

Azure: Cifrado de secretos con SOPS.

Un saludo, espero se encuentre bien.

Tiempo de lectura: 4 minutos.

Descripción.

A continuación detallaremos, desde **Infraestructura Digicem - CyberSecurity**, la guía del paso a paso para:

- Cifrar Secretos de repositorios o estructuras de archivos en :azure: repos desde una llave:key-vault:

Guía. [↗](#)

- :key-vault: : Herramienta de almacenamiento de secretos dentro del :azure: portal.
- :azure:: Herramienta de gestión de recursos en la nube, a nivel de IaaS, PaaS, SaaS.
- :azure: **Repos**: Herramienta de gestión de pipelines en pasos por el branching strategy declarado de git-flow y trunk, entre los entornos: Dev, STA y Prod, algunos HotFixes y Features de recuperación de aplicaciones comprometidas.

Etapas. [↗](#)

Requerimientos. [↗](#)


- Tener una :azure: suscripción.
- Tener un equipo local donde administrar cambios de repositorios y subirlos hacia los repositorios en :azure: DevOps, sobre Repos.
- Contar con el archivo, repositorio o estructura a cifrar en AES256, bajo un formato JSON.

Emojis [↗](#)

:azure: **Plataforma del Portal de Azure - PaaS.**

:key-vault: **Key-Vault.**

Publicaciones. [↗](#)

-  • Paquetes de servicios.
- Entregable formalizado cifrado.



sops_3.7.3_amd64.deb
04 abr 2023, 08:25 pm

-  Se debe clonar el repositorio, respetar la estructura del contenido, para buen efecto al construir la imagen desde Docker.

dockerfile [↗](#)

```
1 FROM ubuntu
2
```

```
3 ADD . .
4
5 ENTRYPOINT tar -xvf azSops.tar; bash ./azSops.sh
6
```

- Se comprimen los **archivos sh** como **scripts** en **tar**, para evitar la conversión que hace **Git** sobre archivos **Windows** en **texto plano** de formato **LF (Linux)** a **CRLF** (el cual se corrompe al ser utilizado en **BASH**) que **no es legible en Unix**.

- El siguiente **Script** automatiza la construcción del contenedor dentro del **dockerfile**.

[docker-compose.yml](#)

```
1 version: '3'
2
3 services:
4
5     sops:
6         build:
7             context: .
8             dockerfile: ./dockerfile
9             tty: true
10
```

- El siguiente **Script automatiza la instalación del :azure: CLI dentro del contenedor.**

[deb_install.sh](#)

```

1  #!/usr/bin/env bash
2
3  #####
4  # This script does three fundamental things:
5  #   1. Add Microsoft's GPG Key has a trusted source of apt packages.
6  #   2. Add Microsoft's repositories as a source for apt packages.
7  #   3. Installs the Azure CLI from those repositories.
8  # Given the nature of this script, it must be executed with elevated privileges, i.e. with `sudo`.
9  #
10 # Remember, with great power comes great responsibility.
11 #
12 # Do not be in the habit of executing scripts from the internet with root-level access to your machine. Only tru
13 # well-known publishers.
14 #####
15
16 set -e
17
18 if [[ $# -ge 1 && $1 == "-y" ]]; then
19     global_consent=0
20 else
21     global_consent=1
22 fi
23
24 function assert_consent {
25     if [[ $2 -eq 0 ]]; then
26         return 0
27     fi

