <1s
Publish catalogs artifact 3s
Download catalogs arti... 3s
Upload catalogs to S3 26s
Sync manifests to S3 15s
Sync pkgsinfo to S3 22s
Invalidate CloudFront c... 1s
Send failure notificati... <1s
Post-job: Checkout R... <1s
Finalize Job <1s
Report build status <1s
Run makecatalogs
View raw log
Starting: Run makecatalogs
==
Task : Command line
Description : Run a command line script using Bash on Linux and macOS and cmd.exe on Windows
Version : 2.250.1
Author : Microsoft Corporation
Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/command-line
==
Generating script.
========================== Starting Command Output ===========================
/bin/bash --noprofile --norc /Users/runner/work/_temp/cc40d45d-4e76-4546-8ac8-35bb89832b95.sh
No entry for terminal type "unknown";
using dumb terminal settings.
Getting list of icons...
Getting list of pkgsinfo...
Getting list of pkgs...
Adding pkgsinfo/prefs/faculty/FacultySingleSignOnPrefs.plist to Development...
Adding pkgsinfo/prefs/faculty/FacultySingleSignOnPrefs.plist to Testing...
Adding pkgsinfo/prefs/faculty/FacultySingleSignOnPrefs.plist to Staging...
Adding pkgsinfo/prefs/faculty/FacultySingleSignOnPrefs.plist to Production...
Adding pkgsinfo/prefs/faculty/FacultySupportAppPrefs.plist to Development...
Adding pkgsinfo/prefs/faculty/FacultySupportAppPrefs.plist to Testing...
Adding pkgsinfo/prefs/faculty/FacultySupportAppPrefs.plist to Staging...
Adding pkgsinfo/prefs/faculty/FacultySupportAppPrefs.plist to Production...
Adding pkgsinfo/prefs/faculty/FacultyManagedInstalls.plist to Development...
Adding pkgsinfo/prefs/faculty/FacultyManagedInstalls.plist to Testing...
Adding pkgsinfo/prefs/faculty/FacultyManagedInstalls.plist to Staging...
Adding pkgsinfo/prefs/faculty/FacultyManagedInstalls.plist to Production...
Adding pkgsinfo/prefs/faculty/FacultyUpkeepOutset-10.14.1.plist to Development...
Adding pkgsinfo/prefs/faculty/FacultyUpkeepOutset-10.14.1.plist to Testing...
Adding pkgsinfo/prefs/faculty/FacultyLoginWindow.plist to Development...
Adding pkgsinfo/prefs/faculty/FacultyLoginWindow.plist to Testing...
Adding pkgsinfo/prefs/faculty/FacultyLoginWindow.plist to Staging...
Adding pkgsinfo/prefs/faculty/FacultyLoginWindow.plist to Production...
Adding pkgsinfo/prefs/faculty/PrinterDefaultFdntSessionals.plist to Development...
Adding pkgsinfo/prefs/faculty/PrinterDefaultFdntSessionals.plist to Testing...
Adding pkgsinfo/prefs/faculty/PrinterDefaultFdntSessionals.plist to Staging...
Adding pkgsinfo/prefs/faculty/PrinterDefaultFdntSessionals.plist to Production...
Adding pkgsinfo/prefs/faculty/PrinterDefaultDesignSessionals.plist to Development...


Pipelines - Run 20250610.4 logs
emilycarru-its-infra / Devices / Pipelines / Munki - Push Repo to Prod... / 20250610.4
Search
Jobs in run #20250610...
Munki - Push Repo to Production
PushMunkiRepo
CloudDeploy 2m 13s
Initialize job 4s
Checkout Repository 26s
Enable sparse checkout 2s
Check and install Co... <1s
Download Munki tools... 14s
Install Munki tools 7s
Run makecatalogs 4s
Sanitize /pkgs Folder <1s
Publish catalogs artifact 3s
Download catalogs arti... 3s
Upload catalogs to S3 26s
Sync manifests to S3 15s
Sync pkgsinfo to S3 22s
Invalidate CloudFront c... 1s
Send failure notificati... <1s
Post-job: Checkout R... <1s
Finalize Job <1s
Report build status <1s
Invalidate CloudFront cache
View raw log
Starting: Invalidate CloudFront cache
==
Task : AWS CLI
Description : Run an AWS Command Line Interface CLI command
Version : 1.20.0
Author : Amazon Web Services
Help : The AWS Command Line Interface (CLI) is a unified tool to manage your AWS services. With just one tool to download and configure, yo
More information on this task can be found in the [task reference](https://docs.aws.amazon.com/vsts/latest/userguide/aws-cli.html).
####Task Permissions
Permissions for this task to call AWS service APIs depend on the configured command.
==
Configuring credentials for task
...configuring AWS credentials from service endpoint '99cf9a9d-13c7-4a6a-b4fa-5a8c4b1cbce5'
...configuring AWS credentials from service endpoint '99cf9a9d-13c7-4a6a-b4fa-5a8c4b1cbce5'
Skipping OIDC: not enabled in service connections
...configuring AWS credentials from service endpoint '99cf9a9d-13c7-4a6a-b4fa-5a8c4b1cbce5'
...endpoint defines standard access/secret key credentials
Configuring region for task
...configured to use region us-east-1, defined in task.
/usr/local/bin/aws cloudfront create-invalidation --distribution-id E3252XEKAXPJEY --paths /*
{
"Location": "https://cloudfront.amazonaws.com/2020-05-31/distribution/E3252XEKAXPJEY/invalidation/I9UAIZFRJ6OBKL45HNW4UE3RU",
"Invalidation": {
"Id": "I9UAIZFRJ6OBKL45HNW4UE3RU",
"Status": "InProgress",
"CreateTime": "2025-06-10T23:00:39.394000+00:00",
"InvalidationBatch": {
"Paths": {
"Quantity": 1,
"Items": [
"/*"
]
},
"CallerReference": "cli-1749596438-745592"
}
}
}

```
trigger:
  branches:
    include:
      - main
  paths:
    include:
      - /deployment/**

pool:
  vmImage: 'macOS-13'

variables:
  - group: munki-repo-secrets
  - name: S3_BUCKET_NAME
    value: 'ecuad-macadmins-munki-cloud-s3-bucket'
  - name: azureServiceConnection
    value: 'ITS - Infrastructure - DevOps (59d35012-b593-4b2f-bd50-28e666ed12f7)'
  - name: REPO_PATH
    value: '/deployment'
  - name: MUNKI_VERSION
    value: 'latest'

stages:
- stage: PushMunkiRepo
  jobs:
  - job: CloudDeploy
    steps:
    - checkout: self
      clean: true
      persistCredentials: true
      displayName: 'Checkout Repository'

    - script: |
        # Enable sparse checkout to include the necessary paths
        git sparse-checkout init --cone
        git sparse-checkout set deployment pipelines
      displayName: 'Enable sparse checkout'

    - script: |
        # Check and install Command Line Tools if not already installed
        if ! xcode-select -p &>/dev/null; then
          echo "Installing Command Line Tools..."
          sudo xcode-select --install
        else
          echo "Command Line Tools already installed"
        fi
      displayName: 'Check and install Command Line Tools'

    - task: DownloadPackage@1
      inputs:
        packageType: 'upack'
        feed: 'f6d33719-4062-45ea-af9e-4e4f881c1409/f471c0ec-b44e-4a98-b737-ab04fe36e448'
        view: 'f9e45205-3e49-45f1-aeab-6ad8fe28746e'
        definition: '8f932bc8-04eb-4210-8594-2f6c4f4d892f'
        version: '6.5.1'
        downloadPath: '$(System.ArtifactsDirectory)'
      displayName: 'Download Munki tools from Azure Artifacts'

    - script: |
        # Verify if the choices XML file exists and is readable
        if [ ! -f "$(Build.SourcesDirectory)/pipelines/munki-tools-choices.xml" ]; then
          echo "Choices file $(Build.SourcesDirectory)/pipelines/munki-tools-choices.xml not found!"
          exit 1
        fi

        echo "Installing Munki tools with specified choices..."
        retry_count=0
        max_retries=5
        success=false

        while [ $retry_count -lt $max_retries ]; do
          sudo installer -pkg "$(System.ArtifactsDirectory)/munki-tools.pkg" -applyChoiceChangesXML $
(Build.SourcesDirectory)/pipelines/munki-tools-choices.xml -target /
          if [ $? -eq 0 ]; then
            success=true
            break
          fi
          retry_count=$((retry_count + 1))
          echo "Installation failed. Retrying $retry_count/$max_retries..."
          sleep 5
        done

        if [ "$success" = false ]; then
          echo "Failed to install Munki tools after $max_retries attempts"
          exit 1
        fi
      displayName: 'Install Munki tools'

    - script: |
        # Verify installation
        if [ ! -x /usr/local/munki/makecatalogs ]; then
          echo "makecatalogs not found after installation!"
          exit 1
```

Push to Prod Pipeline

Azure DevOps pipeline sample:

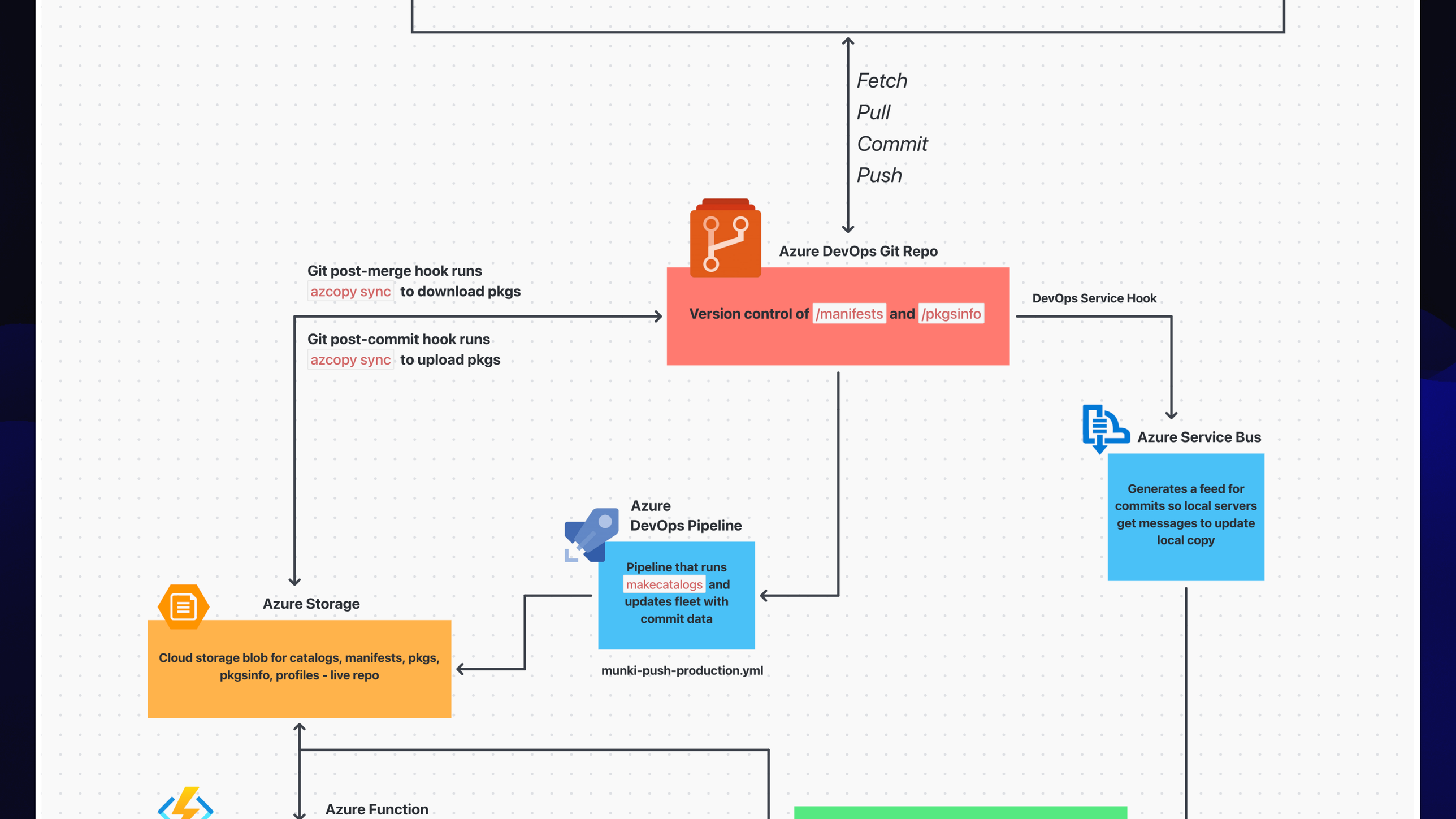
https://github.com/rodchristiansen/munki-devops/blob/main/pipelines/azure/push-to-production-aws.yml

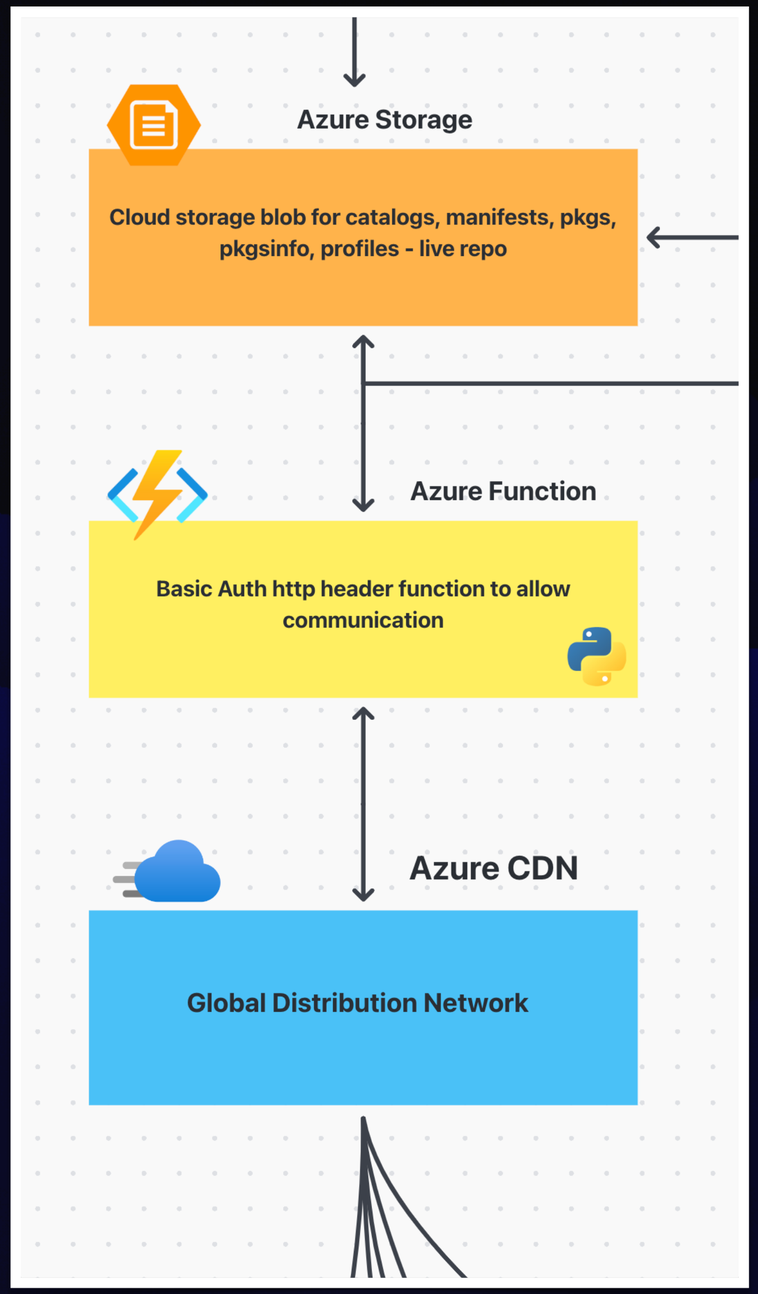
https://github.com/rodchristiansen/munki-devops/blob/main/pipelines/azure/push-to-production-azure.yml

GitHub Action pipeline sample:

https://github.com/rodchristiansen/munki-devops/blob/main/pipelines/github/push-to-production-aws.yml

https://github.com/rodchristiansen/munki-devops/blob/main/pipelines/github/push-to-production-azure.yml


Fetch
Pull
Commit
Push
Azure DevOps Git Repo
Git post-merge hook runs
azcopy sync to download pkgs
Git post-commit hook runs
azcopy sync to upload pkgs
Version control of /manifests and /pkgsinfo
DevOps Service Hook
Azure Service Bus
Generates a feed for commits so local servers get messages to update local copy
Azure
DevOps Pipeline
Pipeline that runs makecatalogs and updates fleet with commit data
munki-push-production.yml
Azure Storage
Cloud storage blob for catalogs, manifests, pkgs, pkgsinfo, profiles - live repo
Azure Function



Munki Cloud Terraform

Azure Storage Blob | AWS S3 Bucket

Everything Munki serves lives here as static objects.

