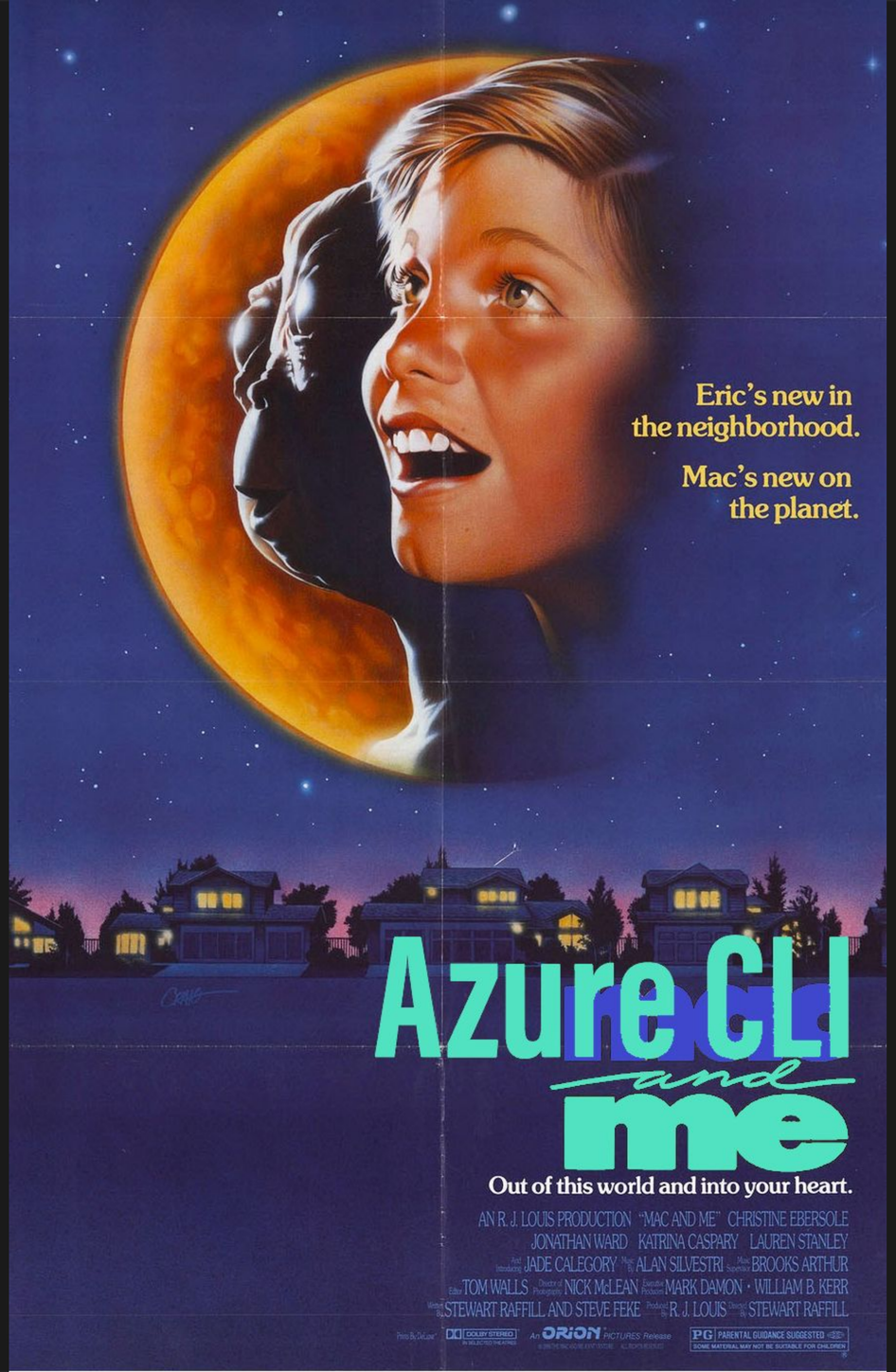
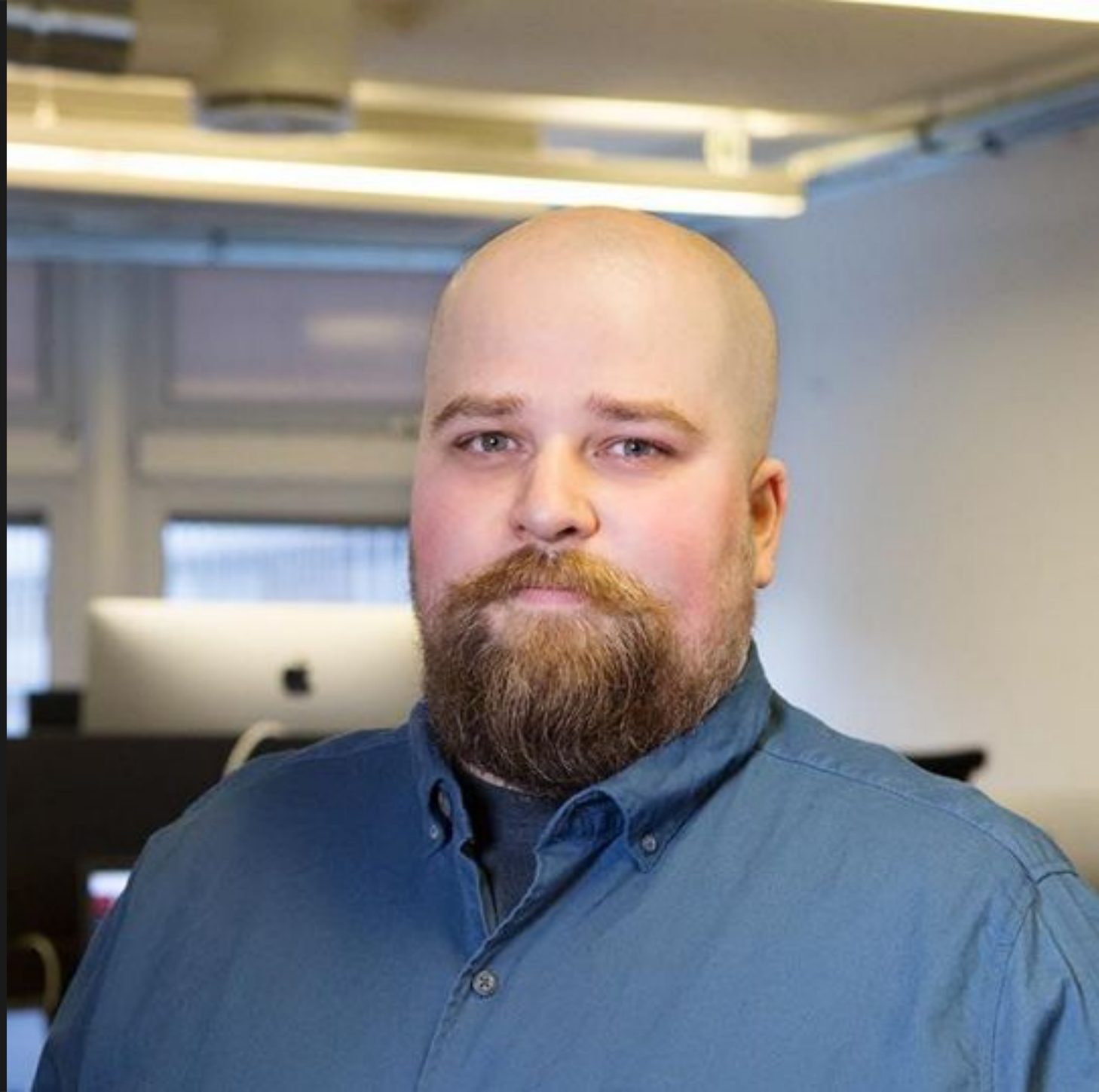
nova
net