```

```

28
29     echo -n "$1 [Y/n] "
30     read consent
31     if [[ ! "${consent}" == "y" && ! "${consent}" == "Y" && ! "${consent}" == "" ]]; then
32         echo "'${consent}'"
33         exit 1
34     fi
35 }
36
37 global_consent=0 # Artificially giving global consent after review-feedback. Remove this line to enable interact
38
39 setup() {
40
41     assert_consent "Add packages necessary to modify your apt-package sources?" ${global_consent}
42     set -v
43     export DEBIAN_FRONTEND=noninteractive
44     apt-get update
45     apt-get install -y apt-transport-https lsb-release gnupg curl
46     set +v
47
48     assert_consent "Add Microsoft as a trusted package signer?" ${global_consent}
49     set -v
50     curl -sL https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor > /etc/apt/trusted.gpg.d/microsoft
51     set +v
52
53     assert_consent "Add the Azure CLI Repository to your apt sources?" ${global_consent}
54     set -v
55     # Use env var DIST_CODE for the package dist name if provided
56     if [[ -z $DIST_CODE ]]; then
57         CLI_REPO=$(lsb_release -cs)
58         shopt -s nocasematch
59         ERROR_MSG="Unable to find a package for your system. Please check if an existing package in https://pack
60         if [[ ! $(curl -sL https://packages.microsoft.com/repos/azure-cli/dists/) =~ $CLI_REPO ]]; then
61             DIST=$(lsb_release -is)
62             if [[ $DIST =~ "Ubuntu" ]]; then
63                 CLI_REPO="jammy"
64             elif [[ $DIST =~ "Debian" ]]; then
65                 CLI_REPO="bullseye"
66             elif [[ $DIST =~ "LinuxMint" ]]; then
67                 CLI_REPO=$(cat /etc/os-release | grep -Po 'UBUNTU_CODENAME=\K.*') || true
68                 if [[ -z $CLI_REPO ]]; then
69                     echo $ERROR_MSG
70                     exit 1
71                 fi
72             else
73                 echo $ERROR_MSG
74                 exit 1
75             fi
76         fi
77     else
78         CLI_REPO=$DIST_CODE
79         if [[ ! $(curl -sL https://packages.microsoft.com/repos/azure-cli/dists/) =~ $CLI_REPO ]]; then
80             echo "Unable to find an azure-cli package with DIST_CODE=$CLI_REPO in https://packages.microsoft.com
81             exit 1
82         fi
83     fi
84     echo "deb [arch=$(dpkg --print-architecture)] https://packages.microsoft.com/repos/azure-cli/ ${CLI_REPO} ma
85     > /etc/apt/sources.list.d/azure-cli.list

```

```

86 apt-get update
87 set +v
88
89 assert_consent "Install the Azure CLI?" ${global_consent}
90 apt-get install -y azure-cli
91
92 }
93
94 setup # ensure the whole file is downloaded before executing
95

```

- Por favor, puedes hacer uso de la siguientes estructuras como bases para lograr el fin solicitado.

.sops.yaml [🔗](#)

```

1 creation_rules:
2   - azure_keyvault: https://dev-kv-devops-compañia.vault.azure.net/keys/sops-key/L1av3DeC1fr4d0
3

```

- Se debe establecer el archivo, bajo extensión JSON, debido a la estructura de salida que relaciona SOPS con el cifrado AES256, el cual puede alterar según la extensión la estructura del cifrado cómo data o sin llaves.
- *El siguiente es un ejemplo, no dudes en modificarlo para cifrar, solo respeta la extensión JSON.*