Azure Function | AWS Lambda

A tiny auth shim so only clients get in

Azure FrontDoor | AWS CloudFront

Fast edge cache for off prem

```
data "azurerm_subscription" "current" {
}

# Azure Resource Group
resource "azurerm_resource_group" "rg" {
  name     = "Munki"
  location = "Canada Central"
  tags = {
    GitOps = "Terraformed"
  }
}

# Azure Key Vault with RBAC
resource "azurerm_key_vault" "kv" {
  name                        = "munki-repo-secrets"
  location                    = azurerm_resource_group.rg.location
  resource_group_name         = azurerm_resource_group.rg.name
  sku_name                    = "standard"
  tenant_id                   = var.azure_tenant_id
  soft_delete_retention_days  = 90
  purge_protection_enabled    = true
  enable_rbac_authorization   = true

  network_acls {
    default_action = "Allow"
    bypass         = "AzureServices"
  }
}

# RBAC Role Assignment - Key Vault Administrator
resource "azurerm_role_assignment" "key_vault_admin" {
  scope                = azurerm_key_vault.kv.id
  role_definition_name = "Key Vault Administrator"
  principal_type       = "Group"
  principal_id         = var.devops_resource_infrasec_group_object_id
}

# Create Azure Storage Account with Soft Delete enabled
resource "azurerm_storage_account" "repo" {
  name                     = "munkicloudstorage"
  resource_group_name      = azurerm_resource_group.rg.name
  location                 = azurerm_resource_group.rg.location
  account_tier             = "Standard"
  account_replication_type = "LRS"

  blob_properties {
    delete_retention_policy {
      days = 7
    }
  }
}

# Create an Azure File Share
resource "azurerm_storage_share" "fileshare" {
  name                 = "repo"
  storage_account_name = azurerm_storage_account.repo.name
  quota                = 500
}

resource "azurerm_role_assignment" "smb_contributor" {
  scope                = "/subscriptions/${data.azurerm_subscription.current.subscription_id}/
resourceGroups/${azurerm_resource_group.rg.name}/providers/Microsoft.Storage/storageAccounts/$
{azurerm_storage_account.repo.name}/fileServices/default/shares/$
{azurerm_storage_share.fileshare.name}"
  role_definition_name = "Storage File Data SMB Share Contributor"
  principal_id         = var.devops_resource_infrasec_group_object_id
}

# Role assignment to give the service principal access to the storage account
resource "azurerm_role_assignment" "blob_contributor" {
  scope                = azurerm_storage_account.repo.id
  role_definition_name = "Storage Blob Data Contributor"
  principal_type       = "Group"
  principal_id         = var.devops_resource_infrasec_group_object_id
}

# Create Role Assignment for Blob Access
resource "azurerm_role_assignment" "blob_contributor_devops" {
  scope                = azurerm_storage_account.repo.id
  role_definition_name = "Storage Blob Data Contributor"
  principal_id         = "1eb0e88c-e07d-4f37-8281-b6153b96252d"  # Service Principal for Devices
DevOps Project
}

# Create a Blob Container for the Munki repo
resource "azurerm_storage_container" "repo" {
  name                  = "repo"
  storage_account_name  = azurerm_storage_account.repo.name
  container_access_type = "private"
}

resource "azurerm_storage_container" "logs" {
  name                  = "logs"
  storage_account_name  = azurerm_storage_account.repo.name
```

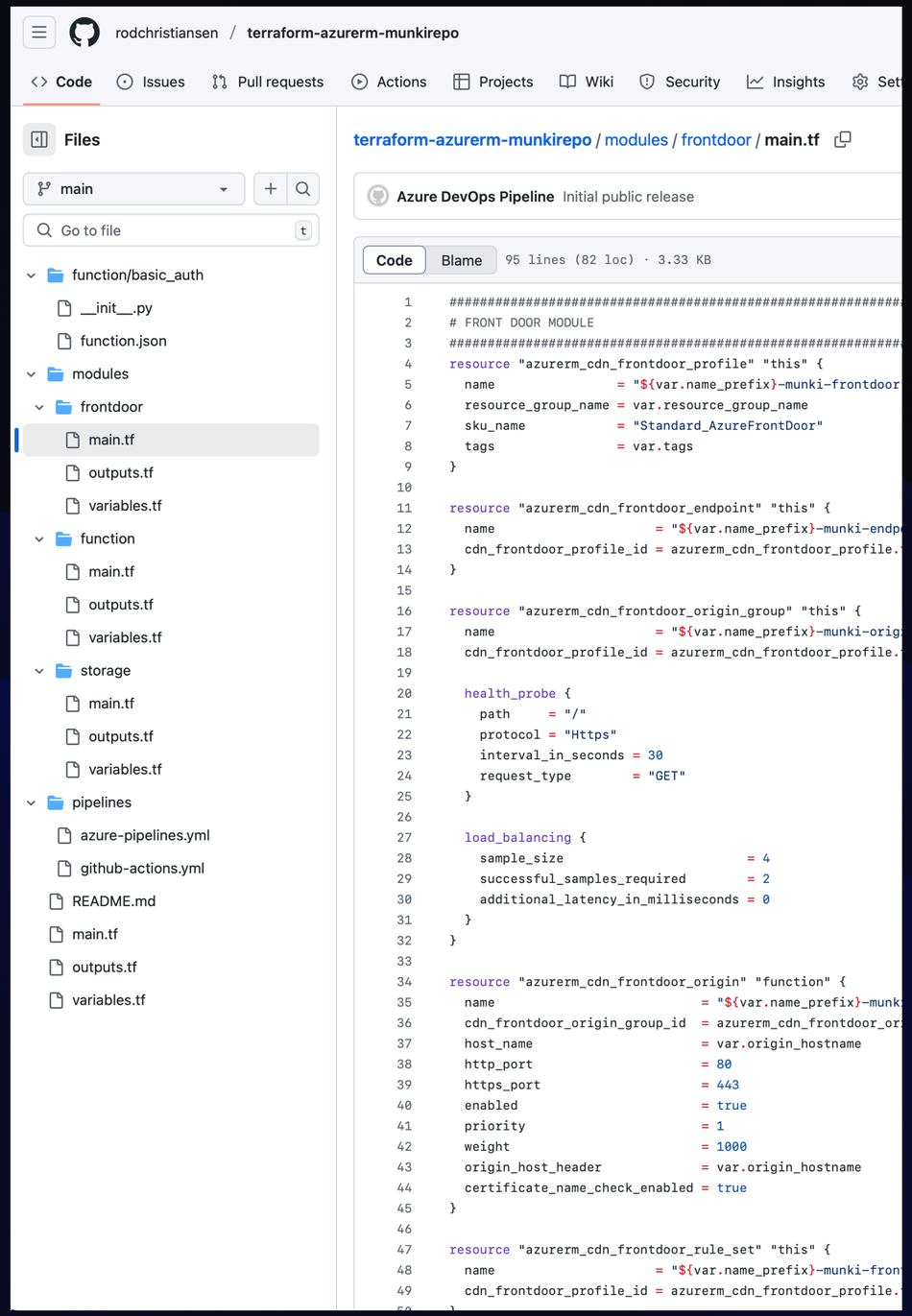
Terraform Munki

Terraform Munki in Azure

https://registry.terraform.io/modules/rodchristiansen/munkirepo/azurerm

Terraform Munki in AWS by Graham Gilbert:

https://registry.terraform.io/modules/grahamgilbert/munki-repo/aws



Terraform Munki Cloud

Terraform Munki in Azure

https://registry.terraform.io/modules/rodchristiansen/munkirepo/azurerm

Terraform Munki in AWS by Graham Gilbert:

https://registry.terraform.io/modules/grahamgilbert/munki-repo/aws

```yaml
trigger:
  branches:
    include: [main]
  paths:
    include:
      - infrastructure/*
      - functions/*

pool:
  vmImage: ubuntu-latest

variables:
- group: repo-secrets
- name: pythonVersion
  value: '3.11'
- name: terraformVersion
  value: '1.7.x'
- name: azureServiceConnection
  value: 'AZURE-SERVICE-CONNECTION'
- name: functionAppName
  value: 'FUNCTION-APP-NAME'
- name: functionAppDirectory
  value: 'functions'

stages:
# ──────────────────── Terraform ────────────────────
- stage: TerraformInfra
  jobs:
  - job: Terraform
    steps:
    - task: TerraformInstaller@1
      inputs:
        terraformVersion: '$(terraformVersion)'

    - task: TerraformTaskV4@4
      displayName: terraform init
      inputs:
        provider: azurerm
        command: init
        backendServiceArm: '$(azureServiceConnection)'
        backendAzureRmResourceGroupName: 'Terraform'
        backendAzureRmStorageAccountName: 'tfstate'
        backendAzureRmContainerName: 'state'
        backendAzureRmKey: 'munki.tfstate'
        workingDirectory: '$(System.DefaultWorkingDirectory)/infrastructure'

    - task: TerraformTaskV4@4
      displayName: terraform plan
      name: plan
      inputs:
        provider: azurerm
        command: plan
        environmentServiceNameAzureRM: '$(azureServiceConnection)'
        workingDirectory: '$(System.DefaultWorkingDirectory)/infrastructure'
        commandOptions: |
          -lock=true
          -lock-timeout=300s
          -var="azure_tenant_id=$(AzureTenantId)"
          -var="azure_storage_connection_string=$(AzureStorageConnectionString)"
          -var="devops_resource_infrasec_group_object_id=$(DevopsResourceInfrasecGroupObjectId)"
          -var="devices_graph_id=$(DevicesGraphId)"
          -var="devices_graph_secret=$(DevicesGraphSecret)"
          -var="munki_username=$(MunkiUsername)"
          -var="munki_password=$(MunkiPassword)"
          -out=tfplan

    - bash: |
        if [ -s tfplan ]; then echo "##vso[task.setvariable variable=TerraformHasChanges]true";
else echo "##vso[task.setvariable variable=TerraformHasChanges]false"; fi
      workingDirectory: '$(System.DefaultWorkingDirectory)/infrastructure'
      displayName: detect changes

    - task: TerraformTaskV4@4
      displayName: terraform apply
      condition: eq(variables['TerraformHasChanges'], 'true')
      inputs:
        provider: azurerm
        command: apply
        environmentServiceNameAzureRM: '$(azureServiceConnection)'
        workingDirectory: '$(System.DefaultWorkingDirectory)/infrastructure'
        commandOptions: '-auto-approve tfplan'

# ──────────────────── Functions ────────────────────
- stage: DeployFunctions
  dependsOn: TerraformInfra
  jobs:
  - job: Deploy
    steps:
    - task: UsePythonVersion@0
      inputs:
        versionSpec: '$(pythonVersion)'
        addToPath: true

    - bash: |
```

Munki Cloud Infra Pipeline

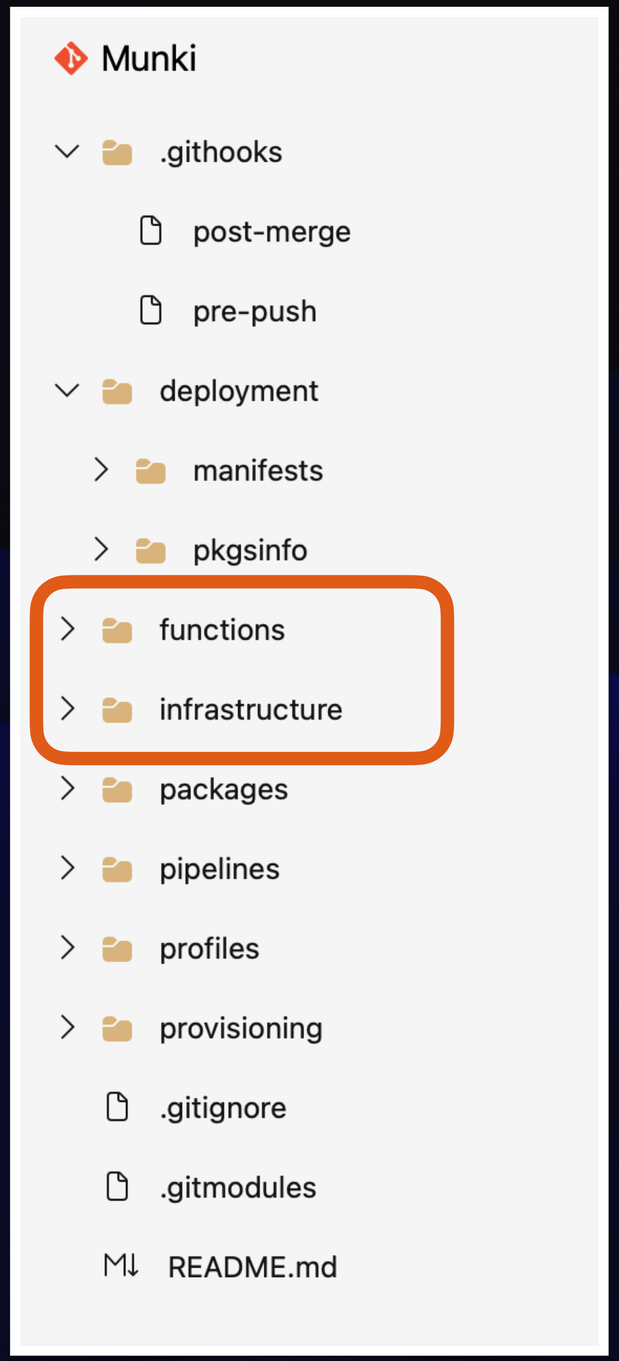
Triggers on main when `infrastructure/` or `functions/` change

Terraform `init` → `plan` → `apply` (if has changes)

Terraform remote state saved in a central Azure Blob

ZIP deployed to Linux Function App via service connection

Secrets come from variable group



Munki Cloud Infra Pipeline

Triggers on main when `infrastructure/` or `functions/` change

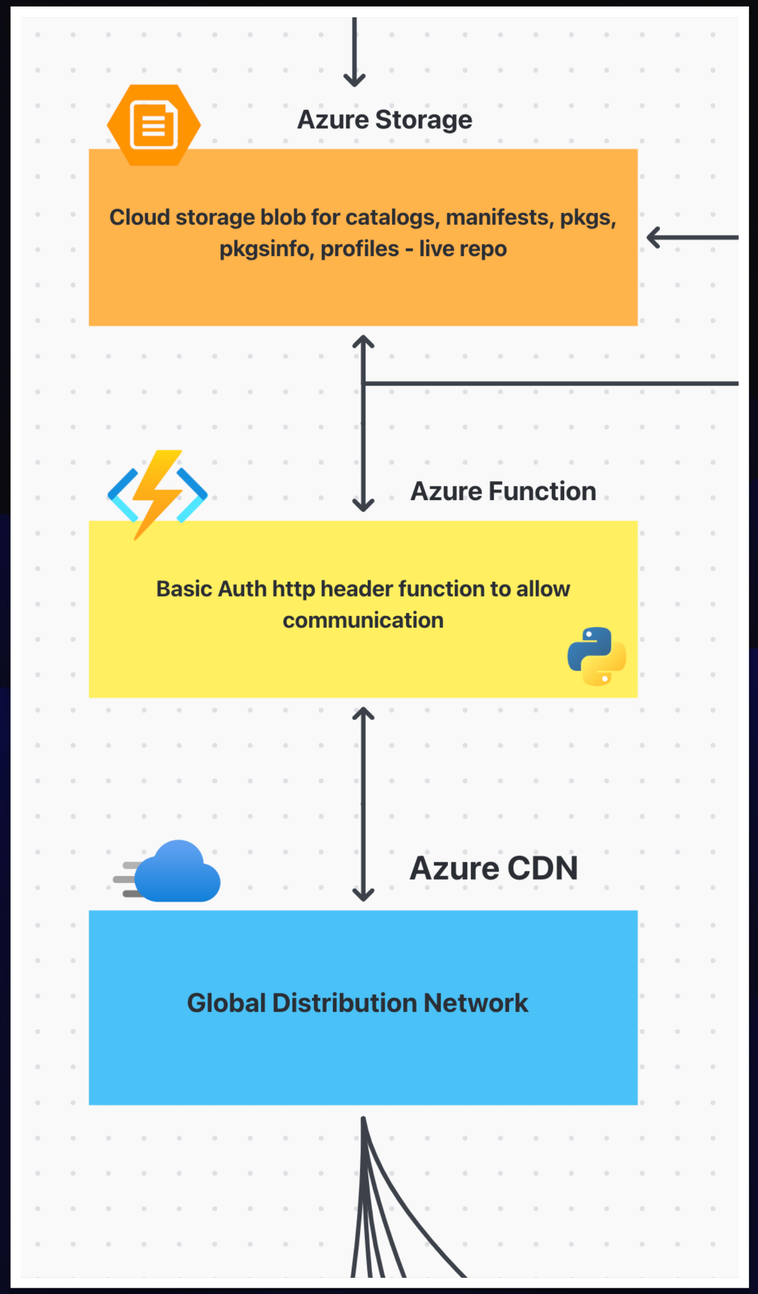
Terraform `init` → `plan` → `apply` (if has changes)

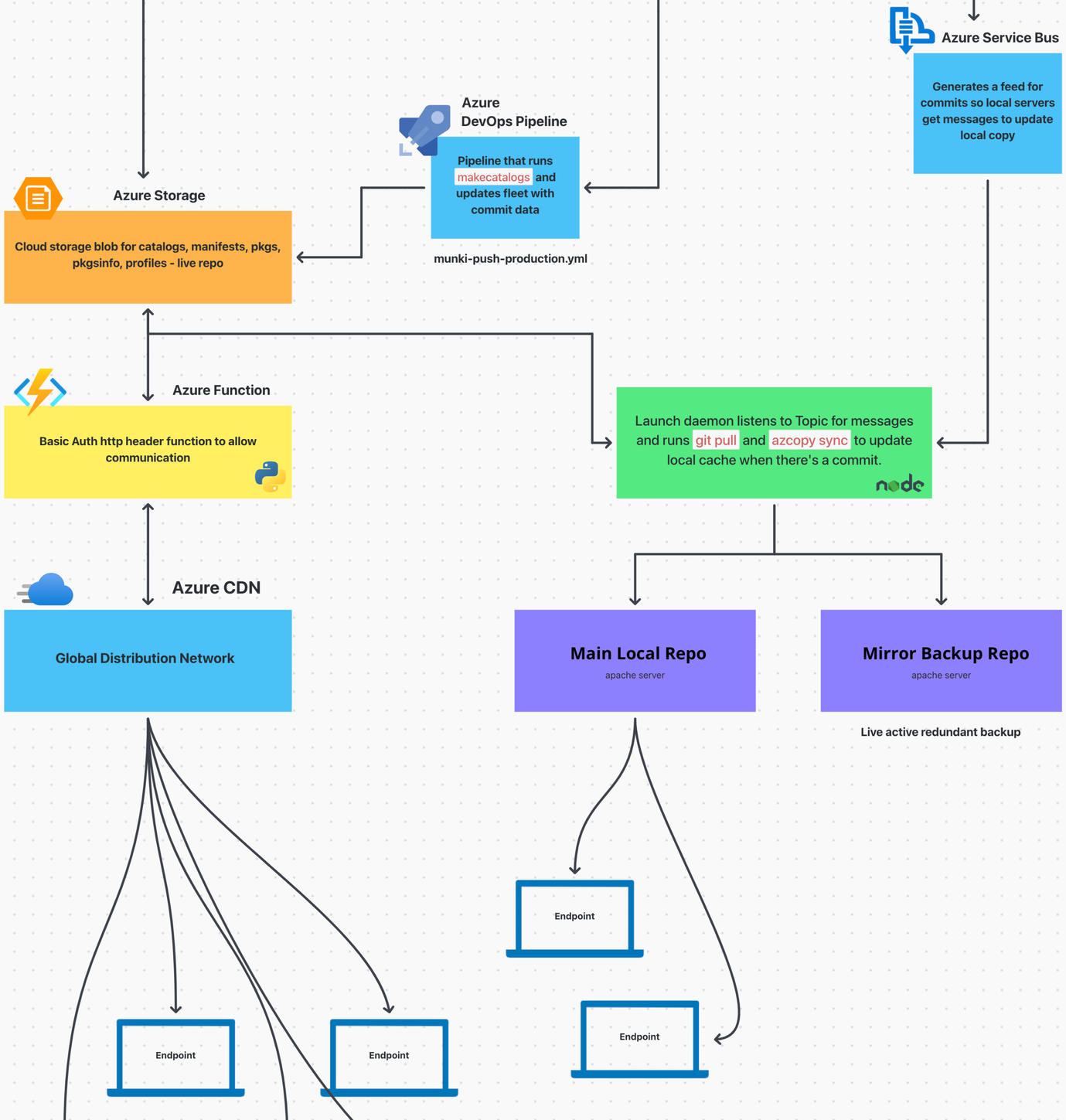
Terraform remote state saved in a central Azure Blob