Eric's new in the neighborhood.
Mac's new on the planet.
Azure CLI
and
me
Out of this world and into your heart.
AN R. J. LOUIS PRODUCTION "MAC AND ME" CHRISTINE EBERSOLE
JONATHAN WARD KATRINA CASPARY LAUREN STANLEY
JADE CALEGORY
ALAN SILVESTRI
BROOKS ARTHUR
TOM WALLS
NICK McLEAN
MARK DAMON • WILLIAM B. KERR
STEWART RAFFILL AND STEVE FEKE
R. J. LOUIS
STEWART RAFFILL
DOLBY STEREO
An ORION PICTURES Release
PG PARENTAL GUIDANCE SUGGESTED
SOME MATERIAL MAY NOT BE SUITABLE FOR CHILDREN

nova net

Johan Grønstad
Senior consultant / Partner

🐦‍⬛pr0nin

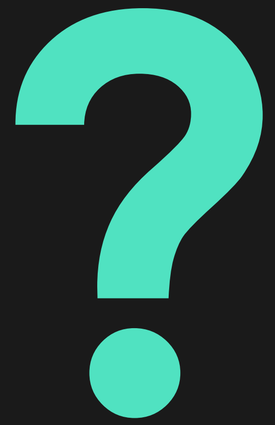


nova
net

nova
net

Azure CLI

nova
net



nova
net

Let's talk a little about the anatomy of az commands.

nova
net

az **group create**
--location westeurope
--name signus-rg

Let's talk a little about the anatomy of az commands.

nova
net

az **appservice plan create**
--resource-group signus-rg
--name signus-app-plan

Let's talk a little about the anatomy of az commands.

nova
net

az **webapp create**
--resource-group signus-rg
--plan signus-app-plan
--name signus

Let's talk a little about the anatomy of az commands.

nova
net

az *thing* verb *--params uuuu*

Let's talk a little about the anatomy of az commands.

nova
net

az *group* create *--name* nn *--location* westeurope

Let's talk a little about the anatomy of az commands.

nova
net

az *group* show *--name nn*

Let's talk a little about the anatomy of az commands.

nova
net

az *group* list

Let's talk a little about the anatomy of az commands.

nova net

Why buy Azure CLI?

Let's talk a little about the anatomy of az commands.

nova
net

$ az *resource* show

Let's talk a little about the anatomy of az commands.

nova
net

$ az *resource* show
  --resource-group signus

Let's talk a little about the anatomy of az commands.

nova
net

$ az *resource* show
  --resource-group signus
  --name appinsight

Let's talk a little about the anatomy of az commands.

nova
net

```
$ az resource show
  --resource-group signus
  --name appinsight
  --resource-type microsoft.insights/component
```

Let's talk a little about the anatomy of az commands.