env.json [🔗](#)

```

1 {
2   "Logging": {
3     "LogLevel": {
4       "Default": "ENC[AES256_GCM,data:pcOXU6/mRmhi4CY=,iv:W1urE1wsbICW25DKYjHD640coRKzhevvcKeZnVM",
5       "Microsoft.AspNetCore": "ENC[AES256_GCM,data:V5W0dx18dw=,iv:sAHY8IhKz4U05MCYT2a3fQgh4+x7ZL",
6     }
7   },
8   "AllowedHosts": "ENC[AES256_GCM,data:Pg==,iv:Itp5F1KaRSqqzCbJPzU2iRIS0vH0xZ7KUSIvhGwcuE=,tag:WaiLpHsL/xCgS",
9   "QuickCommApi": {
10     "AuthorityBaseUrl": "ENC[AES256_GCM,data:YU4j+OV6elr7RQxQtM98Rwutd0ph4TjgsGIqZPYKqdfa40VNJHPGpVpCoIL",
11     "BaseUrl": "ENC[AES256_GCM,data:EjDs36lX5RHRCisXYyBg9SMzie8CMp7NmG0KNbIvmHHEjBukCJYZEw==,iv:a8RXI6V",
12     "ClientId": "ENC[AES256_GCM,data:0PBf2e8mPp9lgxmI/OD9HUXZ8SDSP3Y30i/B1Qi6QwnM7G+7,iv:pxFHuLwc5Ksf3k",
13     "ClientSecret": "ENC[AES256_GCM,data:2CzXguqTXVQDDAbnjE0TGKMKGva4n5pqaPJXPeReJvMlpP29Wr35EcAhgeg=,i",
14     "OrganizationId": "ENC[AES256_GCM,data:PdwIcWFLBL2sus8NdEwI8AA0+VmcGkqfBpCZ1vLopvWBFrmE,iv:Ww2dArWk",
15     "SellerId": "ENC[AES256_GCM,data:0w7cCUWS0sNU3CneFthqF0q7CUVp32uEifQogXIaxjYtCtmf,iv:kw4yd6rPqRNAYC",
16     "LoginClientId": "ENC[AES256_GCM,data:u/kqNATWniIXYT0Tqr9Q/aQtBYaHfZ036JofL0gowms65hcc,iv:NAJRT9Lic",
17     "LoginClientSecret": "ENC[AES256_GCM,data:Ii/dm4lvQiNfizxJk6q+qQJjc3SBApj4UuFHU1pJe9f0v+2c9zQAE4+Kk",
18   },
19   "ShippingExperienceApi": {
20     "BaseUrl": "ENC[AES256_GCM,data:sDe97hMZN3e0WrNsV1eHmuwXskez3qYX0mdGVAM5g8u0VksSjZMek3oCBP8aXCTa0gf",
21     "CreateShippingExperienceFunctionKey": "ENC[AES256_GCM,data:oKIIMbVqCshX/1j79EIicTa8rItbl7U3d8dlzqj",
22     "GetShippingExperienceScoreOptionsFunctionKey": "ENC[AES256_GCM,data:rUSPoR6lNUAe8VjZddJTkqLoIkRuL/A",
23     "GetShippingExperienceByClientRucFunctionKey": "ENC[AES256_GCM,data:aZ+WT33KSiuDw6Xxe3C14mPMxvdTSYe",
24   },
25   "ShippingDatesApi": {
26     "BaseUrl": "ENC[AES256_GCM,data:NxKPIZoPDZQY/gpewIrbbhkufqbLui5F0uZV84fL0nhXEFkN/yH+IxqFVhRTtzcxxn",
27     "GetAvailableShippingDatesFunctionKey": "ENC[AES256_GCM,data:6NzRsQMrGqr/J3/G95uHo25ZPCzj+DlFwbTKTs",
28     "SaveSaleOrderShippingFunctionKey": "ENC[AES256_GCM,data:yNGe6MIFONHUMj8ufPFVfm27pAmpPPmX6M6ImEcEM4Z",
29   },
30 }