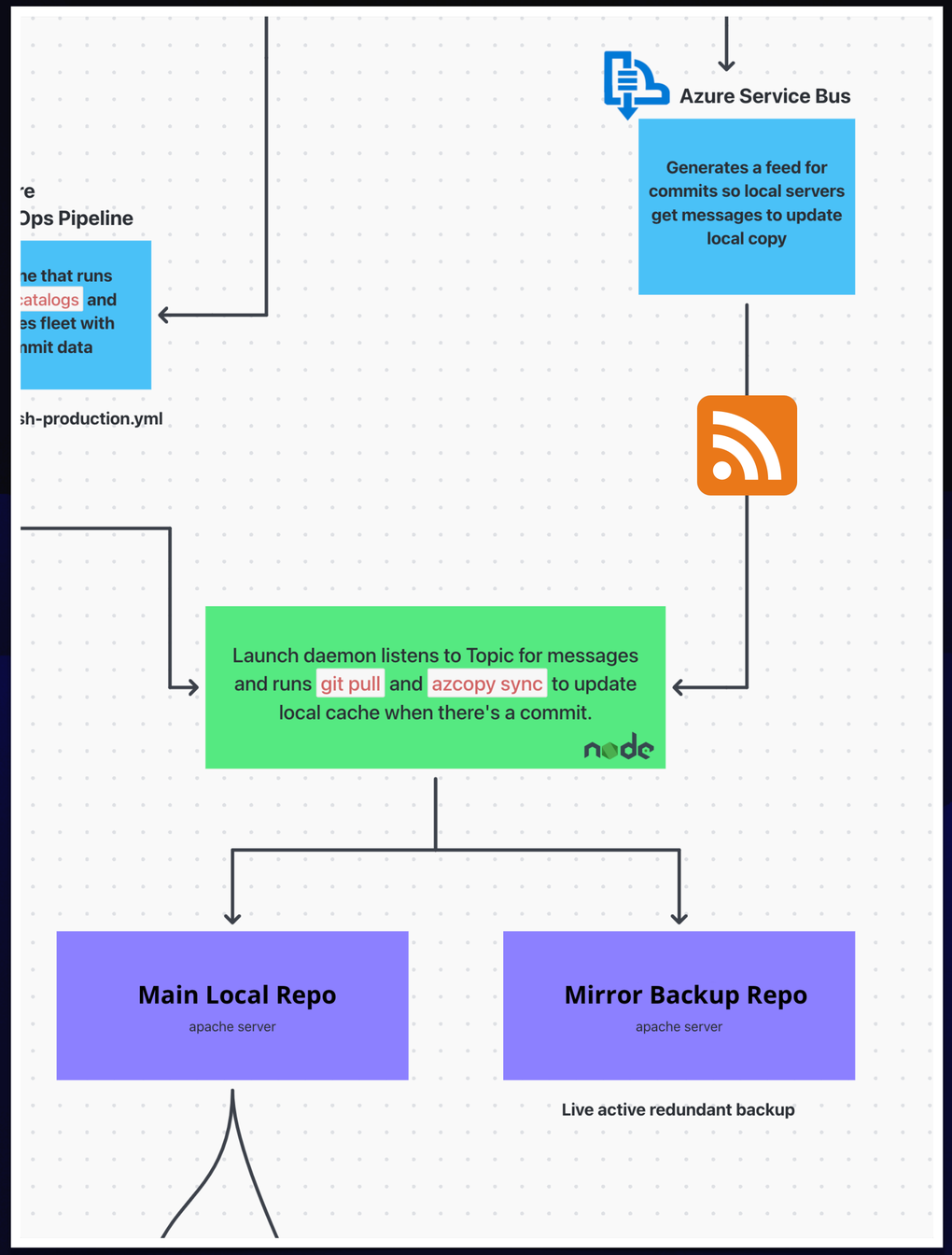
ZIP deployed to Linux Function App via service connection

Secrets come from variable group

Munki Cloud Infra




Azure Service Bus
Generates a feed for commits so local servers get messages to update local copy
Azure DevOps Pipeline
Pipeline that runs makecatalogs and updates fleet with commit data
munki-push-production.yml
Azure Storage
Cloud storage blob for catalogs, manifests, pkgs, pkgsinfo, profiles - live repo
Azure Function
Basic Auth http header function to allow communication
Launch daemon listens to Topic for messages and runs git pull and azcopy sync to update local cache when there's a commit.
node
Azure CDN
Global Distribution Network
Main Local Repo
apache server
Mirror Backup Repo
apache server
Live active redundant backup
Endpoint
Endpoint
Endpoint
Endpoint



Local Caching Servers

Local Mac Studio caching servers

Each server has a dedicated queue it listens to

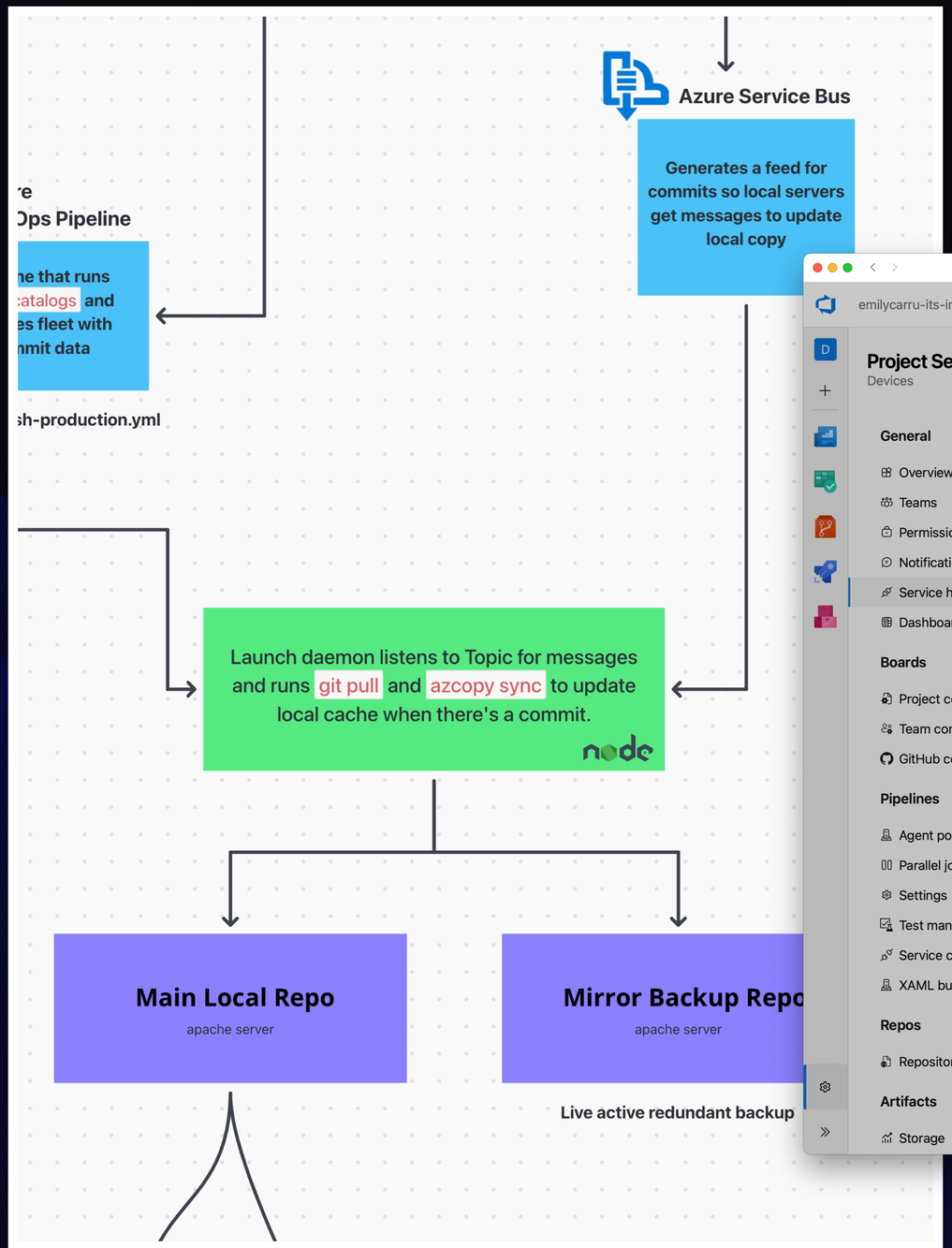
Git force resets local, Git repo is the source of truth

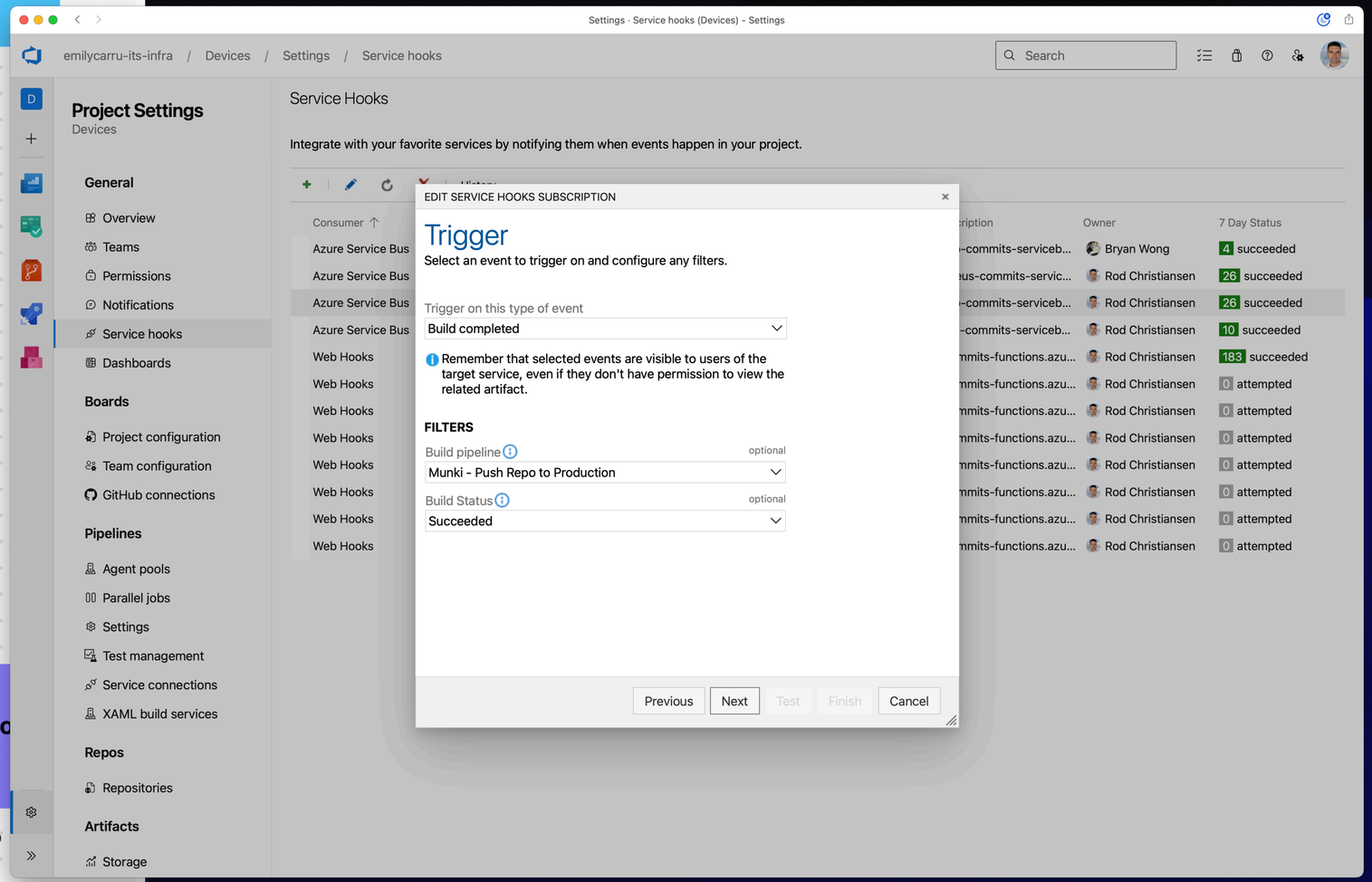
Local on prem FQDN different from cloud repo