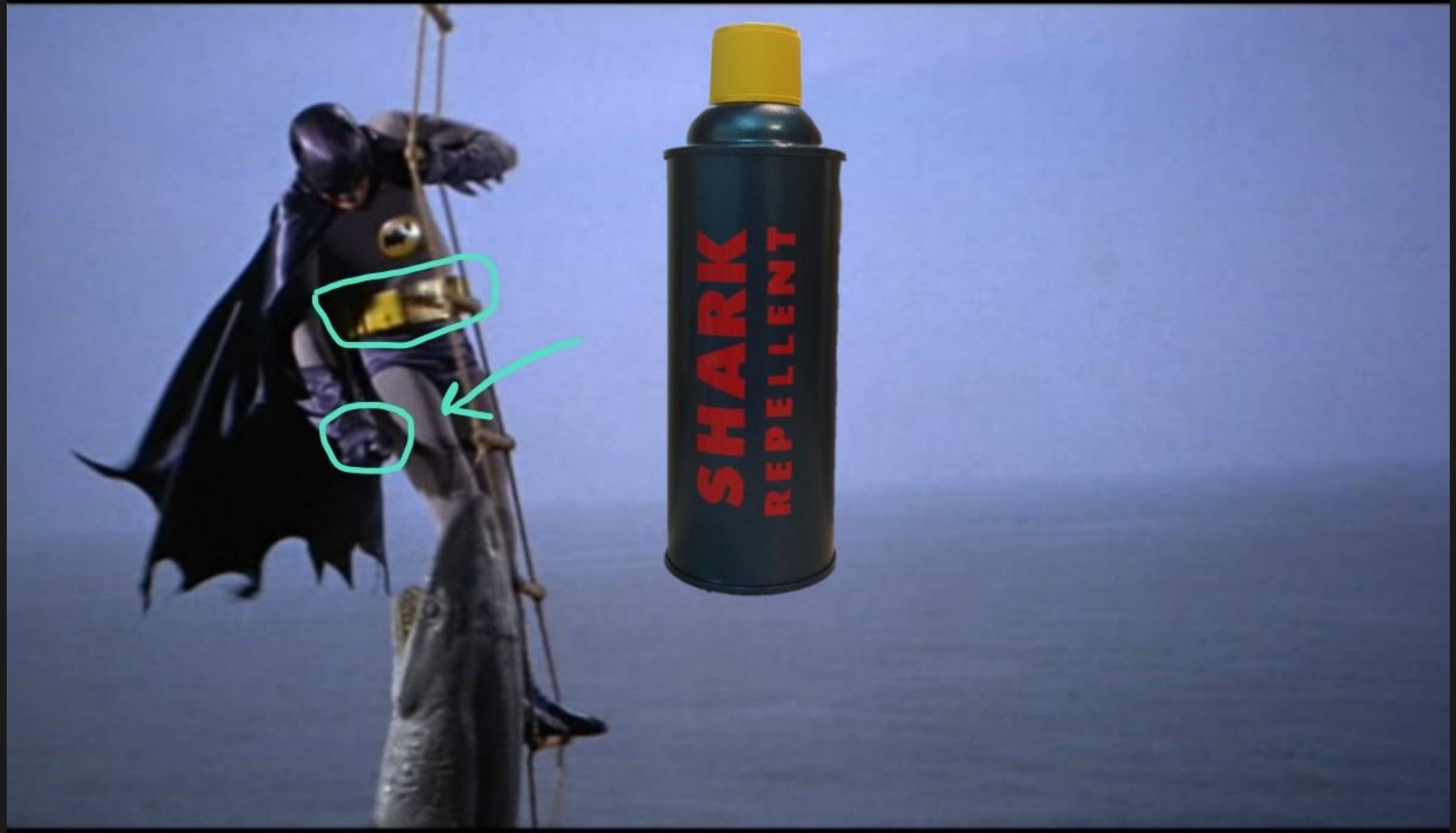
nova
net

```
$ az resource show
  --resource-group signus
  --name appinsight
  --resource-type microsoft.insights/component
  --query “properties.InstrumentationKey”
```

Let's talk a little about the anatomy of az commands.

nova
net

```
$ az resource show
  --resource-group signus
  --name appinsight
  --resource-type microsoft.insights/component
  --query “properties.InstrumentationKey”
“5c1fe79c-6870-11e9-88b0-00163ecccb07”
```