```

```

30     "TicketManagerApi": {
31         "BaseUrl": "ENC[AES256_GCM,data:hw9j64c2u9M0s7q1hE+/1ZxhEy2kyCVkB/m6tDSw/IMUGAUa7zcCLF11ygHMMuwTgC
32     },
33     "PaymentsApi": {
34         "BaseUrl": "ENC[AES256_GCM,data:dMn/Ss5k9mTVGA9ZoEJB1pmNx7hF/GvMT06vKdfz92WtFpN9+TyXYpmyvQDkFJI,iv
35         "CreditSummaryFunctionKey": "ENC[AES256_GCM,data:ouUml1INYqZXpaXU6YcOKt17FMwbWmbGqazLvrvX3tvRrJx/A>
36         "ChargeFunctionKey": "ENC[AES256_GCM,data:WqrIhCPAS0w1GbPFOJmK3uBHY+ZkvFH4GU1mKpBEYJZDzWmd3m93DzR0j
37         "ReimburseFunctionKey": "ENC[AES256_GCM,data:VcVQgh5AoItIhuAGI62DavLb1JfVpFHYIfP1TNjufPsK7ryhQxm3KJ
38     },
39     "LoyaltyApi": {
40         "BaseUrl": "ENC[AES256_GCM,data:h9/aLDqrHdYsjkNrt7d1zq78UZ0oDdkd2zcUouUcnSmwIUyPoC85t2Yhe0tHG19sal
41         "LoyaltyPointsSummaryByClientIdFunctionKey": "ENC[AES256_GCM,data:WFQddDxNk23Y1X2sFSvLYNuon0BfVw5c
42         "LoyaltyPointsTransactionsByClientIdFunctionKey": "ENC[AES256_GCM,data:9vIqsCWC+dXrtnzEQgZnK1SX4BTH
43         "AddLoyaltyPointTransactionFunctionKey": "ENC[AES256_GCM,data:mBqfUJoA5CiyTOqVKPxFsTJ5nkyfWw78kF/or
44     },
45     "MarketingApi": {
46         "BaseUrl": "ENC[AES256_GCM,data:sCQYyvQ9CVYkI415UzmpsdPqvhy56w1l/I+3p64w+f1c3dfuGn70ymBWM0zSf3VvXQ=
47         "GetTestimonialsFunctionKey": "ENC[AES256_GCM,data:rpYZZgpFpVLq8WnLgOvWSU8iFR7Bg53316b9ctA2ozJ4gMV7
48         "GetNoveltiesFunctionKey": "ENC[AES256_GCM,data:GoDcuoBtJrO20rvKJiGy1Fm/JJc8UnRr9WX/nnV5g4zvMyRtx75
49     },
50     "NotificationsApi": {
51         "BaseUrl": "ENC[AES256_GCM,data:BviIjbk9rlhE1QbaypQTPSmVP3hz2nhcg3Hs4m4FrVKJbQYy5uJMwb+6gyGdTQ+EY87
52         "SendOrderClaimEmailFunctionKey": "ENC[AES256_GCM,data:bj4fYNiZwNqPH/E4H0WYTwoR2gwiQ0iyVOqfNnHISHMu
53         "SendWelcomeClientCreatedEmailFunctionKey": "ENC[AES256_GCM,data:06AVq3j1QWHQRvcoUIURh5sDq8hpo/VncC
54     },
55     "ProductsApi": {
56         "BaseUrl": "ENC[AES256_GCM,data:lYvm8wQdXcca2NS/WPKQxhMP2fsretyerFWtL+ho8RxwpbFx4HqxdH9TXovkRapnx9
57         "GetProductCategoriesFunctionKey": "ENC[AES256_GCM,data:q3VY+vvYyG5cnxP2Lhmc1rngq2uZQkXEUc1vAbWGBVf