Local apache server running MAMP PRO on macOS

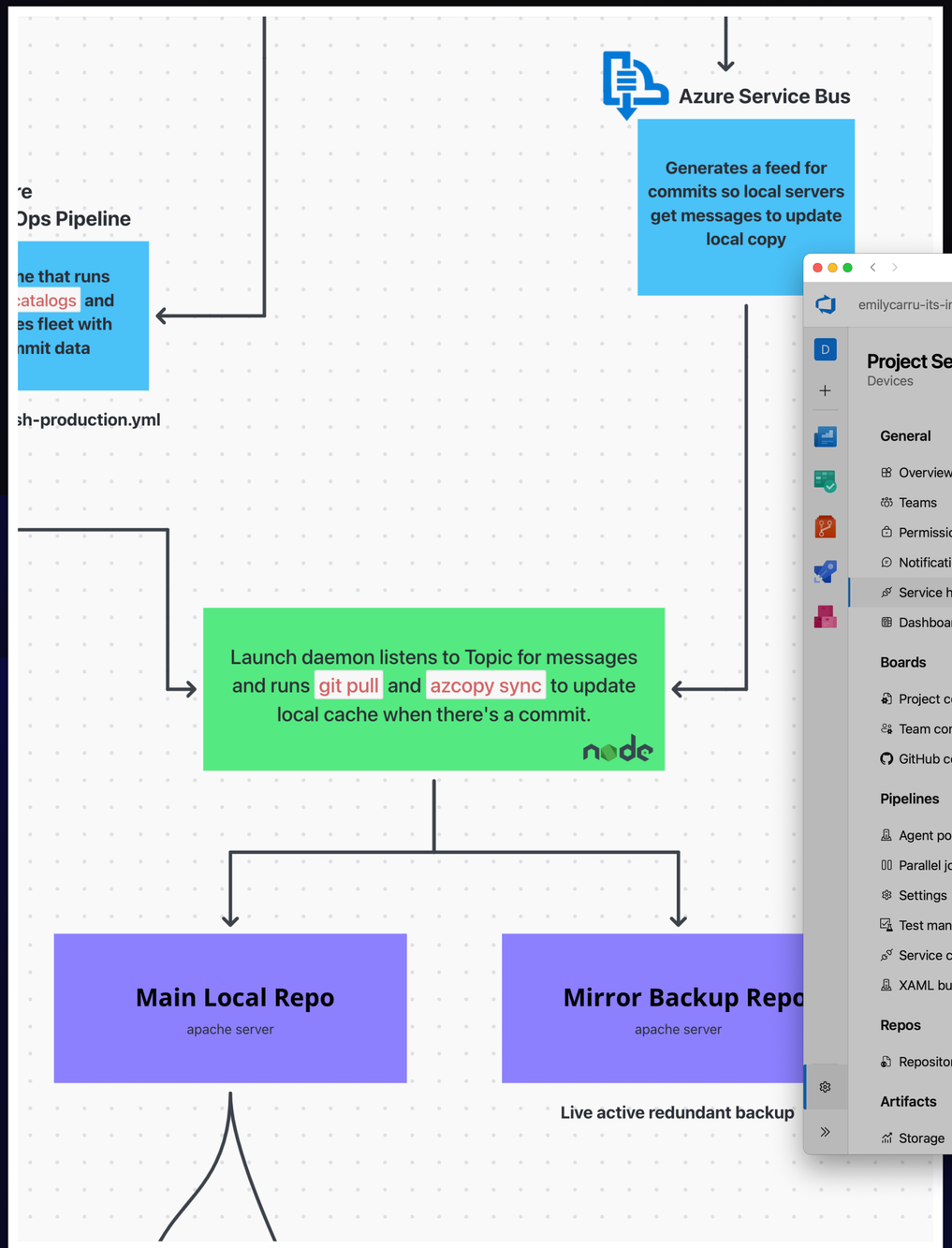
Analogous to a RSS feed

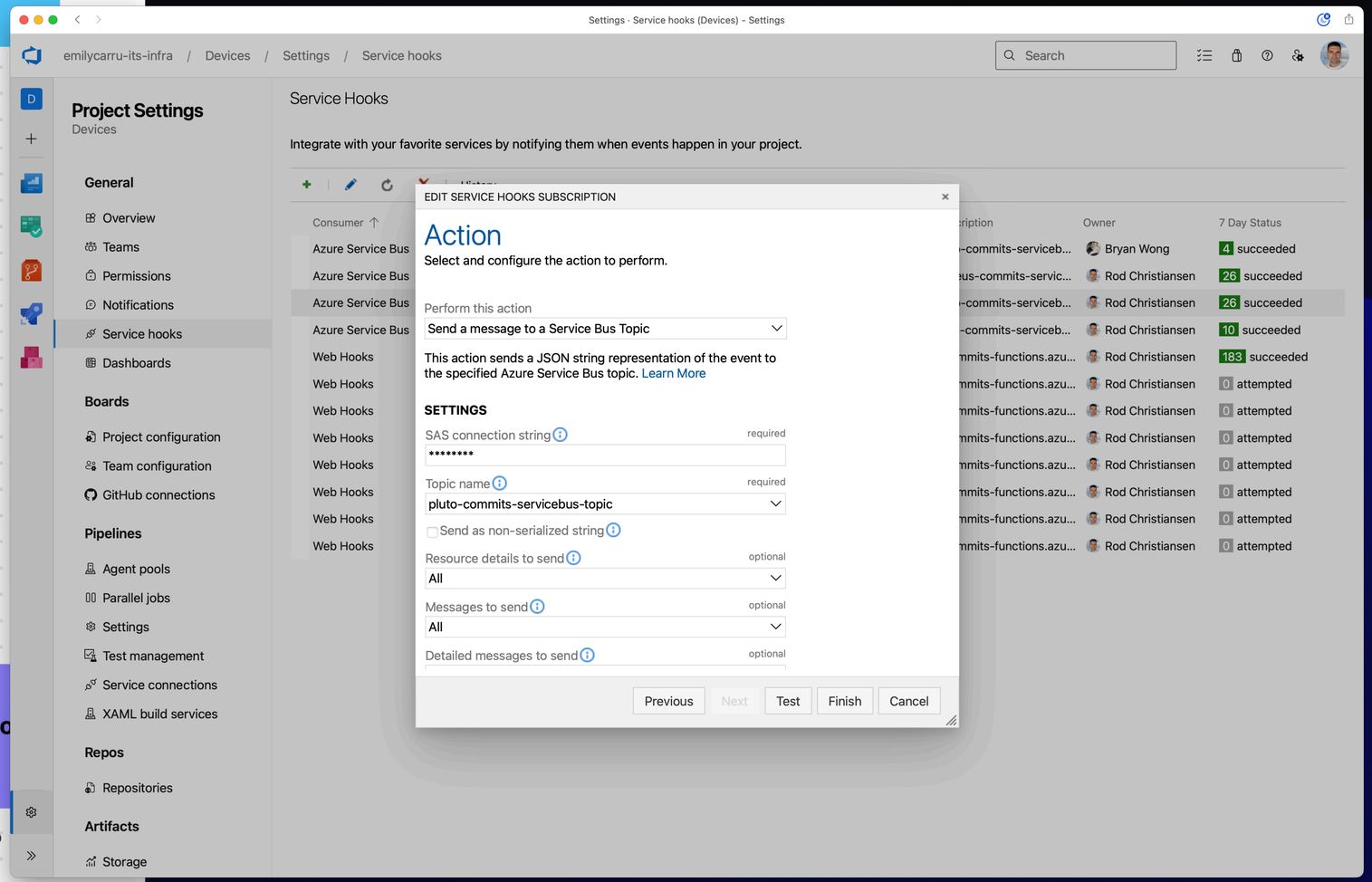
Local Caching Servers





Local Caching Servers





Commits Listener

```
~/DevOps/Munki/packages/CommitsListener
├── build
│   └── CommitsListener-2025.04.16.pkg
├── build-info.json
├── payload
│   ├── LaunchDaemons
│   │   └── ca.ecuad.macadmin.CommitsListener.plist
│   └── Management
│       └── CommitsListener
│           ├── node_modules
│           │   ├── @azure
│           ├── package-lock.json
│           ├── package.json
│           └── service-bus-listener.js
└── scripts
    ├── postinstall
    └── preinstall
```

ca.ecuad.macadmin.CommitsListener.plist

~/DevOps/Munki/packages/CommitsListener/payload/LaunchDaemons/ca.ecuad.macadmin.CommitsListener.plist — main — (functions)

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
3  <plist version="1.0">
4  <dict>
5      <key>Disabled</key>
6      <false/>
7      <key>KeepAlive</key>
8      <true/>
9      <key>Label</key>
10     <string>ca.ecuad.macadmin.CommitsListener</string>
11     <key>ProgramArguments</key>
12     <array>
13         <string>/usr/local/bin/node</string>
14         <string>/Library/Management/CommitsListener/service-bus-listener.js</string>
15     </array>
16     <key>RunAtLoad</key>
17     <false/>
18     <key>UserName</key>
19     <string>admin</string>
20     <key>WorkingDirectory</key>
21     <string>/Library/Management/CommitsListener/</string>
22     <key>StartInterval</key>
23     <integer>3600</integer>
24     <key>EnvironmentVariables</key>
25     <dict>
26         <key>HOME</key>
27         <string>/Users/admin</string>
28         <key>USER</key>
29         <string>admin</string>
30     </dict>
31 </dict>
32 </plist>
33
```

```
// service-bus-listener.js

const fs = require('fs');
const path = require('path');
const { exec, spawn } = require('child_process');
const util = require('util');
const { ServiceBusClient } = require('@azure/service-bus');
const os = require('os');

const execAsync = util.promisify(exec);

const connectionString = 'Endpoint=sb://commits-servicebus-
namespace.servicebus.windows.net/;SharedAccessKeyName=ListenOnlySharedAccessKey;SharedAccessKey=XXXXX';

const hostname = os.hostname().toLowerCase();
let topicName, subscriptionName;

if (hostname.includes('pluto')) {
    topicName = 'pluto-commits-servicebus-topic';
    subscriptionName = 'munki-repo-commits-subscription-pluto';
} else if (hostname.includes('proteus')) {
    topicName = 'proteus-commits-servicebus-topic';
    subscriptionName = 'munki-repo-commits-subscription-proteus';
} else {
    throw new Error('Unknown host. Unable to determine topic and subscription names.');
}

// Helper to format timestamps for logging.
function getFormattedTimestamp() {
    const now = new Date();
    const vancouverTime = new Date(now.toLocaleString('en-US', { timeZone: 'America/Vancouver' }));
    return `${vancouverTime.getFullYear()}-${(vancouverTime.getMonth() + 1).toString().padStart(2, '0')}-$
{vancouverTime.getDate().toString().padStart(2, '0')} ${vancouverTime.getHours().toString().padStart(2, '0')}:$
{vancouverTime.getMinutes().toString().padStart(2, '0')}:${vancouverTime.getSeconds().toString().padStart(2, '0')}
`;
}

// Setup log directories and streams.
const logDir = path.join(process.env.HOME, 'Library', 'Logs', 'CommitsListener');
if (!fs.existsSync(logDir)) {
    fs.mkdirSync(logDir, { recursive: true });
}
const logStream = fs.createWriteStream(path.join(logDir, 'service-bus-listener.log'), { flags: 'a' });
const errorLogStream = fs.createWriteStream(path.join(logDir, 'service-bus-listener_error.log'), { flags: 'a' });

// Override console.log and console.error to log to file.
console.log = (message) => logStream.write(`[${getFormattedTimestamp()}] ${message}\n`);
console.error = (message) => {
    // Filter out known non-critical errors.
    if (!message.includes("fatal: could not read Password")) {
        errorLogStream.write(`[${getFormattedTimestamp()}] ERROR: ${message}\n`);
    }
};

const serviceBusClient = new ServiceBusClient(connectionString);
const receiver = serviceBusClient.createReceiver(topicName, subscriptionName);

// Logs into Azure using the service principal and configures Git to use the access token.
async function getSystemPassword(service, account, keychainPath) {
    const command = `security find-generic-password -s ${service} -a ${account} -w ${keychainPath}`;
    try {
      const { stdout } = await execAsync(command);
      return stdout.trim();
    } catch (error) {
      console.error(`Error retrieving password from keychain: ${error.message}`);
      return null;
    }
  }

  async function azureLoginAndConfigureGit() {
    try {
      await execAsync(
        'security unlock-keychain -p "XXXX" /Library/Keychains/AzureCreds.keychain'
      );

      const servicePrincipalId = 'XXXXX';
      const tenantId = 'XXXXX';
      const keychainPath = '/Library/Keychains/AzureCreds.keychain';
      const service = 'azure-service-principal';

      // Now try to retrieve the password.
      const password = await getSystemPassword(service, servicePrincipalId, keychainPath);
      if (!password) {
        console.error('Azure service principal password not found in system keychain.');
        process.exit(1);
      }

      console.log('Attempting Azure login...');
      await execAsync(
        `/opt/homebrew/bin/az login --service-principal -u '${servicePrincipalId}' -p '${password}' -t '$
{tenantId}'`
      );

      const { stdout: accessToken } = await execAsync(
        "/opt/homebrew/bin/az account get-access-token --query accessToken -o tsv"
```

Commits Listener

Subscribes to a host-specific **Azure Service Bus topic** and waits **listening** for commit messages

Uses a **Entra Service Principal** token from macOS keychain to authenticate to Azure DevOps Git Repo using `vso.code_full`

Hard-resets `/Users/Shared/Munki`, runs `git pull`, then downloads `/pkgs`, `/icons`, and `/catalogs` from cloud bucket

Logs every step to `~/Library/Logs/CommitsLister`

```
// service-bus-listener.js

const fs = require('fs');
const path = require('path');
const { exec, spawn } = require('child_process');
const util = require('util');
const { ServiceBusClient } = require('@azure/service-bus');
const os = require('os');

const execAsync = util.promisify(exec);

const connectionString = 'Endpoint=sb://commits-servicebus-
namespace.servicebus.windows.net/;SharedAccessKeyName=ListenOnlySharedAccessKey;SharedAccessKey=XXXXX';

const hostname = os.hostname().toLowerCase();
let topicName, subscriptionName;

if (hostname.includes('pluto')) {
    topicName = 'pluto-commits-servicebus-topic';
    subscriptionName = 'munki-repo-commits-subscription-pluto';
} else if (hostname.includes('proteus')) {
    topicName = 'proteus-commits-servicebus-topic';
    subscriptionName = 'munki-repo-commits-subscription-proteus';
} else {
    throw new Error('Unknown host. Unable to determine topic and subscription names.');
}

// Helper to format timestamps for logging.
function getFormattedTimestamp() {
    const now = new Date();
    const vancouverTime = new Date(now.toLocaleString('en-US', { timeZone: 'America/Vancouver' }));
    return `${vancouverTime.getFullYear()}-${(vancouverTime.getMonth() + 1).toString().padStart(2, '0')}-$
{vancouverTime.getDate().toString().padStart(2, '0')} ${vancouverTime.getHours().toString().padStart(2, '0')}:$
{vancouverTime.getMinutes().toString().padStart(2, '0')}:${vancouverTime.getSeconds().toString().padStart(2, '0')}
`;
}

// Setup log directories and streams.
const logDir = path.join(process.env.HOME, 'Library', 'Logs', 'CommitsListener');
if (!fs.existsSync(logDir)) {
    fs.mkdirSync(logDir, { recursive: true });
}
const logStream = fs.createWriteStream(path.join(logDir, 'service-bus-listener.log'), { flags: 'a' });
const errorLogStream = fs.createWriteStream(path.join(logDir, 'service-bus-listener_error.log'), { flags: 'a' });

// Override console.log and console.error to log to file.
console.log = (message) => logStream.write(`[${getFormattedTimestamp()}] ${message}\n`);
console.error = (message) => {
    // Filter out known non-critical errors.
    if (!message.includes("fatal: could not read Password")) {
        errorLogStream.write(`[${getFormattedTimestamp()}] ERROR: ${message}\n`);
    }
};

const serviceBusClient = new ServiceBusClient(connectionString);
const receiver = serviceBusClient.createReceiver(topicName, subscriptionName);

// Logs into Azure using the service principal and configures Git to use the access token.
async function getSystemPassword(service, account, keychainPath) {
    const command = `security find-generic-password -s ${service} -a ${account} -w ${keychainPath}`;
    try {
      const { stdout } = await execAsync(command);
      return stdout.trim();
    } catch (error) {
      console.error(`Error retrieving password from keychain: ${error.message}`);
      return null;
    }
  }

  async function azureLoginAndConfigureGit() {
    try {
      await execAsync(
        'security unlock-keychain -p "XXXX" /Library/Keychains/AzureCreds.keychain'
      );

      const servicePrincipalId = 'XXXXX';
      const tenantId = 'XXXXX';
      const keychainPath = '/Library/Keychains/AzureCreds.keychain';
      const service = 'azure-service-principal';

      // Now try to retrieve the password.
      const password = await getSystemPassword(service, servicePrincipalId, keychainPath);
      if (!password) {
        console.error('Azure service principal password not found in system keychain.');
        process.exit(1);
      }

      console.log('Attempting Azure login...');
      await execAsync(
        `/opt/homebrew/bin/az login --service-principal -u '${servicePrincipalId}' -p '${password}' -t '$
{tenantId}'`
      );

      const { stdout: accessToken } = await execAsync(
        "/opt/homebrew/bin/az account get-access-token --query accessToken -o tsv"
```

Commits Listener

Azure Service Bus 🚌

Amazon SQS

Samples: https://github.com/rodchristiansen/munki-devops/tree/main/local-caching

admin — tail -f ~/Library/Logs/CommitsListener/service-bus-listener.log — 239×68

```
[2025-06-07 10:52:16] Received a message
[2025-06-07 10:52:16] Cleaning up temporary files...
[2025-06-07 10:52:16] Temporary files cleaned up successfully.
[2025-06-07 10:52:16] git output: * main

[2025-06-07 10:52:17] git output: On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

[2025-06-07 10:52:17] git output: HEAD is now at d063c382f Migrated to plain text .applescript + `osacompile`

[2025-06-07 10:52:18] Detected stale credentials. Attempting to reconfigure Git credentials...
[2025-06-07 10:52:18] Attempting Azure login...
[2025-06-07 10:52:24] Configuring Git with new access token...
[2025-06-07 10:52:24] Azure login and Git configuration completed successfully
[2025-06-07 10:52:26] git info: From https://dev.azure.com/                    /Devices/_git/Munki
   d063c382f..964beced8  main       -> origin/main
Auto packing the repository in background for optimum performance.
See "git help gc" for manual housekeeping.

[2025-06-07 10:52:26] git output: Updating d063c382f..964beced8
Fast-forward
 .githooks/post-merge                                | 138 ++++++++++++++--------
 .githooks/pre-commit                                |  70 ----------
 .githooks/pre-push                                  | 129 ++++++++++++-------
 .githooks/setup-hooks                               |  69 ----------
 .../mgmt/MacAdminsGitHooks-2025.06.07.plist         |  72 ++++++++++
 packages/MacAdminsGitHooks/build-info.json          |   4 +-
 .../MacAdminsGitHooks/payload/MacAdminsGitHooks.sh  |  27 ++++
 packages/MacAdminsGitHooks/scripts/postinstall      |   5 +
 packages/MacAdminsGitHooks/scripts/preinstall       |  14 +++
 9 files changed, 279 insertions(+), 249 deletions(-)
 delete mode 100755 .githooks/pre-commit
 delete mode 100755 .githooks/setup-hooks
 create mode 100644 deployment/pkgsinfo/mgmt/MacAdminsGitHooks-2025.06.07.plist
 create mode 100755 packages/MacAdminsGitHooks/payload/MacAdminsGitHooks.sh
 create mode 100755 packages/MacAdminsGitHooks/scripts/postinstall
 delete mode 100755 packages/MacAdminsGitHooks/scripts/preinstall

[2025-06-07 10:52:26] git info: Auto packing the repository in background for optimum performance.
See "git help gc" for manual housekeeping.
Auto packing the repository in background for optimum performance.
See "git help gc" for manual housekeeping.
```

```
Auto packing the repository in background for optimum performance.
See "git help gc" for manual housekeeping.

[2025-06-07 10:52:26] git output: Updating d063c382f..964beced8
Fast-forward
 .githooks/post-merge                                | 138 ++++++++++++++--------
 .githooks/pre-commit                                |  70 ----------
 .githooks/pre-push                                  | 129 ++++++++++++-------
 .githooks/setup-hooks                               |  69 ----------
 .../mgmt/MacAdminsGitHooks-2025.06.07.plist         |  72 ++++++++++
 packages/MacAdminsGitHooks/build-info.json          |   4 +-
 .../MacAdminsGitHooks/payload/MacAdminsGitHooks.sh  |  27 ++++
 packages/MacAdminsGitHooks/scripts/postinstall      |   5 +
 packages/MacAdminsGitHooks/scripts/preinstall       |  14 +--
 9 files changed, 279 insertions(+), 249 deletions(-)
 delete mode 100755 .githooks/pre-commit
 delete mode 100755 .githooks/setup-hooks
 create mode 100644 deployment/pkgsinfo/mgmt/MacAdminsGitHooks-2025.06.07.plist
 create mode 100755 packages/MacAdminsGitHooks/payload/MacAdminsGitHooks.sh
 create mode 100755 packages/MacAdminsGitHooks/scripts/postinstall
 delete mode 100755 packages/MacAdminsGitHooks/scripts/preinstall

[2025-06-07 10:52:26] git info: Auto packing the repository in background for optimum performance.
See "git help gc" for manual housekeeping.
Auto packing the repository in background for optimum performance.
See "git help gc" for manual housekeeping.
hint: The '.git/hooks/post-merge' hook was ignored because it's not set as executable.
hint: You can disable this warning with `git config advice.ignoredHook false`.