SHARK
REPELLENT

nova
net

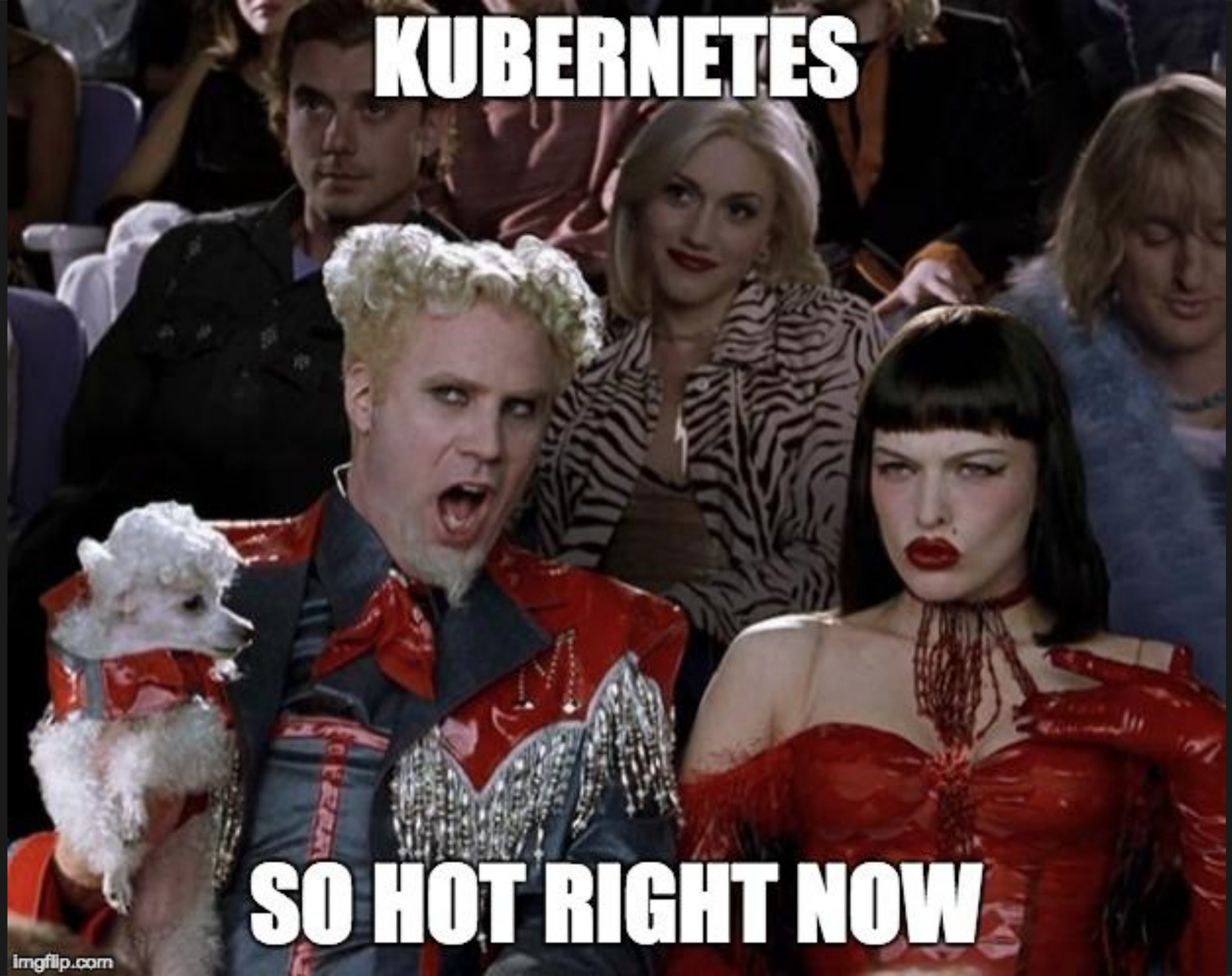
nova
net

Idempotent
(nearly always)

nova
net

Kubernetes

nova
net



nova
net

```
33
34
35
36  echo ---------------------- Secrets ----------------------
37
38  echo "Collecting Identity configuration"
39
40  IDENTITY_CLIENTSECRET=$(az keyvault secret show --vault-name signus-identity-$ENVNAME-key --name signus-app --query value) &&
41  IDENTITY_CLIENTSECRET=${IDENTITY_CLIENTSECRET//\"/}
42
43  echo "Collecting Redis configuration"
44  REDIS_KEY=$(az keyvault secret show --vault-name signus-$ENVNAME-key --name signus-app-$ENVNAME-redis-key --query value) &&
45  REDIS_KEY=${REDIS_KEY//\"/} &&
46  REDIS_CONFIGURATION="app-$ENVNAME-redisdb.redis.cache.windows.net:6380,password=$REDIS_KEY,ssl=True,abortConnect=False"
47
48  echo "Collecting Cosmos DB configuration"
49
50  COSMOSDB_KEY=$(az keyvault secret show --vault-name signus-$ENVNAME-key --name signus-app-$ENVNAME-cosmosdb-key --query value) &&
51  COSMOSDB_KEY=${COSMOSDB_KEY//\"/}
52
53  echo "Collecting Servicebus configuration"
54
55  SERVICEBUS_CONNECTIONSTRING=$(az servicebus queue authorization-rule keys list --resource-group signus-$ENVNAME-rg --namespace-name signus-$ENVN
56
57  if [[ $SERVICEBUS_CONNECTIONSTRING = *"Not Found"* ]]; then
58      SERVICEBUS_CONNECTIONSTRING=""
59  else
60      SERVICEBUS_CONNECTIONSTRING=${SERVICEBUS_CONNECTIONSTRING//\"/}
61  fi
62
```

```
$ kubectl create secret generic signus-app
    --from-literal=identity_clientsecret="$IDENTITY_CLIENTSECRET"
    --from-literal=redis_configuration="$REDIS_CONFIGURATION"
    --from-literal=cosmosdb_key="$COSMOSDB_KEY"
    --from-literal=connectionstrings_servicebus="$SERVICEBUS_CONNECTIONSTRING"
```