58         "GetWarehousesFunctionKey": "ENC[AES256_GCM,data:GJWfoXzj5G/zLqlxVtMsWSKqMb3kGkZn5F0Cpb3V5zrLueLh1u
59     },
60     "BackOfficeApi": {
61         "BaseUrl": "ENC[AES256_GCM,data:07M3JTQyVJaX/1N3zN0hifs+Pj0+4Q06SCYGE53dn0xPp1SsvAT3t2I9nDNMk0cflUV
62         "CreateDistributorSaleOrderFunctionKey": "ENC[AES256_GCM,data:vKWGNcZ3JxvG03ncn/vXidcilfN1lQGM0H+c>
63     },
64     "TMS": {
65         "ProviderId": "ENC[AES256_GCM,data:GjWm81WtXHlgO7iyLylX0zfEA6ycmaxWB830LF+2vyDMg8fh,iv:z8V7efanX4+H
66     },
67     "SacContactInformation": {
68         "Name": "ENC[AES256_GCM,data:jLKR0aq6a1v5BwGqLVaVhpr9ed0tTuI,iv:RgR9jZvunguiLPQmbm5WNXZNA9HuxAZfo1
69         "Email": "ENC[AES256_GCM,data:00pZ8nC0NgpiK5HxvGu9IHBa25iLQIiX6k=,iv:HqT72TF+M970+L45TkITGdLi1Pe0+
70         "PhoneNumber": "ENC[AES256_GCM,data:xTniukYqXlfsTEk=,iv:uMwxSR4NC1/Hf/65cZGiAfZ4tJD1rsqgcYVmqpxA6iI
71     },
72     "Cache": [
73         {
74             "Name": "ENC[AES256_GCM,data:stBipS3udA==,iv:Mr8470TFYn0hpQ0m7ZnhyybKxRzpJacbfD80fKwyOyM=,t
75             "Hours": "ENC[AES256_GCM,data:Bw==,iv:Pd1XAyeEr3vSQ36vG0PFrcu0AFHcgGRRqIxvE4/Yi+o=,tag:z0Tf
76             "Minutes": "ENC[AES256_GCM,data:SQ==,iv:X/61Br5p0ghVx46I4f6Zc7n43jz2yJMo7/J01xEoCgw=,tag:g1
77             "seconds": "ENC[AES256_GCM,data:Xw==,iv:rV7fH4xyFfyzeRgfCJvAkrX2G0Kax1Rd2t37V74mbDs=,tag:cg
78         },
79         {
80             "Name": "ENC[AES256_GCM,data:zVq4jocJtVwh91Iq0R+3WB4ajAar9ZHT3g==,iv:HFenOuHXzyEqycLqt/03Cy
81             "Hours": "ENC[AES256_GCM,data:nA==,iv:EzU5xp6SvBa1Sba5un/DFDcfFff4cFHv6ZwnihUx7ZI=,tag:h3Kj
82             "Minutes": "ENC[AES256_GCM,data:Tg==,iv:DaQj+8LpQcDZb03S6ewwsVHgjK6SAL2MUYMaQTNh+TI=,tag:Y>
83             "seconds": "ENC[AES256_GCM,data:Xg==,iv:gZs9Avkej0cNvIKc2pprc5MVz6/ZPunm099eiLL4KHY=,tag:25
84         },
85         {
86             "Name": "ENC[AES256_GCM,data:U+Iz1/0Ca/OtxqZ9Bugn,iv:Znx3V50MZne/2Dfg1Q//qCZ9ysV7QrSz1sr2FE
87             "Hours": "ENC[AES256_GCM,data:qQ==,iv:qhjViy2GHvhPg2wz+K1ZnXZWYuKTQHYp585NJmr8g1E=,tag:aHbc