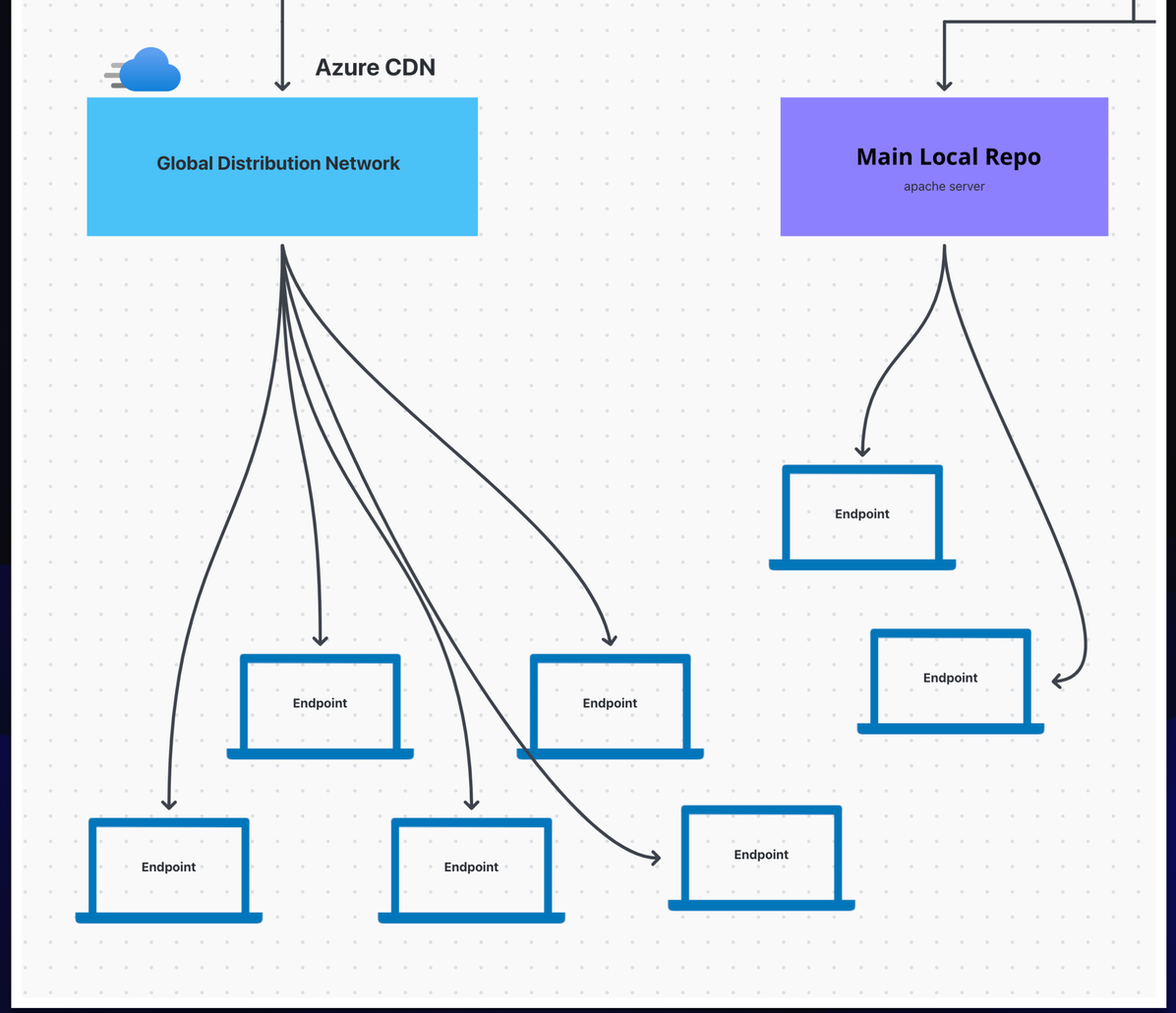
[2025-06-07 10:52:26] git output: commit 964beced8ed9171b9a040f25cb3e9390dc4de8c2
Author: Rod Christiansen <rchristiansen@ecuad.ca>
Date:   Sat Jun 7 10:49:35 2025 -0700

    Slight tweak on removing old hooks, pushing pkg

[2025-06-07 10:52:26] Starting S3 sync for packages...
download:                                          repo/deployment/pkgs/mgmt/MacAdminsGitHooks-2025.06.07.pkg to deployment/pkgs/mgmt/MacAdminsGitHooks-2025.06.07.pkg


[2025-06-07 10:52:48] S3 sync completed for packages.
[2025-06-07 10:52:48] Starting S3 sync for icons...
[2025-06-07 10:52:49] S3 sync completed for icons.
[2025-06-07 10:52:49] Starting S3 sync for catalogs...
download:                                         'repo/deployment/catalogs/Development to deployment/catalogs/Development
download:                                         'repo/deployment/catalogs/Production to deployment/catalogs/Production
download:                                         'repo/deployment/catalogs/all to deployment/catalogs/all

[2025-06-07 10:52:51] S3 sync completed for catalogs.
[2025-06-07 10:52:51] Message processing completed successfully
```



Dynamically switching between on/off prem with munki preflight

Sends a lightweight HTTP HEAD to URLs

If the mirror answers 200 OK within ~3 s, it is picked

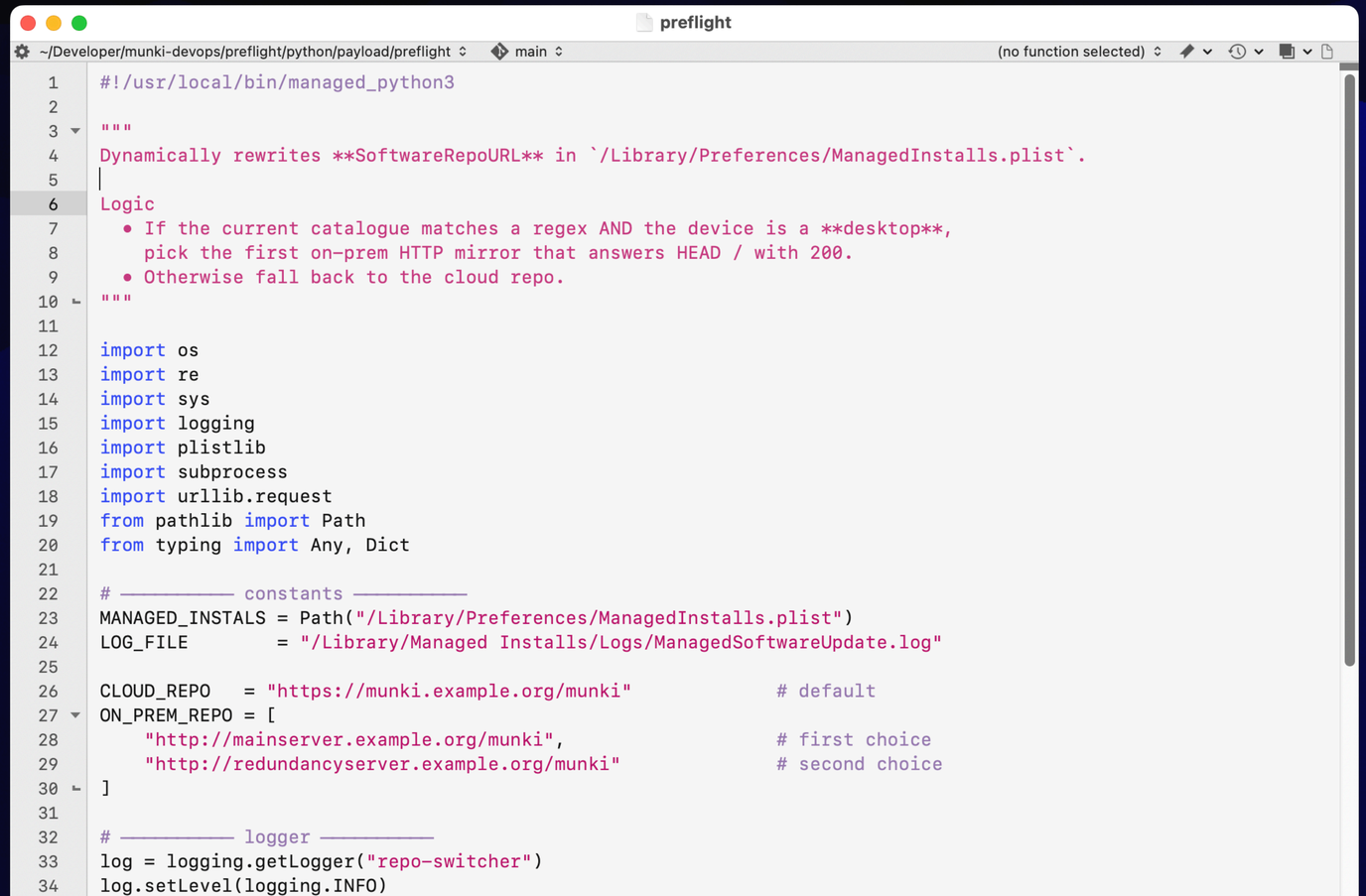
Result is written to SoftwareRepoURL and logged, then the script exits

```
rod@RodChristiansen ~ % sudo managedsoftwareupdate -v
Managed Software Update Tool
Version 6.6.3.4704
Copyright 2010-2024 The Munki Project
https://github.com/munki/munki

Starting...
    Performing preflight tasks...
    preflight stdout:
Staff/IT/RodChristiansen manifest exists, skipping API enrollment.
Set SoftwareRepoURL to Munki cloud.
Connected to https://munki.ecuad.ca/deployment
```

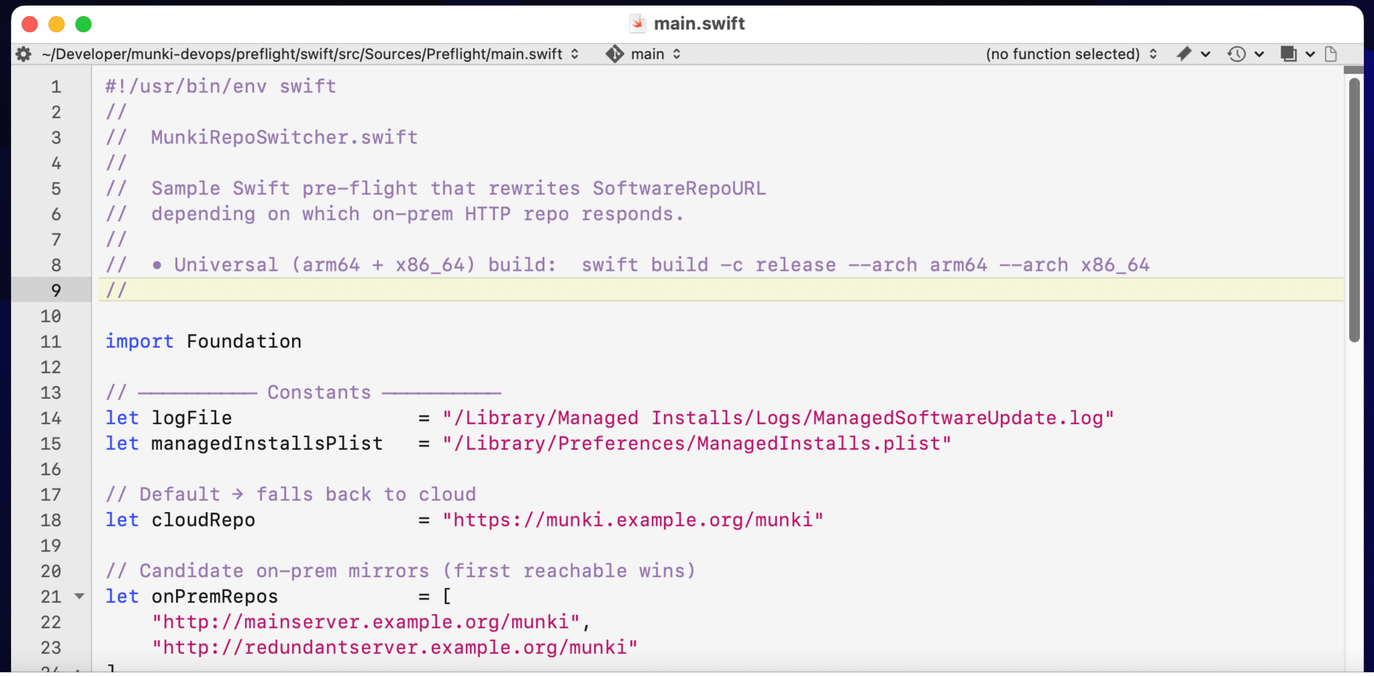
Preflight pkg samples

Python preflight sample:
https://github.com/rodchristiansen/munki-devops/blob/main/preflight/python/payload/preflight

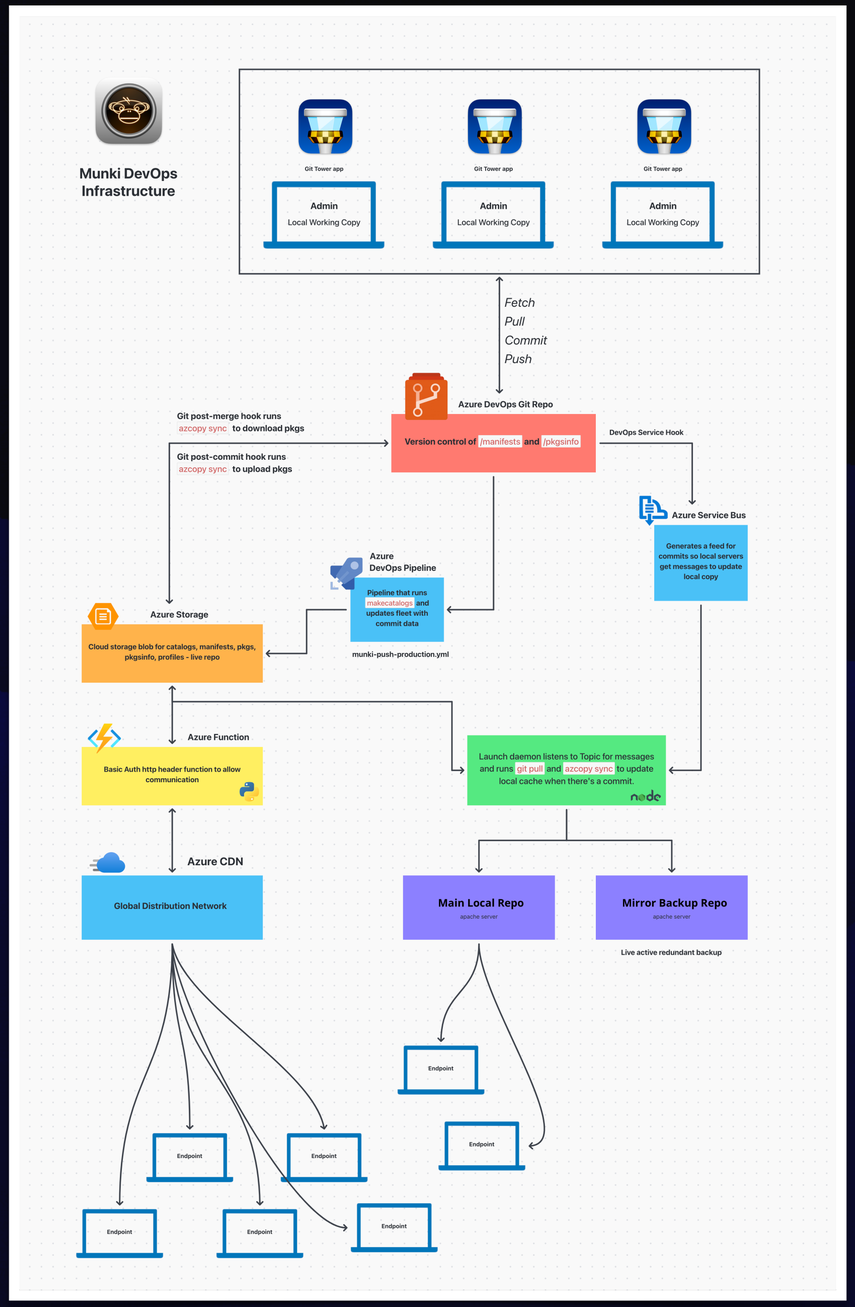


Preflight pkg samples

Swift preflight sample:
https://github.com/rodchristiansen/munki-devops/blob/main/preflight/swift/src/Sources/Preflight/main.swift







Fully DevOps Munki

Admins local working copies

Git Repos

Git Hooks

CI/CD Pipeline

Cloud serverless infra

Updating local cache

Dynamic Repo URL

devicesMgmtOS



Inventory



DevOps



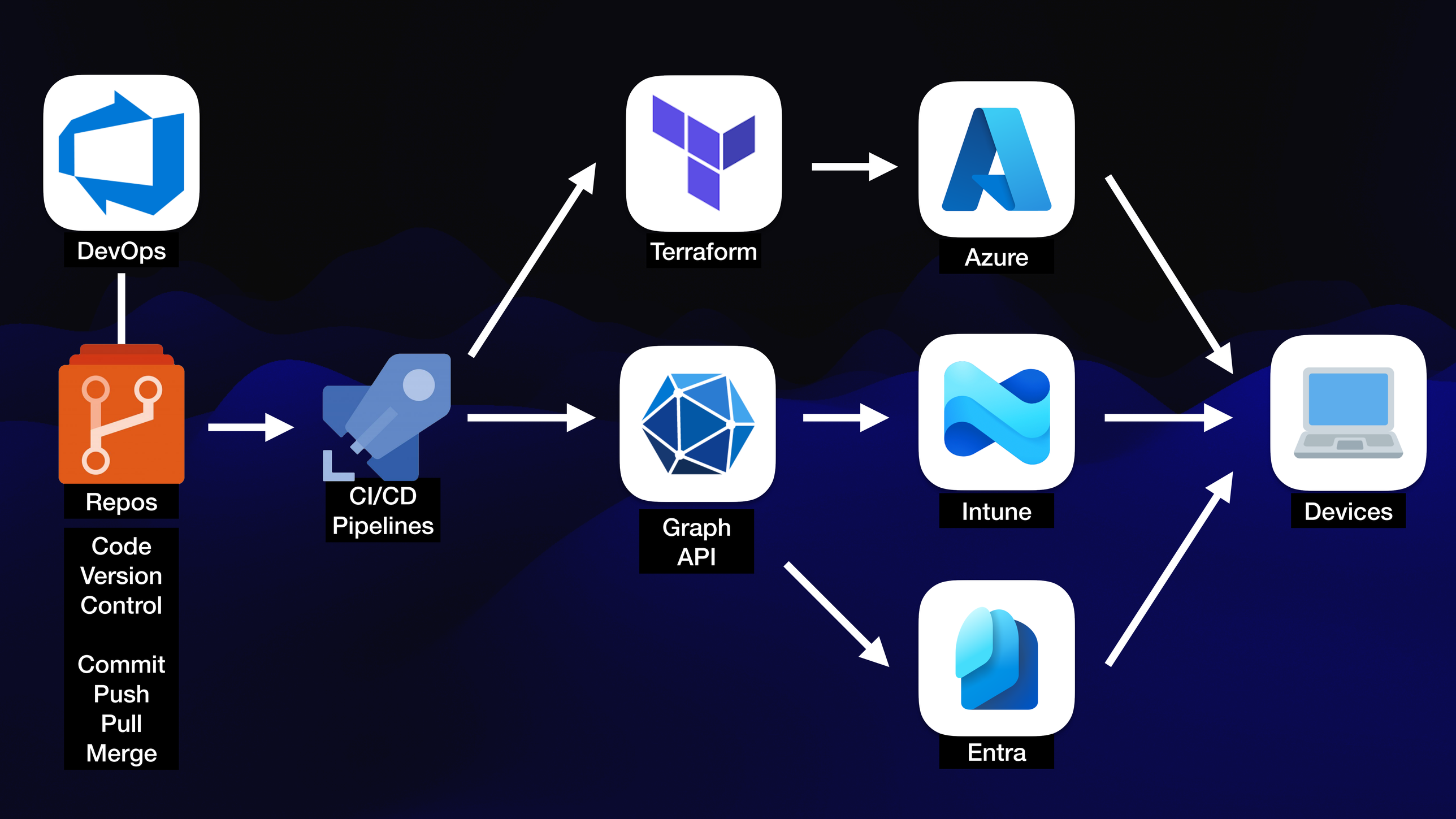
Intune

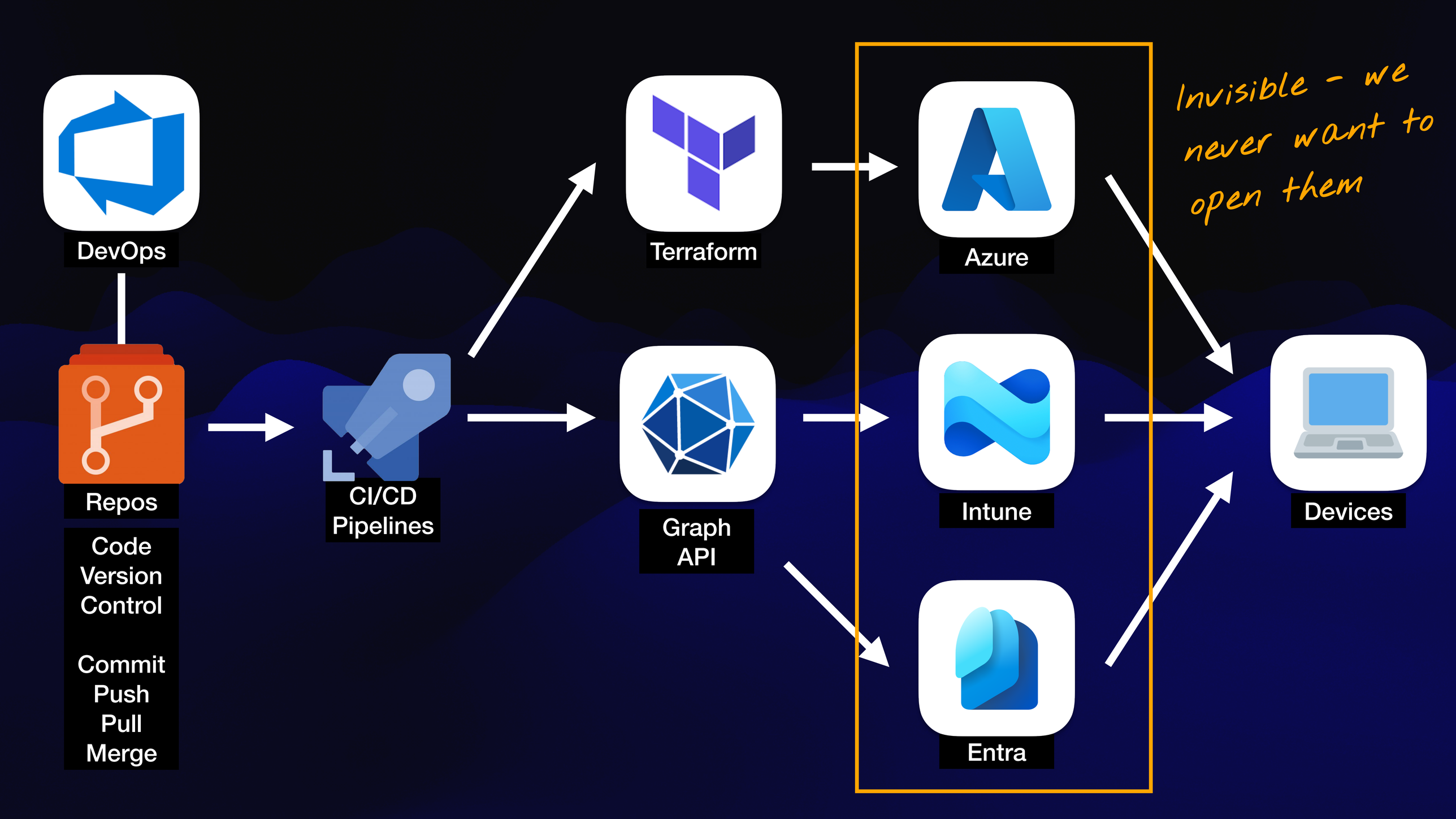


Entra



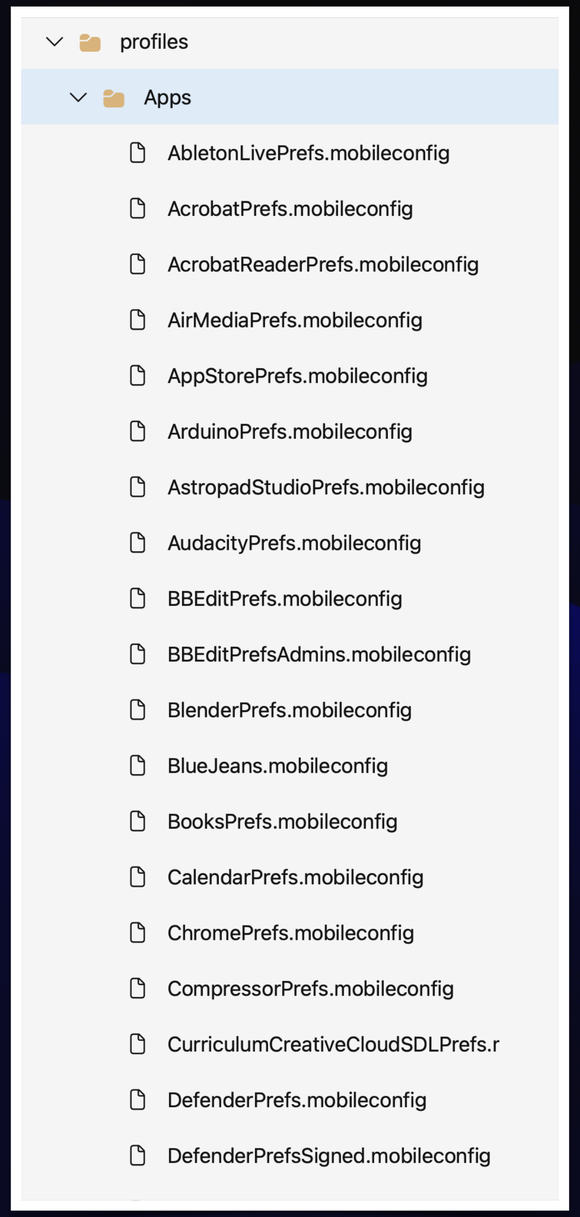
Azure

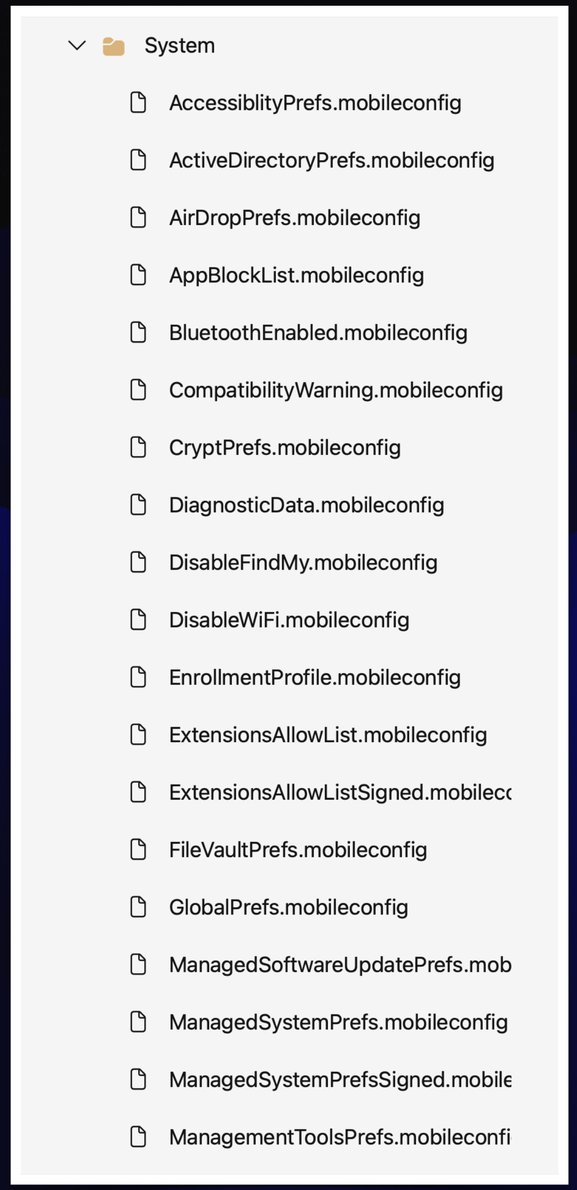

DevOps
Repos
Code
Version
Control
Commit
Push
Pull
Merge
CI/CD
Pipelines
Terraform
Graph
API
Azure
Intune
Entra
Devices


DevOps
Repos
Code
Version
Control
Commit
Push
Pull
Merge
CI/CD
Pipelines
Terraform
Graph
API
Azure
Intune
Entra
Devices
Invisible - we never want to open them

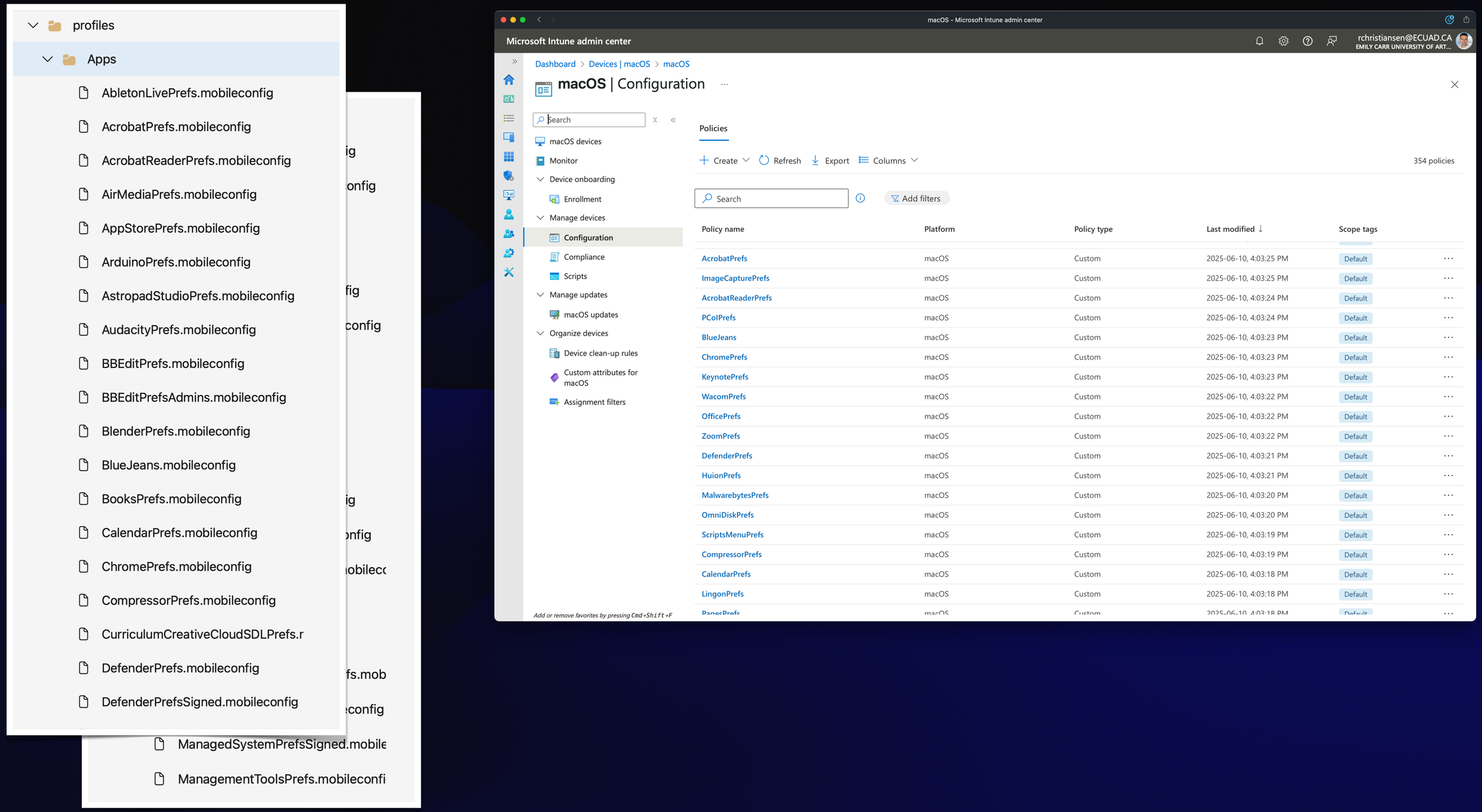


Git Repo Orchestrating



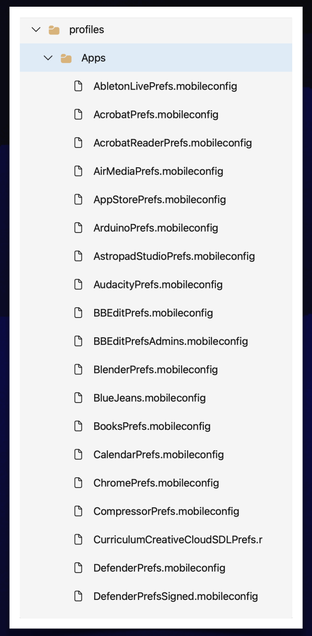


An example for Mobile configs


profiles
Apps
AbletonLivePrefs.mobileconfig
AcrobatPrefs.mobileconfig
AcrobatReaderPrefs.mobileconfig
AirMediaPrefs.mobileconfig
AppStorePrefs.mobileconfig
ArduinoPrefs.mobileconfig
AstropadStudioPrefs.mobileconfig
AudacityPrefs.mobileconfig
BBEditPrefs.mobileconfig
BBEditPrefsAdmins.mobileconfig
BlenderPrefs.mobileconfig
BlueJeans.mobileconfig
BooksPrefs.mobileconfig
CalendarPrefs.mobileconfig
ChromePrefs.mobileconfig
CompressorPrefs.mobileconfig
CurriculumCreativeCloudSDLPrefs.r
DefenderPrefs.mobileconfig
DefenderPrefsSigned.mobileconfig
ManagedSystemPrefsSigned.mobile
ManagementToolsPrefs.mobileconfi
macOS - Microsoft Intune admin center
Microsoft Intune admin center
rchristiansen@ECUAD.CA
EMILY CARR UNIVERSITY OF ART...
Dashboard > Devices | macOS > macOS
macOS | Configuration
Search
macOS devices
Monitor
Device onboarding
Enrollment
Manage devices
Configuration
Compliance
Scripts
Manage updates
macOS updates
Organize devices
Device clean-up rules
Custom attributes for macOS
Assignment filters
Add or remove favorites by pressing Cmd+Shift+F
Policies
Create
Refresh
Export
Columns
354 policies
Search
Add filters
Policy name
Platform
Policy type
Last modified
Scope tags
AcrobatPrefs macOS Custom 2025-06-10, 4:03:25 PM Default
ImageCapturePrefs macOS Custom 2025-06-10, 4:03:25 PM Default
AcrobatReaderPrefs macOS Custom 2025-06-10, 4:03:24 PM Default
PCoIPrefs macOS Custom 2025-06-10, 4:03:24 PM Default
BlueJeans macOS Custom 2025-06-10, 4:03:23 PM Default
ChromePrefs macOS Custom 2025-06-10, 4:03:23 PM Default
KeynotePrefs macOS Custom 2025-06-10, 4:03:23 PM Default
WacomPrefs macOS Custom 2025-06-10, 4:03:22 PM Default
OfficePrefs macOS Custom 2025-06-10, 4:03:22 PM Default
ZoomPrefs macOS Custom 2025-06-10, 4:03:22 PM Default
DefenderPrefs macOS Custom 2025-06-10, 4:03:21 PM Default
HuionPrefs macOS Custom 2025-06-10, 4:03:21 PM Default
MalwarebytesPrefs macOS Custom 2025-06-10, 4:03:20 PM Default
OmniDiskPrefs macOS Custom 2025-06-10, 4:03:20 PM Default
ScriptsMenuPrefs macOS Custom 2025-06-10, 4:03:19 PM Default
CompressorPrefs macOS Custom 2025-06-10, 4:03:19 PM Default
CalendarPrefs macOS Custom 2025-06-10, 4:03:18 PM Default
LingonPrefs macOS Custom 2025-06-10, 4:03:18 PM Default

Munki manifest `managed_profiles` array

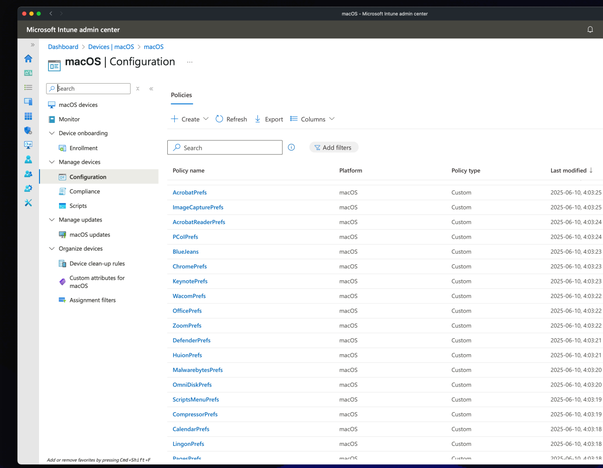
Munki Manifest = Entra assignment group



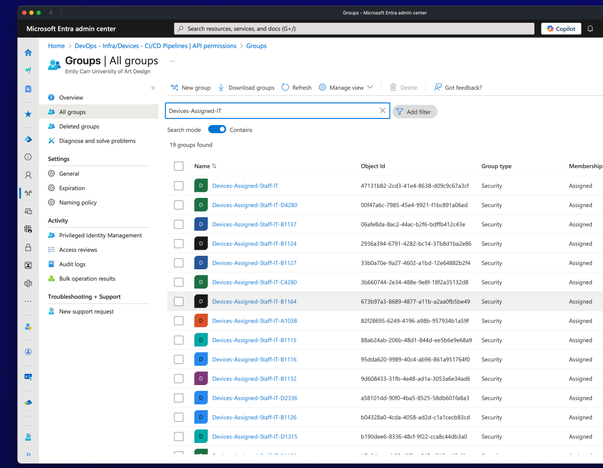
CI/CD Pipelines

Graph API

Intune

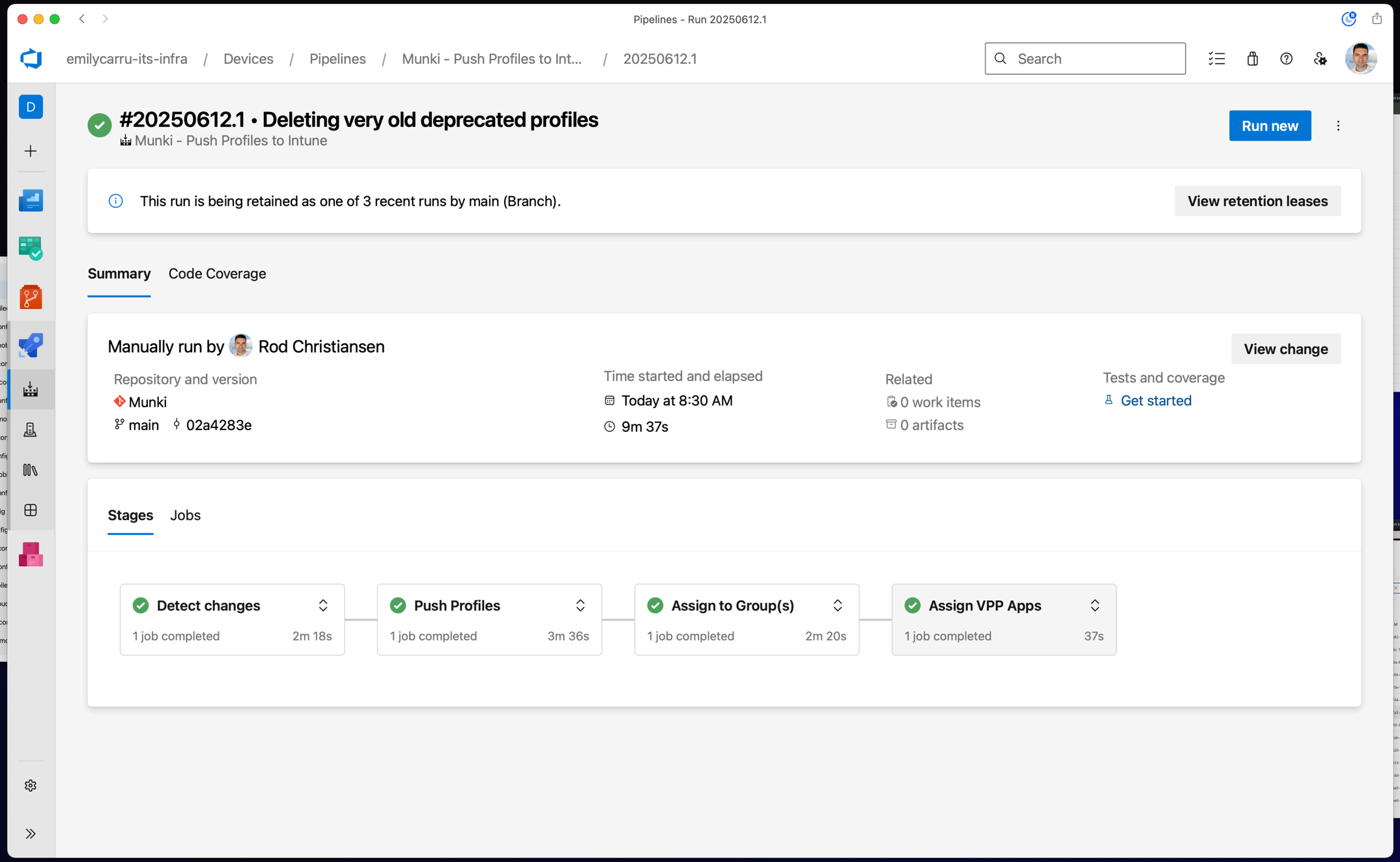


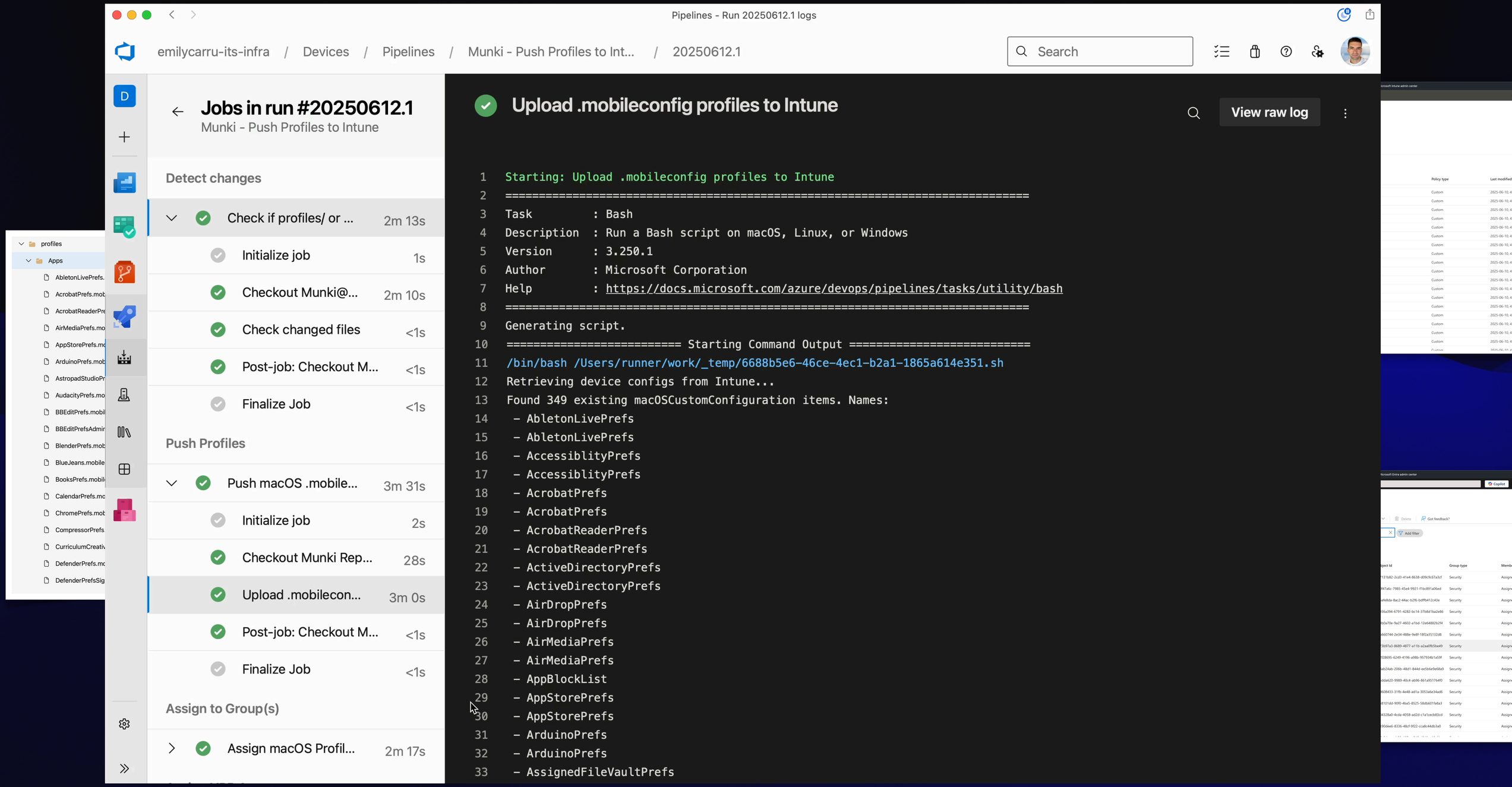
Entra



Manifest: [Assigned/Staff/IT/B1115/RodChristiansen]

Entra group: [Devices-Assigned-Staff-IT-B1115]


Pipelines - Run 20250612.1
emilycarru-its-infra / Devices / Pipelines / Munki - Push Profiles to Int... / 20250612.1
Search
#20250612.1 • Deleting very old deprecated profiles
Munki - Push Profiles to Intune
Run new
This run is being retained as one of 3 recent runs by main (Branch).
View retention leases
Summary
Code Coverage
Manually run by Rod Christiansen
View change
Repository and version
Munki
main
02a4283e
Time started and elapsed
Today at 8:30 AM
9m 37s
Related
0 work items
0 artifacts
Tests and coverage
Get started
Stages
Jobs
Detect changes
1 job completed
2m 18s
Push Profiles
1 job completed
3m 36s
Assign to Group(s)
1 job completed
2m 20s
Assign VPP Apps
1 job completed
37s

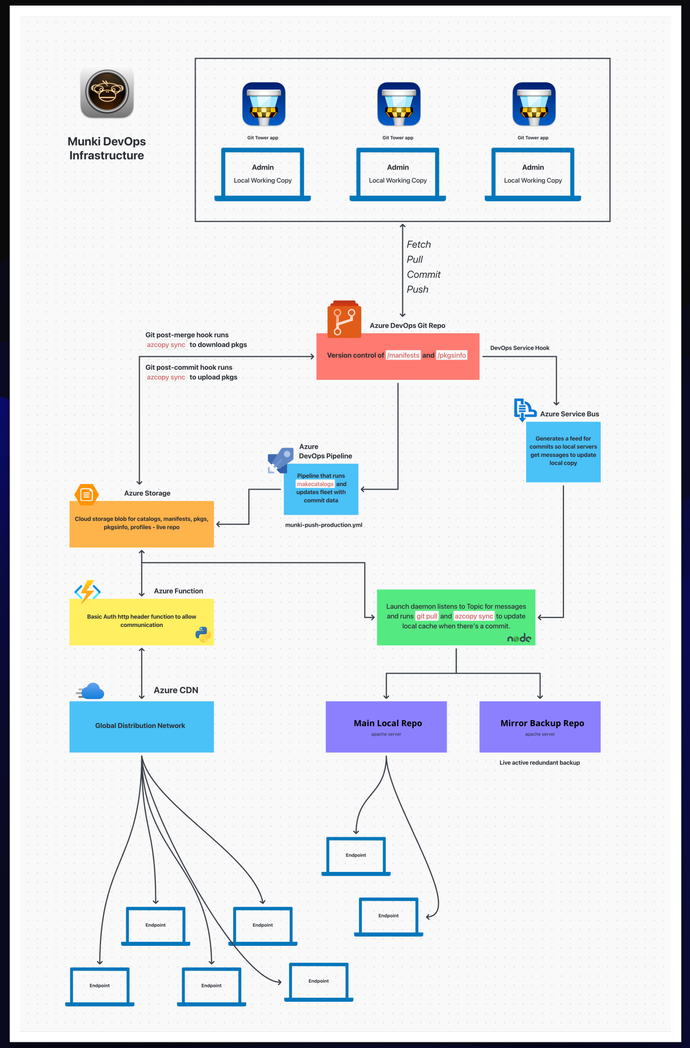

Pipelines - Run 20250612.1 logs
emilycarru-its-infra / Devices / Pipelines / Munki - Push Profiles to Int... / 20250612.1
Search
Jobs in run #20250612.1
Munki - Push Profiles to Intune
Detect changes
Check if profiles/ or ... 2m 13s
Initialize job 1s
Checkout Munki@... 2m 10s
Check changed files <1s
Post-job: Checkout M... <1s
Finalize Job <1s
Push Profiles
Push macOS .mobile... 3m 31s
Initialize job 2s
Checkout Munki Rep... 28s
Upload .mobilecon... 3m 0s
Post-job: Checkout M... <1s
Finalize Job <1s
Assign to Group(s)
Assign macOS Profil... 2m 17s
Upload .mobileconfig profiles to Intune
View raw log
1 Starting: Upload .mobileconfig profiles to Intune
2 ==
3 Task : Bash
4 Description : Run a Bash script on macOS, Linux, or Windows
5 Version : 3.250.1
6 Author : Microsoft Corporation
7 Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/bash
8 ==
9 Generating script.
10 ========================== Starting Command Output ===========================
11 /bin/bash /Users/runner/work/_temp/6688b5e6-46ce-4ec1-b2a1-1865a614e351.sh
12 Retrieving device configs from Intune...
13 Found 349 existing macOSCustomConfiguration items. Names:
14 - AbletonLivePrefs
15 - AbletonLivePrefs
16 - AccessiblityPrefs
17 - AccessiblityPrefs
18 - AcrobatPrefs
19 - AcrobatPrefs
20 - AcrobatReaderPrefs
21 - AcrobatReaderPrefs
22 - ActiveDirectoryPrefs
23 - ActiveDirectoryPrefs
24 - AirDropPrefs
25 - AirDropPrefs
26 - AirMediaPrefs
27 - AirMediaPrefs
28 - AppBlockList
29 - AppStorePrefs
30 - AppStorePrefs
31 - ArduinoPrefs
32 - ArduinoPrefs
33 - AssignedFileVaultPrefs


DevOps
Repos
Code
Version
Control
Commit
Push
Pull
Merge
Terraform
Azure
CI/CD
Pipelines
Graph API
Intune
Entra
Devices
Git Repo
Orchestrating
one more thing

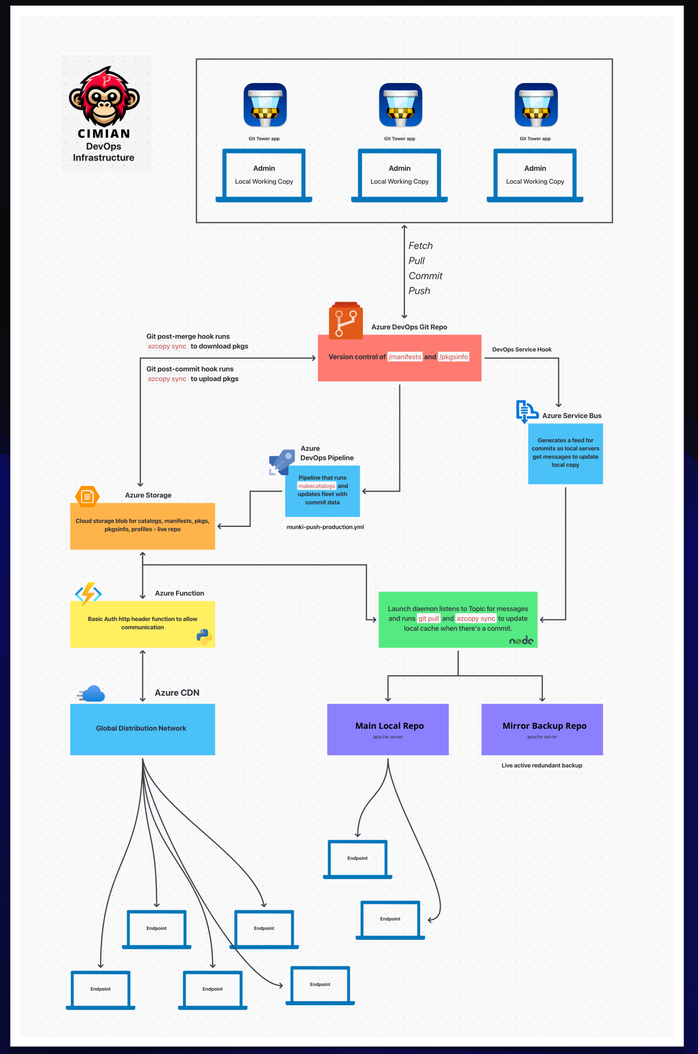


Git Repo Orchestrating

Macs



Windows





Macs

Cimian is a DevOps Software Deployment System for Windows that can install MSI(X), EXE, NUPKG, and run PWSH scripts.



Windows



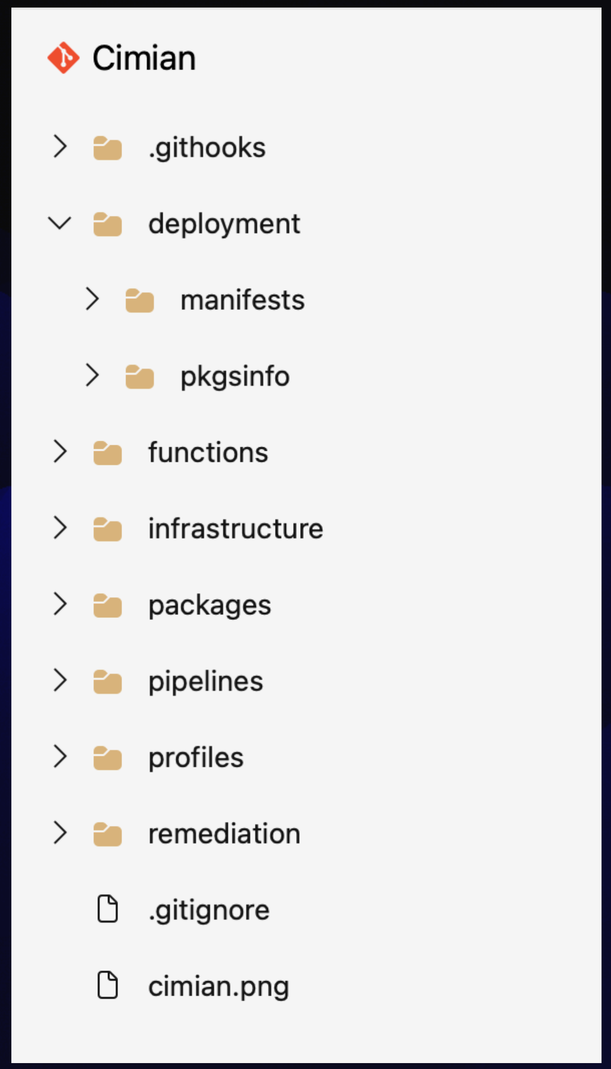
Macs

```
app_usage_monitor
makecatalogs
makepkginfo
managedsoftwareupdate
manifestutil
munkiimport
munkipkg
```

```
app_usage_monitor.exe
makecatalogs.exe
makepkginfo.exe
managedsoftwareupdate.exe
manifestutil.exe
cimiimport.exe
cimiipkg.exe
```



Windows



app_usage_monitor.exe

makecatalogs.exe

makepkginfo.exe

managedsoftwareupdate.exe

manifestutil.exe

cimiimport.exe

cimiipkg.exe

Windows

Animation.yaml

~/DevOps/Cimian/deployment/manifests/Shared/Curriculum/Animation.yaml main (functions)

```yaml
name: Animation
catalogs:
managed_installs:
- AutoDeskMaya2024
- Anaconda
- Bifrost
- Blender
- CLO
- Resolve
- DeadlineClient
- DragonFrame
- DuikAngela
- EpicGamesLauncher
- Houdini
- Katana
- KatanaToArnold
- KeyFramePro
- Keyshot
- Krita
- Mari
- MarvelousDesigner
- MaxonApp
- MayaBonusTools
- MayaComponents
- MayaReset
- MeshRoom
- Modo
- MtoA
- ngSkinTools
- Nuke
- Processing
- Redshift
- SideFXLabs
- StoryboardPro
- Tinkerine
- UnityComponents
- UnityHub
- UnrealEngine
- UnrealEnginePreReq
- Wacom
- ZBrush
managed_profiles:
- HideShutDown
- DisableCopilot
- DisableFirewall
- DisablePinningStore
- HideTaskViewButton
- SharedChromePrefs
- SharedDefaultBrowser
- SharedEdgePrefs
- SharedFirefoxPrefs
- SharedGlobalPrefs
managed_uninstalls:
managed_updates:
optional_installs:
included_manifests:
- Shared\Curriculum.yaml

```

app_usage_monitor.exe

makecatalogs.exe

makepkginfo.exe

managedsoftwareupdate.exe

manifestutil.exe

cimiimport.exe

cimiipkg.exe



Windows

Mari-x64-7.0.2.yaml

~/DevOps/Cimian/deployment/pkgsinfo/apps/animation/Mari-x64-7.0.2.yaml main (functions)

```yaml
name: Mari
display_name: Mari
version: 7.0.2
description: Mari version 7.0.2 for ca.emilycarru.winadmins.Mari by EmilyCarrU
category: animation
developer: EmilyCarrU
catalogs:
  - Development
  - Testing
  - Staging
  - Production
installs:
  - type: 'file'
    path: 'C:\Program Files\Mari7.0v2\Bundle\bin\Mari7.0v2.exe'
    md5checksum: 'e6cf94e5c51bf86e9ba7629af42578fbba5a7ea4'
    version: '7.0.2'

supported_architectures:
  - x64
unattended_install: true
unattended_uninstall: true
installer:
  location: \apps\animation\Mari-x64-7.0.2.nupkg
  hash: e477d7200ac6dbfc24ce6dba01e17c52e5e83e12ab9ee9b89b0077da1e91d5de
  type: nupkg
  size: 1449937

postinstall_script: |-
  "C:\Program Files\Mari6.0v2\Uninstall.exe" /S
  dir /b /a-d > version.txt
  For /F "tokens=1,2 delims=i-" %%G in (version.txt) do set VERSION=%%H
  SET SHORTCUT="C:\Users\Public\Desktop\Mari*.lnk"
  ::Main Application
  Mari7.0v2-win-x86-release-64.exe /S /LOG="Mari.log"
  ::License
  copy /Y "foundry_client.lic"  "C:\ProgramData\The Foundry\RLM " foundry_client.lic
  ::Create Remove.bat
  echo "C:\Program Files\Mari%VERSION%\uninstall.exe" /S>>"%~dp0remove.bat"
  ::Remove Shortcuts
  if exist %SHORTCUT% del /f %SHORTCUT%
  del /f  "C:\ProgramData\Microsoft\Windows\Start Menu\Programs\The Foundry\Mari
%VERSION%\Mari %VERSION% (Non-commercial).lnk"
  del /S /Q "C:\Windows\Temp\mari"

uninstall_script: |-
  "C:\Program Files\Mari7.0v2\uninstall.exe" /S
```

app_usage_monitor.exe

makecatalogs.exe

makepkginfo.exe

managedsoftwareupdate.exe

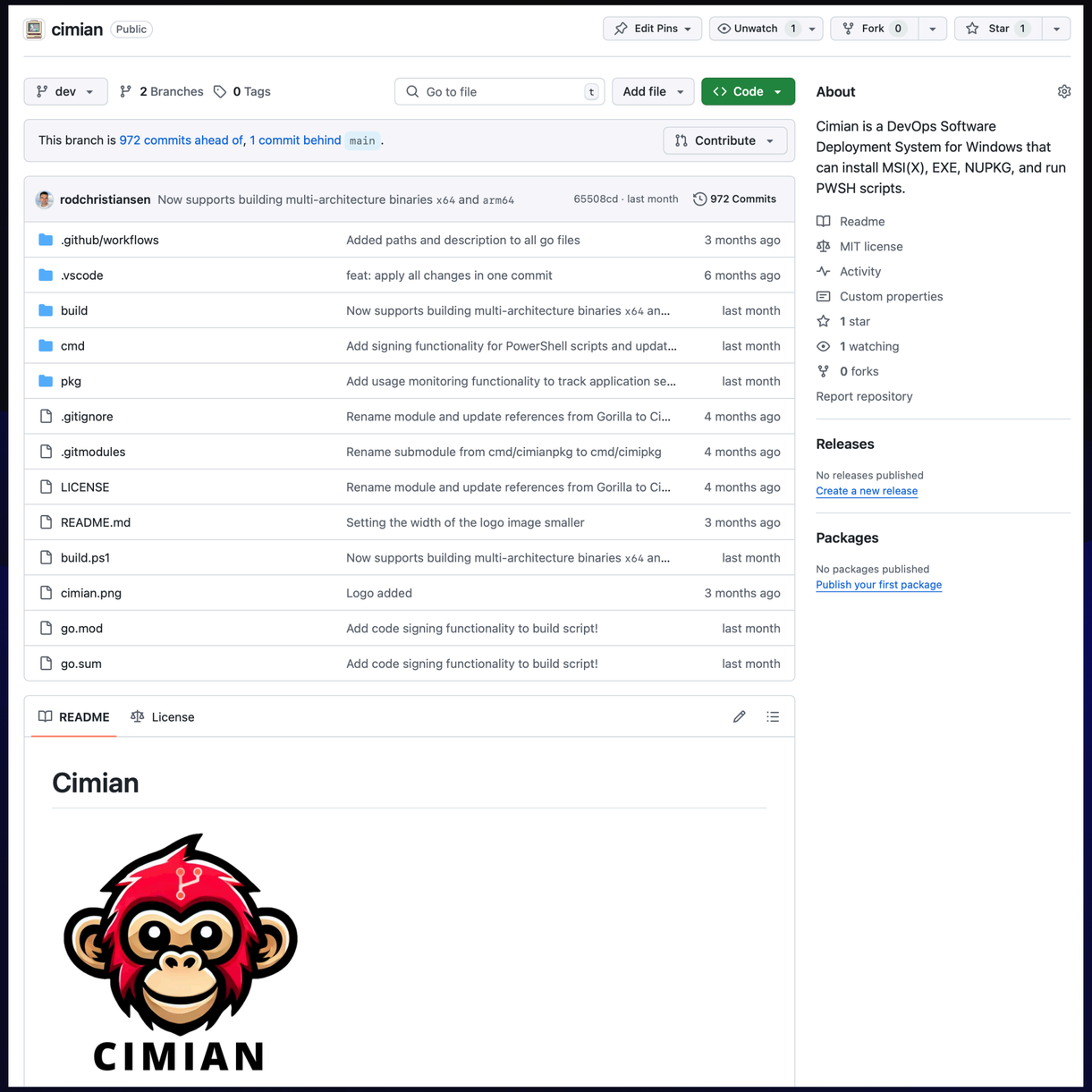
manifestutil.exe

cimiimport.exe

cimiipkg.exe



Windows



https://github.com/windowsadmins/cimian/tree/dev

```
app_usage_monitor.exe
makecatalogs.exe
makepkginfo.exe
managedsoftwareupdate.exe
manifestutil.exe
cimiimport.exe
cimiipkg.exe
```



Windows

cimipkg

`cimipkg` is a tool for building `.nupkg` packages for deploying software on Windows in a consistent, repeatable manner. It leverages **NuGet** for package creation and **Chocolatey** (or **Cimian**) for deployment, with support for **pre- and post-installation scripts** and **code signing**.

This tool simplifies the complexities of deployment by abstracting YAML-based configuration and script-based actions and offers **flexible certificate signing** using Windows `SignTool`.

Features

- **Dynamic File Inclusion**: Automatically packages all files and folders in the `payload/` directory.
- **Script Support**: Supports **pre-install** and **post-install** scripts executed with **elevated privileges**.
- **Custom Installation Paths**: Configure where the payload files are installed via the YAML configuration.
- **Post-Install Actions**: Supports automatic **logout** or **restart** after package installation.
- **Package Signing with `SignTool`**: Allows seamless signing of `.nupkg` packages using a `.pfx` certificate or certificate store.
- **Smart Readme Inclusion**: Automatically creates and includes a `readme.md` if a description is provided in `build-info.yaml`.
- **Automated Packaging**: Uses the `nuget` CLI to build `.nupkg` packages for deployment with **Chocolatey** or **Cimian**.
- **YAML-Driven Configuration**: All metadata and installation instructions come from the `build-info.yaml` file.

Folder Structure for Packages

```
project/
├── payload/                    # Files/folders to be written to disk
│   └── example.txt
├── scripts/                    # Pre-/Post-install scripts
│   ├── preinstall.ps1          # Runs before files are installed
│   └── postinstall.ps1         # Runs after files are installed
└── build-info.yaml             # Metadata for package generation
```

YAML Configuration: `build-info.yaml`

The `build-info.yaml` file contains configuration settings for the package:

```yaml
product:
  name: "Cimian"
  version: "2024.10.11"
  identifier: "com.cimiancorp.cimian"
  publisher: "Cimian Corp"
  description: "This is the StartSet installer package."
install_location: "C:\Program Files\Cimian"
postinstall_action: "none"
signing_certificate: "Cimian Corp EV Certificate"
```

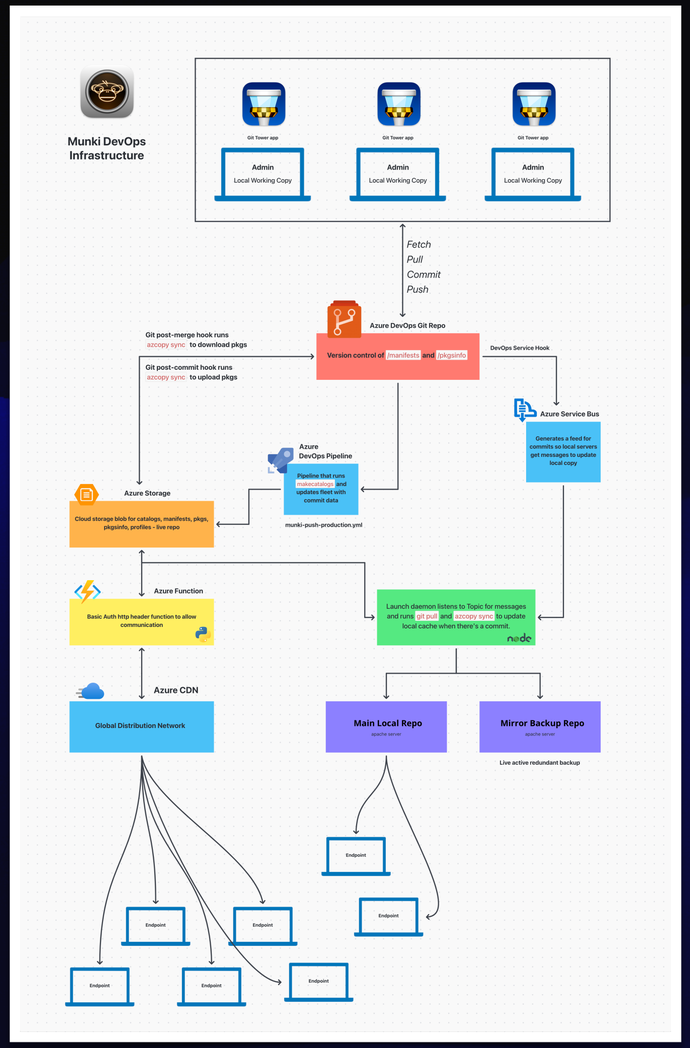
https://github.com/windowsadmins/cimian-pkg

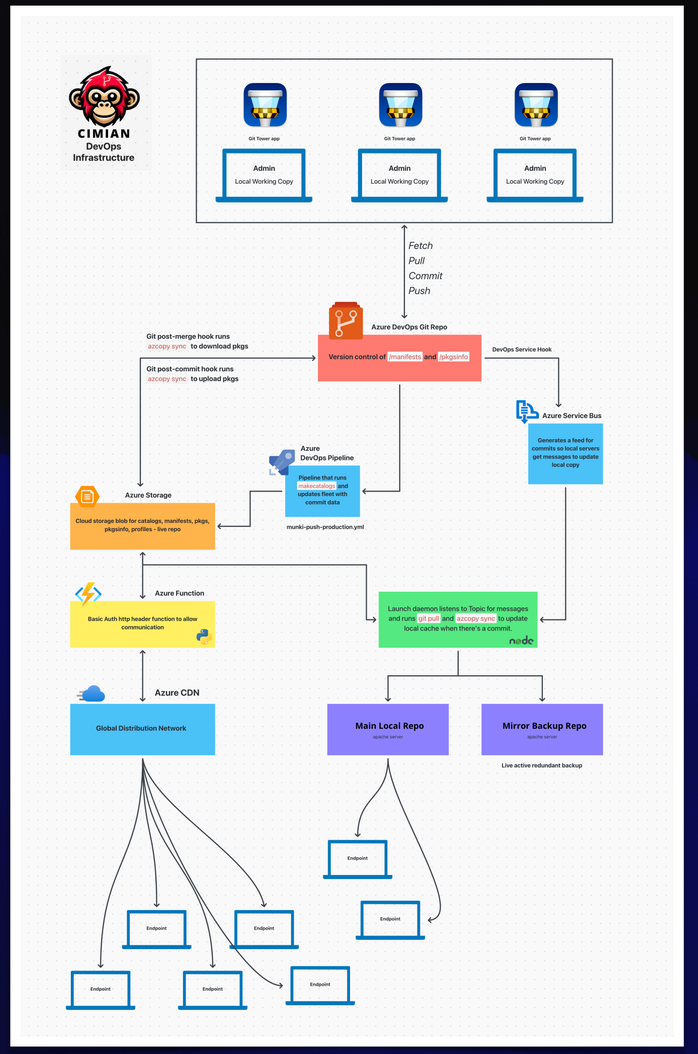




Windows

Macs

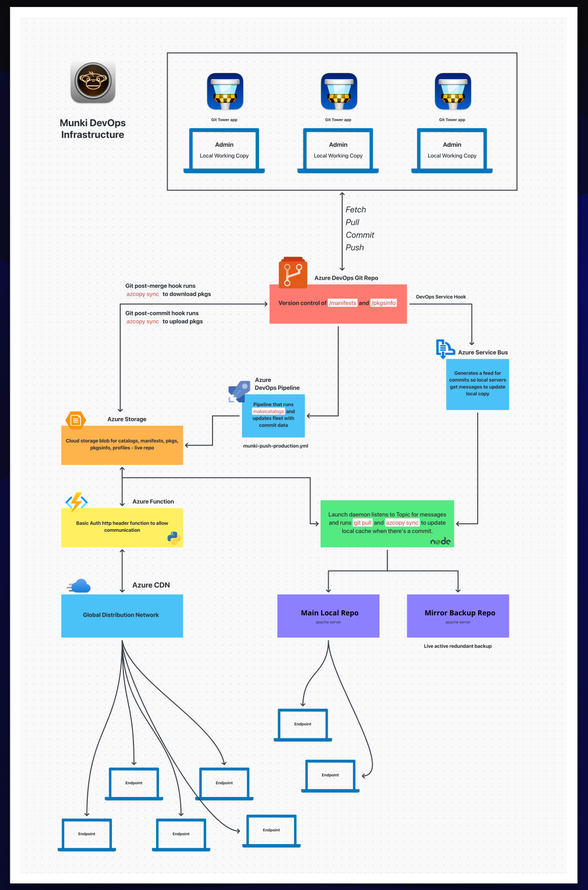




Next year!

Windows

Fully Running our Munki Repo Operations in DevOps




mauruuru
Kiitos
mahalo
Köszi
Merci
спасибо
Grazie
Thank you
Dank U
obrigado
Takk
Děkuju
Dziękuję
Gracias
danke