nova
net

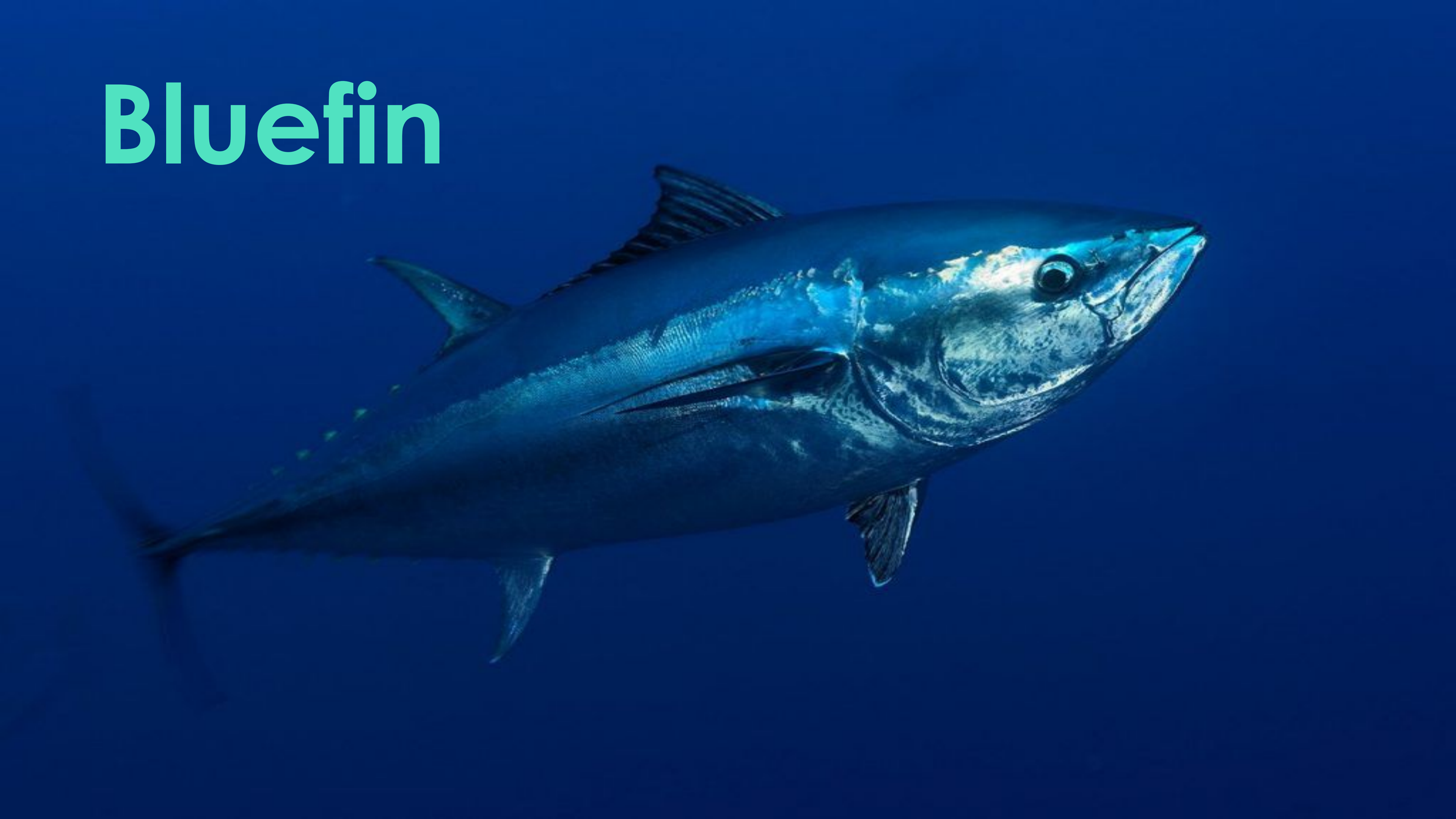
ARM (What is it good for)

```
5
6
7
8
9
10
11
12
13
14  let applyArmTemplate rg templatePath templateFile parameterFile =
15      azWorkingDir (sprintf "group deployment create -g %s --template-file %s --parameters %s"
16          rg templateFile parameterFile) templatePath
17          |> ignore
18
19
20
21
22
23
24
```

nova
net

FAKE

Bluefin



nova
net

```
Target.create "Azure-login" (fun _ →
    let azureServicePrincipalId = getArg "--servicePrincipalId"
    let azureServicePrincipalPassword = getArg "--servicePrincipalPassword"
    let azureTenantId = "6798f877-a1d1-41fb-a343-0d3e3a99876a"
    Az.login azureTenantId azureServicePrincipalId azureServicePrincipalPassword
)

Target.create "Apply-arm-template" (fun _ →
    let armTemplates = __SOURCE_DIRECTORY__ + "/arm-templates"
    let resourceGroup = sprintf "atlas-%s-%s-rg" applicationName env
    Az.applyArmTemplate env resourceGroup armTemplates
)
```

nova
net

```fsharp
let az arguments workingDirectory =
    execProcess "az" arguments (fun o →
        { o with DisplayName = "Azure CLI"; WorkingDirectory = workingDirectory })
```

nova
net

dotnet user-secrets

```bash
#!/bin/bash
# secrets.sh
ENVNAME=$(echo $1 | tr '[:upper:]' '[:lower:]')
if [ "$ENVNAME" ≠ "local" ] && [ "$ENVNAME" ≠ "dev" ]
then
    echo "usage: ./secrets.sh local|dev"
    echo "example: ./secrets.sh dev"
    exit 0
fi

azCmd="az.cmd" # Git-bash does not accept 'az'
if [ "$(uname)" == "Darwin" ] || [ "$(expr substr $(uname -s) 1 5)" == "Linux" ]
then
    azCmd="az"
fi

echo "Environment:  $ENVNAME"

dotnet user-secrets set mysecret "Global Azure Bootcamp"