```

```

88         "Minutes": "ENC[AES256_GCM,data:uQ==,iv:kHxpYLoEXW6++CTEEw0hmzj5tETH2EbJ0qbdAzwr/1U=,tag:D6
89         "seconds": "ENC[AES256_GCM,data:OQ==,iv:c4BprMkztShvBHPsmCHA943ssN+sICeLJ501rCJX1Fk=,tag:W0
90     },
91     {
92         "Name": "ENC[AES256_GCM,data:oG7odeVSiwxTF1ka0g==,iv:qmU/50iScbKys6S2S551S54G1M5r2XgaGvMHks
93         "Hours": "ENC[AES256_GCM,data:9yg4LQ==,iv:ey08mEHtXajOQpgyKbnEiFR2Huz5GFM2qguxCsAcnsQ=,tag:
94         "Minutes": "ENC[AES256_GCM,data:zQ==,iv:Vv89UgIVmJWzmIVB0cHwIUQ0GUp00cNKNQoWBpMQQ8=,tag:Rc
95         "seconds": "ENC[AES256_GCM,data:0g==,iv:jTgd+JKbDwDpMT8L4kDe+Myr1qYhseM9I0BH8GzI3BI=,tag:A7
96     },
97     {
98         "Name": "ENC[AES256_GCM,data:nMGWH/6zZAoaamrj5UqHz0Yf,iv:sNR5aS+JgXB2hXxduwqKTEubRWBO/3SSeA
99         "Hours": "ENC[AES256_GCM,data:zQY=,iv:/VmGbq0LsHqrbCWWhitNo5ZASGePgZqRT2qE/Y4jzR4=,tag:gqNf
100        "Minutes": "ENC[AES256_GCM,data:PW==,iv:rUpp30Kc2pT2TaEbIjZX0WYzVtjG7+oB+LRDPH24BD0=,tag:oE
101        "seconds": "ENC[AES256_GCM,data:vQ==,iv:yZeV7HzZlIIJiMg6gdWM/V+LrCi9Rs6fdt1pt41PnSs=,tag:Cj
102    },
103    {
104        "Name": "ENC[AES256_GCM,data:uNcZ5tP/IXuPcwHD38KzfE32,iv:+rB4JnPw21tTdpEV7SalkSiXSJ1ghnSCAC
105        "Hours": "ENC[AES256_GCM,data:BUo=,iv:Behwn5V3y6n7AjtYqFKMqdc9utSyDatSjuR/bJ8YCbW=,tag:MT4i
106        "Minutes": "ENC[AES256_GCM,data:Xg==,iv:1S05fFqo7eQnKi+1HL81ldHz6EqsfiyXucy+uRTmKS=,tag:Jc
107        "seconds": "ENC[AES256_GCM,data:+A==,iv:8XpA041DZSY0qw7G6LF6HD0bqJUXsA0SJdx1t0lrUhI=,tag:+c
108    },
109    {
110        "Name": "ENC[AES256_GCM,data:cFGrDo8FnRjymPy8TJJW7aEGCQ==,iv:sn4cUNjmWcJd9ad3PcQFMNk0XeYDit
111        "Hours": "ENC[AES256_GCM,data:4v8=,iv:6m9L4Pf/I4jTuV4efvQRb7Y52M0VxCmm02ZFfqxqQUM=,tag:Tr+C
112        "Minutes": "ENC[AES256_GCM,data:JQ==,iv:xbARUEFm/J3v+8wWlyfNdH0WKmSwDIBSX/CZaBQLEns=,tag:dE
113        "seconds": "ENC[AES256_GCM,data:NA==,iv:9wWrbJA37LQMTfkCeV/pkCK+9YYF6u15VKMYicw7g9w=,tag:hA
114    }
115 ],
116 "sops": {
117     "kms": null,
118     "gcp_kms": null,
119     "azure_kv": [
120         {
121             "vault_url": "https://dev-kv-devops-compañia.vault.azure.net",
122             "name": "sops-key",
123             "version": "L1av3DeC1fr4d0",
124             "created_at": "2023-04-04T02:38:51Z",
125             "enc": "S3cR3t0C1fr4d0"
126         }
127     ],
128     "hc_vault": null,
129     "age": null,
130     "lastmodified": "2023-04-04T02:38:53Z",
131     "mac": "ENC[AES256_GCM,data:6Qkv30ppqXKNymWbP+arhTMHfxx22HRIukbTQdTJB1clWaqC/KjUp+JX1NR16mPnE0uIBi
132     "pgp": null,
133     "unencrypted_suffix": "_unencrypted",
134     "version": "3.7.3"
135 }
136 }
137

```



• El siguiente **Script automatiza el inicio de sesion dentro del portal, desde :azure: CLI dentro del contenedor.**

azure.sh [↗](#)

```
1 #!/usr/bin/env bash
2
3 azure login
4
```

- ☐ Corre una **tarea interactiva** con la **imagen creada** dentro del **contenedor**.

```
o * Ejecutando tarea: docker run --rm -it sops:latest

To sign in, use a web browser to open the page https://microsoft.com/devicelogin and enter the code EG7F4343T to authenticate.
```

Inicio del docker, desde las sentencias declaradas en el Entrypoint.

- ☐ **Autentica** dentro de la **API** :azure: con la **llave temporal**.



Especificar el código

Escriba el código que se muestra en su aplicación o dispositivo.

EG7F4343T

Siguiente

Autenticación de Azure CLI.

Microsoft Azure



Selección de la cuenta

Está iniciando sesión en **Microsoft Azure Cross-platform Command Line Interface** en otro dispositivo que se encuentra en **Colombia**. Si no es usted, cierre esta página.



Dev Tool Progresol
dev.tools@progresol.com
Conectado

⋮

Seleccionamos cuenta de acceso.



dev.tools@progresol.com

¿Está intentando iniciar sesión en Microsoft Azure CLI?

Continúe solo si ha descargado la aplicación de una tienda de aplicaciones o un sitio web de confianza.

Cancelar

Continuar

Finalizamos la autenticación.

Construcción. [↗](#)

- Por favor, puedes hacer uso de la siguiente sentencia para construir tu proyecto.

Comandos Docker. [↗](#)

```
1 docker compose -f "path\docker-compose.yml" up -d --build
```

- **docker:** API.
- **compose:** Orquestador de contenedores.
- **-f:** Establece un argumento para la ruta de un archivo..
- **path:** Ruta relativa del archivo docker compose.
- **docker-compose.yml:** Script de construcción del dockerfile.
- **up:** Levanta el servicio desde la construcción.
- **-d:** activa una sesion detach o permanente, ante cortes de conexión o salidas idle, sin hacer exit o logout.
- **--build:** Construye el proyecto del contenedor como una imagen para docker.

- Por favor, puedes hacer uso de las siguientes sentencias para validar accesos a los recursos y la herramienta.

Comandos Docker Compose. [↗](#)

```
1 docker run --rm -it name-tag -d
```

- **docker:** API.
- **run:** Corre la imagen desde la API.
- **--rm:** Elimina la imagen actual cacheada o predeterminada.
- **-i:** Inicia una tarea o sesion interactiva.
- **-t:** Etiqueta para la imagen.
- **name-tag:** Nombre de la imagen: versión, como ejemplo "**sops:lastest**".
- **-d:** activa una sesion detach o permanente, ante cortes de conexión o salidas idle, sin hacer exit o logout.

Pruebas. [↗](#)

- Por favor, puedes hacer uso de las siguientes sentencias para validar accesos a los recursos y la herramienta.

:azure: Comandos. [↗](#)

```
1 az login
```

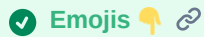
- **az**: API.
- **login**: Inicio de sesión sobre la plataforma :azure:, en una ventana aparte bajo un código de validación.

```
1 export AZURE_CLIENT_ID="AzUr3-C1eNt-1D"
```

- **export**: Declara la variable en sistemas operativos Unix igual Linux.
- **AZURE_CLIENT_ID**: Nombre de variable a setear.
- **=**: Operador lógico del valor de la variable.
- **"AzUr3-C1eNt-1D"**: Valor de la variable a retornar, al ser llamada.

```
1 export AZURE_CLIENT_SECRET="AzUr3-C1eNt-5E(r3T";  
2 export AZURE_TENANT_ID="AzUr3-T3n4Nt-1D"
```

- **;**: Operador de salto de sentencia en línea.



:azure: **Plataforma del Portal de Azure - PaaS.**

Desplegar. [↗](#)

Comandos Docker. [↗](#)

```
1 docker run --rm -it name:tag -d
```

- **docker**: API.
- **run**: Corre la imagen desde la API.
- **--rm**: Elimina la imagen actual cacheada o predeterminada.
- **-i**: Inicia una tarea o sesión interactiva.
- **-t**: Etiqueta para la imagen.
- **name:tag**: Nombre de la imagen: versión, como ejemplo **"sops:lastest"**.
- **-d**: activa una sesión detach o permanente, ante cortes de conexión o salidas idle, sin hacer exit o logout.

:key-vault:Comandos. [↗](#)

```
1 sops -e env.json > env.enc.dev2; cat env.enc.dev2
```

- **sops**: Herramienta de cifrado.
- **-e**: Parámetro o argumento de cifrar secretos.
- **env.json**: Archivo o estructura a cifrar.
- **>**: Redireccionamiento de salida según error, de la sentencia anterior.
- **env.enc.dev2**: Documento receptor de salida.
- **cat**: Concatena los valores desde consola sin requerir abrir un binario sobre el archivo.