```

nova
net

```
32
33    # secrets.sh modified
34    echo "Environment:  $ENVNAME"
35    echo "Logged into:  $($azCmd account show --query name)"
36    echo "Logged in as: $($azCmd account show --query user.name)"
37
38    dotnet user-secrets set mysecret "Global Azure Bootcamp"
39
```

nova
net

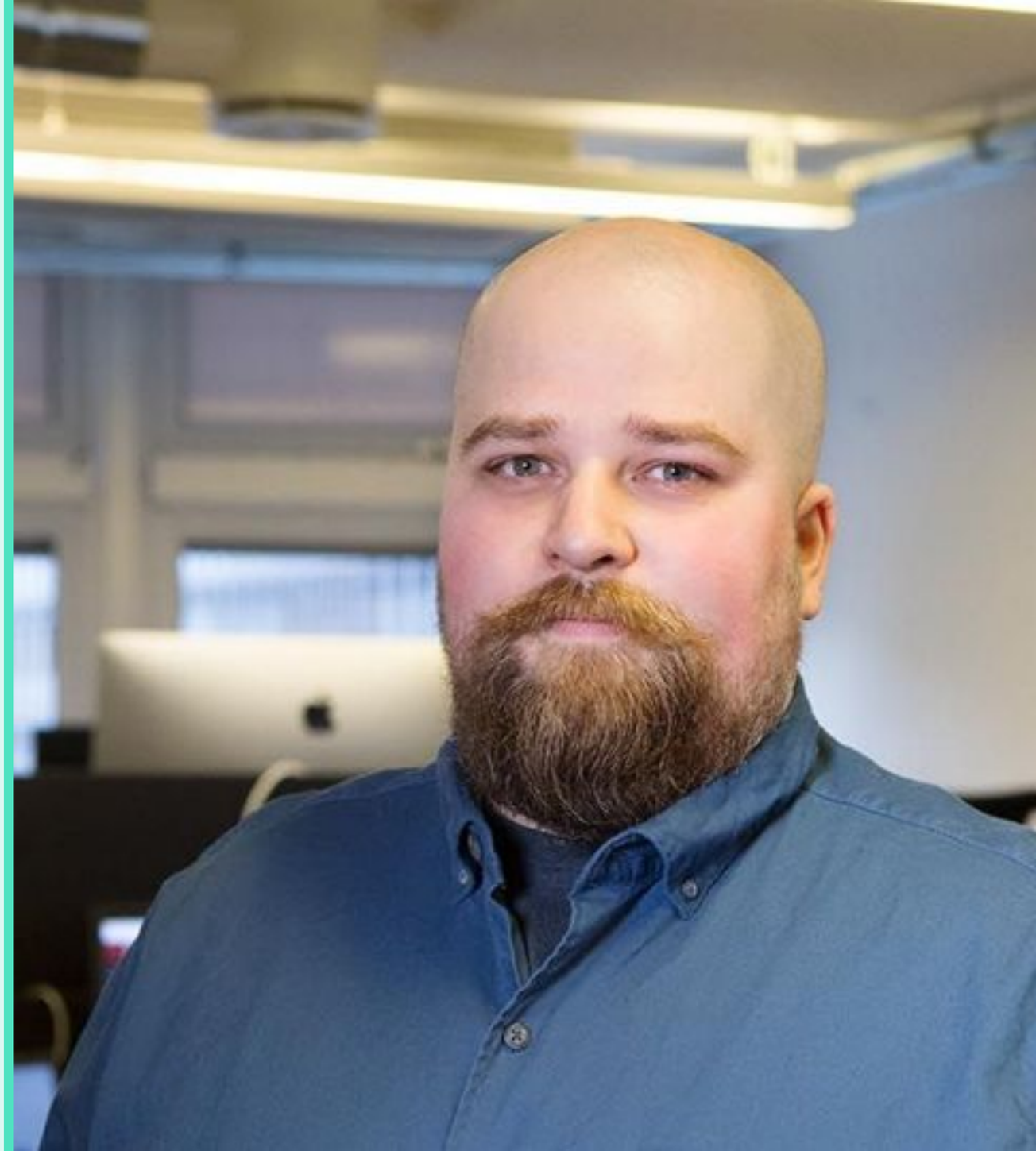
```
57  # AZURE KEYVAULT ANNOYING STRINGS
58  # mysecret = fire
59  $ MYSECRET=$($azCmd keyvault secret show --vault-name mykeyvault-$ENVNAME --name mysecret --query value)
60  $ echo "This is $MYSECRET!"
61  This is "fire"!
62  $ MYSECRET=$($azCmd keyvault secret show --vault-name mykeyvault-$ENVNAME --name mysecret --query value | sed -e 's/^"//' -e 's/"$//')
63  $ echo "This is $MYSECRET!"
64  This is fire!
```

nova
net

Johan Grønstad

Johan is a senior consultant at Novanet. He's currently working for Thon Hotels on both infrastructure and applications using his trusty Chromebook. His main passions include breaking bottlenecks and delegating all his responsibilities to scripts.

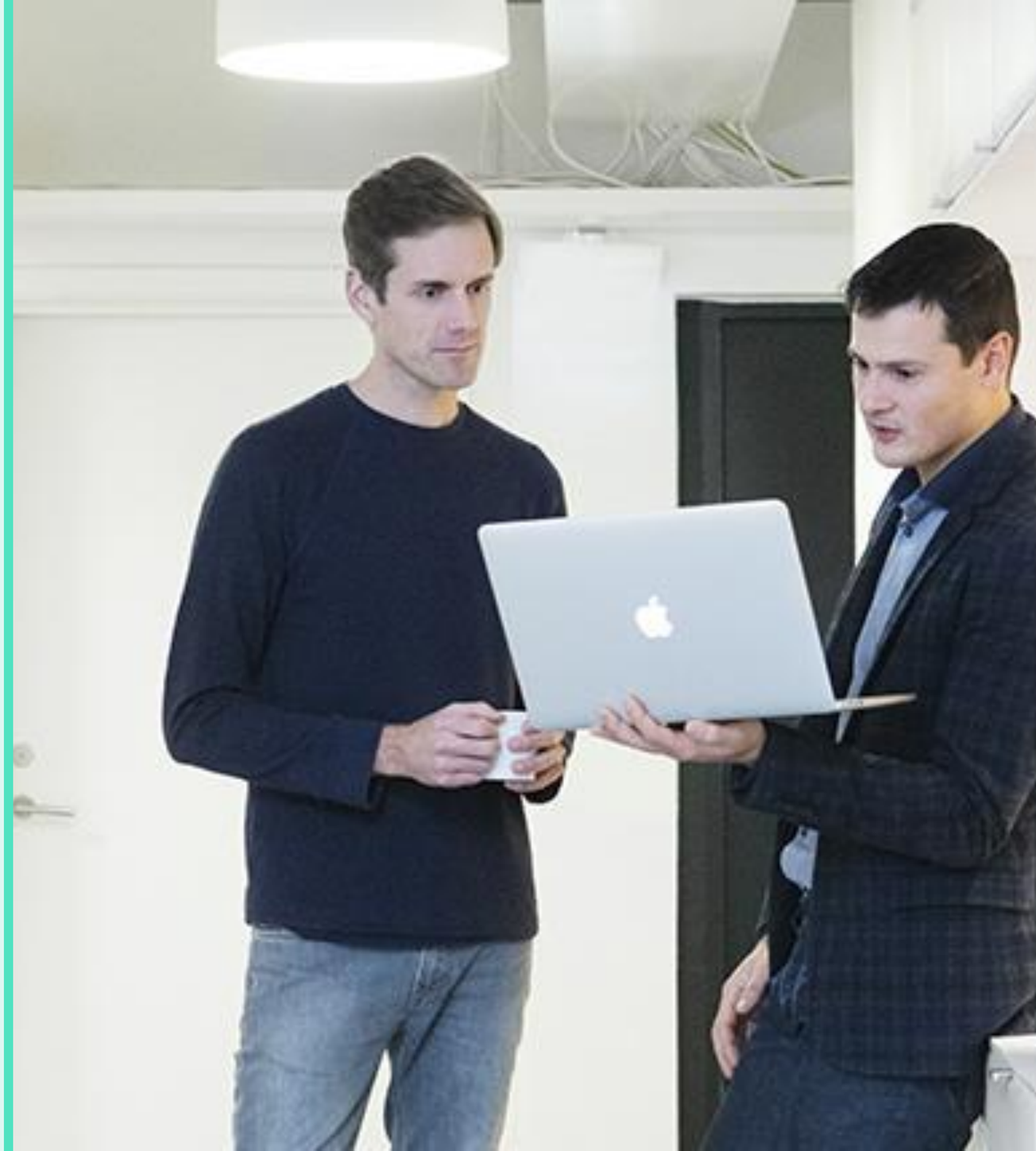
When he's not command lining all the things he enjoys silent mornings when no one else is awake.



nova
net

Hva gjør vi

- Applikasjonsarkitektur
- Applikasjoner
- Webapplikasjoner
- Hybride mobilløsninger
- Skyløsninger
- Sikkerhet i applikasjoner
- Autentisering og autorisasjon
- Integrasjoner
- Søk
- Rådgivning, ekspert-tjenester
- Tech lead/scrum master



nova
net

Bonus

nova
net

```
module Az =

    let exec arguments = az arguments "" |> ignore

    let login tenantId servicePrincipalId password =
        az (sprintf "login --service-principal --username %s  --password %s --tenant %s" servicePrincipalId password tenantId) ""
        |> ignore
```

nova
net

```fsharp
let private execProcess name arguments createOptions =

    let arguments = // Split a command on whitespace, ignoring quoted sections
      let regex = new Regex(@"[ ](?=(?:[^""]*""[^""]*"")*[^""]*$)", RegexOptions.Multiline)
      regex.Split(arguments)
      |> Array.filter(String.isNullOrWhiteSpace >> not)

    let joinArgs = String.concat " "

    let options = createOptions defaultOptions

    let cli = ProcessUtils.tryFindFileOnPath name
              |> function
              | Some cli -> cli
              | None -> failwithf "Can't find %s on path" (if String.isNotNullOrEmpty options.DisplayName then name else options.DisplayName)

    try
        CreateProcess.fromRawCommand cli (arguments)
        |> CreateProcess.withWorkingDirectory options.WorkingDirectory
        |> CreateProcess.redirectOutput
        |> CreateProcess.disableTraceCommand
        |> CreateProcess.addOnSetup (fun () ->
                Trace.tracefn "%s> \"%s\" %s \n" options.WorkingDirectory cli (joinArgs >> options.CensorTrace <| arguments)
        )
        |> Proc.run
        |> fun res ->
              if res.ExitCode <> 0 then failwithf "Step failed: %O" res.Result.Error
              res.Result.Output.Trim().Trim('"')

    with ex ->
        failwithf "Error calling %s %s  dir: %s \n %O" name (joinArgs >> options.CensorTrace <| arguments) options.WorkingDirectory ex
```

nova
net

Azure Service Bus

nova
net

```
3   namespace Bluefin
4
5   open Bluefin.Core
6
7   module Servicebus =
8       let getQueueConnectionString rg namespaceName queueName =
9           az (sprintf "servicebus queue authorization-rule keys list -g %s --namespace-name %s --queue-name %s --name read-write --query primaryConnectionString"
10              rg namespaceName queueName)
11
12      let getTopicConnectionString rg namespaceName topicName =
13          az (sprintf "servicebus topic authorization-rule keys list -g %s --namespace-name %s --topic-name %s --name read-write --query primaryConnectionString"
14              rg namespaceName topicName)
```

nova
net

Azure CLI Extensions

```
pr0nin@penguin:~$ az extension list-available | jq '.[] | {name: .name, summary: .summary}' -c
{"name":"aem","summary":"Manage Azure Enhanced Monitoring Extensions for SAP"}
{"name":"aks-preview","summary":"Provides a preview for upcoming AKS features"}
{"name":"alias","summary":"Support for command aliases"}
{"name":"appconfig","summary":"Provides a preview for upcoming App Configuration features."}
{"name":"azure-batch-cli-extensions","summary":"Additional commands for working with Azure Batch service"}
{"name":"azure-cli-iot-ext","summary":"Provides the data plane command layer for Azure IoT Hub, IoT Edge and IoT Device Provisioning Service"}
{"name":"azure-devops","summary":"Tools for managing Azure DevOps."}
{"name":"azure-firewall","summary":"Manage Azure Firewall resources."}
{"name":"db-up","summary":"Additional commands to simplify Azure Database workflows."}
{"name":"dev-spaces","summary":"Dev Spaces provides a rapid, iterative Kubernetes development experience for teams."}
{"name":"dms-preview","summary":"Support for new Database Migration Service scenarios."}
{"name":"dns","summary":"An Azure CLI Extension for DNS zones"}
{"name":"eventgrid","summary":"Microsoft Azure Command-Line Tools EventGrid Command Module."}
{"name":"express-route","summary":"Manage ExpressRoutes with preview features."}
{"name":"express-route-cross-connection","summary":"Manage customer ExpressRoute circuits using an ExpressRoute cross-connection."}
{"name":"find","summary":"Intelligent querying for CLI information."}
{"name":"front-door","summary":"Manage networking Front Doors."}
{"name":"image-copy-extension","summary":"Support for copying managed vm images between regions"}
{"name":"interactive","summary":"Microsoft Azure Command-Line Interactive Shell"}
{"name":"keyvault-preview","summary":"Preview Azure Key Vault commands."}
{"name":"log-analytics","summary":"Support for Azure Log Analytics query capabilities."}
{"name":"managementgroups","summary":"An Azure CLI Extension for Management Groups"}
{"name":"managementpartner","summary":"Support for Management Partner preview"}
{"name":"mesh","summary":"Support for Microsoft Azure Service Fabric Mesh - Public Preview"}
{"name":"mixed-reality","summary":"Mixed Reality Azure CLI Extension."}
{"name":"rdbms-vnet","summary":"Support for Virtual Network rules in Azure MySQL and Azure PostgreSQL resources"}
{"name":"resource-graph","summary":"Support for querying Azure resources with Resource Graph."}
{"name":"sap-hana","summary":"Additional commands for working with SAP HanaOnAzure instances."}
{"name":"signalr","summary":"Support for signalr management preview."}
{"name":"sqlvm-preview","summary":"Tools for managing SQL virtual machines, groups and availability group listeners."}
{"name":"storage-preview","summary":"Provides a preview for upcoming storage features."}
{"name":"subscription","summary":"Support for subscription management preview."}
{"name":"virtual-network-tap","summary":"Manage virtual network taps (VTAP)."}
```

nova
net

Azure Powershell Az Module

```
pr0nin@penguin:~$ pwsh
PowerShell 6.1.2
Copyright (c) Microsoft Corporation. All rights reserved.

https://aka.ms/pscore6-docs
Type 'help' to get help.

PS /home/pr0nin> get-help Get-AzApiManagement

NAME
    Get-AzApiManagement

SYNOPSIS
    Gets a list or a particular API Management Service description.


SYNTAX
    Get-AzApiManagement [-DefaultProfile
    <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] -Name
    <System.String> -ResourceGroupName <System.String> [<CommonParameters>]

    Get-AzApiManagement [-DefaultProfile
    <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] -Resource
    <System.String> [<CommonParameters>]
```