☐ **Copia** la salida de la **concatenación** en **Linux** con el **archivo que contendrá los secretos**.

```

pe:str]",
      "GetClientsInformationFunctionKey": "ENC[AES256_GCM,data:c5+Ephl09u0VVG09cYqx0tFwjvK0dj8HG
wVEgUg2d9mZIXnSt88AYD+u7tygk4SiiYk27/fot7g=,iv:uogTkdcg6BdSQa7FzF5l1f6mJemAcxkh1Z0KXpXAR/4=,tag:aRsTbIAkMe
bwHnNB4CHLJw==,type:str]",
      "GetProductCategoriesFunctionKey": "ENC[AES256_GCM,data:tvSmoY6ISURIdYFRivvqJ3trAA0b17yyKR
iJL0e0hx676sE60I2hnq5vT6cjNnarSPLeE/YtDw=,iv:E0opTwnjSWghyCD4kp4DhSe26/1S/dWJaGVMwJqWvag=,tag:5ZsYkgu8b+f
5+DEJEmcL/A==,type:str]",
      "UpdateProductCategoryFunctionKey": "ENC[AES256_GCM,data:rQ2M4ytr50yEwgUuLKVP5Q5uHL64dEhbn
KY278GFMzZdiCEdb8meUWdmvKSUHVGMp+7DCu3bQPQ=,iv:/Yqp79ZuzhPb/9EAS7QN88v50yq7RE3xHMcBQizj5HE=,tag:dVegt0uu+I
0/md4L1J7yzQ==,type:str]"
    },
    "Auth0": {
      "Domain": "ENC[AES256_GCM,data:zH+k4jAh7p0AuJvj+Tf7gMknFANryYmbmKYQKw9qWw,iv:Qjtx3HXkIHb
JDFx6ZTCXzM0dGT236cGC8f4SHP/Zb2I=,tag:zAZ4fqCLafsvnsKB615M6Q==,type:str]",
      "ClientId": "ENC[AES256_GCM,data:EEeAjQ53srMFNuFR05kviIdagAsY4Z5e0VcX0lr0LKw=,iv:HT7sZG3R+
9f50IZdJtPuYwZmmRyOBE1Gmkc9hbCbAU=,tag:Yyg+jRsmW9667bi3cBZiXw==,type:str]"
    },
    "sops": {
      "kms": null,
      "gcp_kms": null,
      "azure_kv": [
        {
          "vault_url": "https://dev-kv-devops-nanaykuna.vault.azure.net",
          "name": "sops-key",
          "version": "dc7611e069b64c1ea46a9dbb98452d72",
          "lastmodified": "2023-04-26T20:44:00Z",
          "mac": "ENC[AES256_GCM,data:DKKLpUBvNu17e1oLw2EVHaAsMfsh5/DuJyAeCdIIg3VvDt0Pz+gw6Vs7ytrjwg
ypHVKT9z07hM0XGGPrmiQJYZoQbLc8Q3yPPJ5l1E8ceIiu1VLF28M1lC1nY+5/nRXksZl1gsbjPraj30bdtvR70rasjMhU+2Yy1mm/hdMF
iqA=,iv:oecmYVfc+NLY70prJUBHhrXvoj1+uaZmhXcSwiamEFs=,tag:IpCUj6V1lWnqTorQf0Tawg==,type:str]",
          "pgp": null,
          "unencrypted_suffix": "_unencrypted",
          "version": "3.7.3"
        }
      ]
    }
  }
}

```

Output del binario UNIX "Cat".

- ✗ • **Cat** como **binario** en **Unix** será deprecado, como **buena practica de shell code** se debe usar **<**, sin necesidad del **pipeline** como **tubería de filtrado en salida (|)**.

✓ Emojis 🗨️ 🔗

:key-vault: **Key-Vault**.

👉 Ahora se puede dimensionar el nivel de ejecución, como tiempo a la hora de gestionar este requerimiento.

¡Mil gracias por la atención prestada!

Cualquier duda me puedes contactar...

:WhatsApp: **+573058288031**

como mi usuario :slack: