

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. [↗](#)

i 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. [↗](#)

- i** • Paquetes de servicios.
- Entregable formalizado cifrado.



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

- i** 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**.

- i** • 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 trust #
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 interactive mode
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.gpg
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://packages.microsoft.com/repos/azure-cli/dists/ can be used"
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/repos/azure-cli/dists/."
81         exit 1
82     fi
83 fi
84 echo "deb [arch=$(dpkg --print-architecture)] https://packages.microsoft.com/repos/azure-cli/ ${CLI_REPO} main" \
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/mRmh4CY=,iv:W1urE1wsbICW250KYjHD640coRkZhevvCkeZnVMGBK8=,tag:Gst537NfKyyRIWT3jiBzDA=,type:str]",
5       "Microsoft.AspNetCore": "ENC[AES256_GCM,data:V5w0dx18dw=,iv:sAHY8IhKz4U05MCTY2a3fQgh4+x7zLQ3oWqy4t6GjyzE=,tag:DPI5eN8EVxRONTF0wakcow=,type:str]"
6     }
7   },
8   "AllowedHosts": "ENC[AES256_GCM,data:Pg==,iv:Itp5F1KaRSqqzCbjqPzU2iRISovH0xZ7KUSIvhGwcuE=,tag:WaiLpHsL/xCg9x4jKkGxTg==,type:str]",
9   "QuickCommApi": {
10     "AuthorityBaseUrl": "ENC[AES256_GCM,data:YU4j+0V6elr7RQxQtM98Wutd0ph4Tjs6IqZPYKqdfa40VNJHPgVpCoIUro/sEjdg/F,iv:/h6gmZTVxIx5f/JIw1Hn/4R41b+F+QRKWrpenG"
11     "BaseUrl": "ENC[AES256_GCM,data:EjDs36lX5RHCisXyYBg9SMzie8CMp7NmGOKNbIvmHHEjBukCJYZEw==,iv:a8RXI6RVwG+K+3le+nwTmvG1hgeTPctjEHGF6yXb1g=,tag:ugdBjq5dsV"
12     "ClientId": "ENC[AES256_GCM,data:0PBf2e8mPp9lgxmI/OD9HUXZ8SDSP3Y30i/B1QieQwnM76+7,iv:pxFHuLwc5Ksf3bBPqCdK5l151xM1vSgtYKNUY0Etuk4=,tag:wjQ3NNiIXRIUDu0hJ"
13     "ClientSecret": "ENC[AES256_GCM,data:2CzXguqTVXDQAbnjEOTGKMKGa4v5npqAPJXPeReJvM1pP29Wr35EcAhgeq=,iv:wMk5GLBLrUM5rfZm+jcV19setqgpmGky7Sf6IjWfMgg=,tag:8"
14     "OrganizationId": "ENC[AES256_GCM,data:PdwIcWFLBL2sus8NdEwI8AAO+VmcGkqfBpCZ1vLopvWBFrmE,iv:Ww2dArwKY4cv1z2ffHmA78CMZd/kORQPokPQWsyBC5U=,tag:ErAJ+gxAlGd"
15     "SellerId": "ENC[AES256_GCM,data:0w7cCUW50sNU3CneFthqF0q7CUPv32uEifQogXIaxjYtCtmf,iv:kw4yd6rPqRNAV0QoemUEG9Zx25Es2HjvTx8m5J1R7Tc=,tag:ZkqZRBZs03oKe3YEL"
16     "LoginClientId": "ENC[AES256_GCM,data:u/kqNATWniIXtYT0qr9Q/aQtBYaHfZ036JofL0gowms65hcc,iv:NAJRT9Lid0WLC85tfSoFNY2qrvyD1AQLdTzoIy2j00o=,tag:50CPWphMwk33c"
17     "LoginClientSecret": "ENC[AES256_GCM,data:Ii/dm41vQinfiZxJk6q+qQJjC3SBApj4UuFHu1pJe9f0v+2c9zQAE4+Kbms=,iv:I3pps20ZclHpfJ3jKRLMu6ktfk+OzSuyGvChYazuJPw=,"
18   },
19   "ShippingExperienceApi": {
20     "BaseUrl": "ENC[AES256_GCM,data:sDe97hMZN3e0WrNsVieHmuwXskez3qYX0mdGVAM5g8u0VksSjZMek30CBP8aXCTa0gFzSurr26uqhLU=,iv:tVvDwcWn4gzTAPih5uE0Pu6jW4c9gNJ+Czq"
21     "CreateShippingExperienceFunctionKey": "ENC[AES256_GCM,data:oK1lMbVqCsHX/1j79EIIcTa8rItb1U73d8dlzqj13h1mAWYLmx/233BKWBTr/1AnuKFnoclH3aM=,iv:W2Sm/vxozdJ"
22     "GetShippingExperienceScoreOptionsFunctionKey": "ENC[AES256_GCM,data:rUSP0r61NUAe8VjZddJTKqLoIkRuLA9u0R7w6Rf00dn2A88twa0TnvG783aqL/qv92uK8EUm/14=,iv:W8"
23     "GetShippingExperienceByClientRucFunctionKey": "ENC[AES256_GCM,data:az+WT33KSiuDw6Xxe3C14mPMxvdTSYenKZJ0640bysRT0DqT3+AuxUtwPqpn7K+CRMEVzPLVa64=,iv:+T9"
24   },
25   "ShippingDatesApi": {
26     "BaseUrl": "ENC[AES256_GCM,data:NxKPIZoPDZQY/gpewIrbbhkufqBLUi5F0uZV84fL0nhXEFkn/yH+IxqFvHRTtzcxxmFuU0N,iv:Xt8Dpa/7F/1vGpzE0Csfdi0QRQfwrPvadKukSEPy1dA"
27     "GetAvailableShippingDatesFunctionKey": "ENC[AES256_GCM,data:6NzRsqMgGr/J3/695uHo25ZPCzj+DlFwbTKTs1UP15i3YHzKDT3q4TJTk2Y7p83vGN07YtNqUy=,iv:QsnXiKsZSxl"
28     "SaveSaleOrderShippingFunctionKey": "ENC[AES256_GCM,data:yNGe6M1FONHUMj8rPFVfmZ7pAmpPPmX6MImEcEM42LXHdc3C20CxtYvaGS0xs5Umqa011Puf9g=,iv:kny9Mm8qQR/Rq"
29   },
30   "TicketManagerApi": {
31     "BaseUrl": "ENC[AES256_GCM,data:hw9j64c2u9M0s7q1he+/1ZxhEy2kyCVkwB/m6tDSW/IMUGAUa7zcCLF1lygHMMuwtG0MTZFA,iv:kpvhaNpN21l6BpwZXo9xIVMLwPUMgDv1YwLXvbuUbY"
32   },
33   "PaymentsApi": {
34     "BaseUrl": "ENC[AES256_GCM,data:dMn/Ss5k9mTVGA9ZoEJB1pmNnx7Hf/GvMT06vKdfz92WtFpN9+TyXypmyvQdkFJI,iv:jwDnfc8rDQWZummMhbxehZjLVrRM1xCG3ISjRzDw=,tag:Ew"
35     "CreditSummaryFunctionKey": "ENC[AES256_GCM,data:ouUml1INyqZxpaXU6YcOKt17FMwbWmbGqazLvrVx3tVrRjX/AxcntGwtKpmimShZo8ma3JyrGbo=,iv:V0cyb3h76NP0fYNY02uJdkl"
36     "ChargeFunctionKey": "ENC[AES256_GCM,data:WqrIHcPAS0w1GbPFOJmK3uBHy+ZkvFH46U1mkPbEYJZDzWmd3m93DzR0j9x+9l1QY+QGURlTYL0=,iv:4+d7v4Qn4Bkmfuq+3jyVc7Fko/1T0"
37     "ReimburseFunctionKey": "ENC[AES256_GCM,data:VcQqgh5aoltIhuAGI62DavlLbJlFvPhYIFp1TnJufPsK7ryhQxm3KIOW2lPKgcsZi77fL7Ky79Y=,iv:f8i7tSNYtEx0/F3VmoZj+DRTAm"
38   },
39   "LoyaltyApi": {
40     "BaseUrl": "ENC[AES256_GCM,data:h9/aLDqrHdysjKnrT7dizq78Uz8oDdkd2zcUouUcnSmwIuyPoC85t2Yhe0tHG19saUbfN+HD3M=,iv:SLYUHLhkT3ZCmoegleRrNkWyM04gdT3DWNpcqMc"
41     "LoyaltyPointsSummaryByClientIdFunctionKey": "ENC[AES256_GCM,data:WfQddXnK23Y1X2sFsvLYNuonOBfVw5qgWlQq6Wv7Dq830aNa0EV3cBexBey8esay+/75D5gwc=,iv:4uzgLi"
42     "LoyaltyPointsTransactionsByClientIdFunctionKey": "ENC[AES256_GCM,data:9vIqgsCWc+dXrtnzEQgZnK1SX4BTH9+moWkSQzXqmLVp6DP66b0tRUHVyh1SPnXej4orveLa2H0=,iv::"
43     "AddLoyaltyPointTransactionFunctionKey": "ENC[AES256_GCM,data:mBqfJJoA5C1yToqVKPxFsTJ5NkpfWw78Kf/org+Dpv5Z7yQxfq4VeesaaI6rpqrfC7yQUuY6WAY=,iv:akF+L2dtr"
44   },
45   "MarketingApi": {

```

```
46         "BaseUrl": "ENC[AES256_GCM,data:scQYyvQ9CVYkI415UzmpsdPqvhy56w1l/I+3p64w+f1c3dfuGn70ymBWM0zSf3VvXQ==,iv:ndrT1Q/p79IGLgqYwufj2zf3rZRBiD93uZw1f8aSheQ==,tag:
47         "GetTestimonialsFunctionKey": "ENC[AES256_GCM,data:rpYZZgpFpVLq8wnLg0vWSU81FR7Bg5316b9cTa2ozJ4gMVT4RbB0RiE6F9G66b1D39iV9u8nUY4==,iv:K8byeHqZbjymrsrbP0IWI
48         "GetNoveltiesFunctionKey": "ENC[AES256_GCM,data:GoDocuoBtRj020rVKJi6y1Fm/JJC8UnRr9Wx/nNv5g4zvMyRtx75Jrd0FHxuyFuyKFTQVvyqphd9q2w==,iv:fx9RpKGNfneeFxbiveX8kFw
49     },
50     "NotificationsApi": {
51         "BaseUrl": "ENC[AES256_GCM,data:BviIjbk9rlhe1QibaypQTPSmVP3hz2nhcg3Hs4m4FrVKJbQYy5uJMbw+6gyGdTQ+EY87dmg==,iv:meHaeQci6lSoTWHF7jvbrCkSr/N9v1WPbcMmrnlxno:
52         "SendOrderClaimEmailFunctionKey": "ENC[AES256_GCM,data:bj4fYNIzwnqPh/E4H0WYtwor2Gwiq0iYVQqfNnHishMu/UIN5XfK71U0qo+FrRxErER3WUk02Jbs=,iv:ofta4k9lovG031qi!
53         "SendWelcomeClientCreatedEmailFunctionKey": "ENC[AES256_GCM,data:06AVq3j1QWHRvcoUIURh5sDq8hpo/Vnc6/MeQZgGLbQpxl3vu7pb0C2C+6ew9IvNzLsD7/m7GY=,iv:6ftX/gv
54     },
55     "ProductsApi": {
56         "BaseUrl": "ENC[AES256_GCM,data:lYvm8wqOdXcca2NS/WPKQxhMP2fsretyerFwtL+ho8RxpwbFxF4HqxH9TovkRapnx9UMKs9/GPorQ==,iv:qcZwXnuXiMegyc6lebEsCy3DeCZ2kkf9X0H
57         "GetProductCategoriesFunctionKey": "ENC[AES256_GCM,data:q3VY+vvYyG5cnpX2Lhmc1rngq2uZQkxEUc1vAbwGBVf/H48I2A9IkaRS+jq76vsOPCdUnLsz+Qg=,iv:0jNrmA/pH99wwcR
58         "GetWarehousesFunctionKey": "ENC[AES256_GCM,data:GJWfoXzj56/zLq1xVtMsWSkQMb3kGwZN5F0Cpb3V5zrLueLh1u0nYTKuo+1MPnQIDmePul+D+vY=,iv:F9Bjx1zT5t9UdeCHiagzTI
59     },
60     "BackOfficeApi": {
61         "BaseUrl": "ENC[AES256_GCM,data:07M3JTQyVJaX/1N3zN0hifs+Pj0+4Q06SCYGE53dn0xPp1SsvAT3t2I9nDNMk0cf1UVrdnXPPsfJ5nKEoA==,iv:0PKxL+ge1BBLDSnVJuokuaVaMJG0wPe
62         "CreateDistributorSaleOrderFunctionKey": "ENC[AES256_GCM,data:vKWGncZ3ixvG03ncR/vX1dcilfn1L1Qgm0H+cxYmwoXj1CW0C16mRXQ70DQXqX7MxpN/T8xb140o=,iv:6+XSk2cEr.
63     },
64     "TMS": {
65         "ProviderId": "ENC[AES256_GCM,data:GjWm81WtXhlg07iyLyLX0zfEA6ycmaxwB830LF+2vyDMg8fh,iv:z8V7efanX4+HT2Dw00rcvVVV74q30R3PtJbVsMszXSA=,tag:t0tByIHxBiXCJHH:
66     },
67     "SacContactInformation": {
68         "Name": "ENC[AES256_GCM,data:JLKR0aq6a1v5BwGqLVaVhpbr9ed0tTuI,iv:RgR9jZvungu1LPQmbm5WNXZNa9HuxAZfo1KJoHOAML8=,tag:zf3NFjcpRpK0tjz6xN5nnA==,type:str]",
69         "Email": "ENC[AES256_GCM,data:00pZ8nCCoNgpiK5HxvGu9IHBa25l1QIiX6k=,iv:HqT72TF+M970+L45TKITGdLi1Pe0+q+85tjZc4gfsMc=,tag:GqW65FaxCyo0PoFyy3Dzrg==,type:str]
70         "PhoneNumber": "ENC[AES256_GCM,data:xTniukYqXlfsTEk=,iv:uMwxSR4NC1/Hf/65cZGiAfZ4tJDi1rsqgcYVmqpxA6iI=,tag:uOoC7EKb9LNxApzc6lccw==,type:str]"
71     },
72     "Cache": [
73         {
74             "Name": "ENC[AES256_GCM,data:stBipS3udA==,iv:Mr8470TFYn0hpQ0m7ZnhyybKxRzpJacbfD80fKwyOyM=,tag:0lsXADxGQx905M6fhfR0tg==,type:str]",
75             "Hours": "ENC[AES256_GCM,data:Bw==,iv:Pd1XAyeEr3vSQ36vG0PFrcu0AFHcgGRRqIxxvE4/Yi+o=,tag:z0Tp62n0fFDb1GQxMgYTNQ==,type:str]",
76             "Minutes": "ENC[AES256_GCM,data:SQ==,iv:X/61Br5p0ghVx46I4f6Zc7n43jz2yJMo7/J01xEoCgw=,tag:g11g0tgd87T2Moe8E033ng==,type:str]",
77             "seconds": "ENC[AES256_GCM,data:Xw==,iv:rV7fH4xyFfyzeRgCjVakR2G0KAx1Rd2t37V74mbDs=,tag:c0iUDjYmLw/Y8ddcv4PbZw==,type:str]"
78         },
79         {
80             "Name": "ENC[AES256_GCM,data:zVq4jocJtVvh91Iq0R+3wB4ajAar9ZHT3g==,iv:Hfen0uHXZyEgycLqt/03Cy3Nh0cgF0H8zq/MeatDjVY=,tag:uQJ6lQggBii2qLe0Ps/pg==,t
81             "Hours": "ENC[AES256_GCM,data:nA==,iv:EzU5xp6SvBalSbaSun/DFDcfFff4cFHv6ZwmiHux7ZI=,tag:h3kj7C24R+f2aXf8leRADg==,type:str]",
82             "Minutes": "ENC[AES256_GCM,data:Tg==,iv:DaQj+8LpQcDZb03S6ewwsVHgJ6SAL2MUyMaQTnh+TI=,tag:YXCQcHE8xk2Cv1BUBSsy1g==,type:str]",
83             "seconds": "ENC[AES256_GCM,data:Xg==,iv:gZs9AvkejOcNvIKc2pprc5MVz6/ZPunm099eiLL4KHY=,tag:25l11MT6wM5P7mm315G0q==,type:str]"
84         },
85         {
86             "Name": "ENC[AES256_GCM,data:U+Iz1/0Ca/0txqZ9Bugn,iv:Znx3V50MzNe/2Dfg1Q//qCZ9ysv7QrSz1sr2FEVrx4M=,tag:GQ3eavdR4412tNmpZ6zStA==,type:str]",
87             "Hours": "ENC[AES256_GCM,data:qQ==,iv:qjViy2GhvhPg2wz+K1ZnXZWyUKTQHYp585Njmr8g1E=,tag:aHbqyk6VNCi2X2zPGrRYTQ==,type:str]",
88             "Minutes": "ENC[AES256_GCM,data:uQ==,iv:kHxpYLoEXw6++CTEEewhmzj5tETH2EbJ0qbdAzwr/U=,tag:D6UoH3VQhftVphkDKv6nBw==,type:str]",
89             "seconds": "ENC[AES256_GCM,data:0Q==,iv:c4BprMkztShvBHPMSMCHA943ssN+sICeLJ501rCJX1FK=,tag:W0rbbcWm4G0s3kM89nSQ==,type:str]"
90         },
91         {
92             "Name": "ENC[AES256_GCM,data:oG7odeVSiwXTF1ka0g==,iv:qmuU/50iScbKys6S2S551S54G1M5r2Xga6VMHksHsWWE=,tag:+2E/q0T+HLRwt60sEvjmIw==,type:str]",
93             "Hours": "ENC[AES256_GCM,data:9yg4LQ==,iv:ey08mEhTxaJ0PqpyKbne1FR2HuZ56FM2gquxCsAcnsQ=,tag:05RUQbBJQXmoSL0a4qVnvA==,type:str]",
94             "Minutes": "ENC[AES256_GCM,data:zQ==,iv:Vv89UgIVmJwmIVB0cHwIUQ0G0pp00cNKNQ0wBPMQ08=,tag:RcdHVn44SILEKU1fuIP4lw==,type:str]",
95             "seconds": "ENC[AES256_GCM,data:0g==,iv:jTgd+JKbDwDpMT8L4kDe+Myr1qYhseM9I0B8GzI3BI=,tag:A7IS0TsmzTaLjP+tyY5/Qw==,type:str]"
96         },
97         {
98             "Name": "ENC[AES256_GCM,data:nMGWH/6zZAOaamrj5UqH20Yf,iv:sNR5a5+JgXb2hXxduwqKTEubRWB0/3SSeAnc6jn+seA=,tag:vH9aE4e/t5tDPaunoGaiww==,type:str]",
99             "Hours": "ENC[AES256_GCM,data:zQY=,iv:/VmGbq0LsHqrbCWRhItNo5ZAsGePgZqRT2qE/Y4jzR4=,tag:gqNfoYf2K3EQ7SK/Nhs20w==,type:str]",
100            "Minutes": "ENC[AES256_GCM,data:Pw==,iv:rUpp30Kc2pt2TaEb1jZXOWyZVTjG7+oB+LRDPH24BD0=,tag:0EZM12Kn1LEWUzJubX6Xzg==,type:str]",
101            "seconds": "ENC[AES256_GCM,data:vQ==,iv:yZev7HzLIIJiMg6gdWM/V+LrC19Rs6fdt1pt41PhSs=,tag:CjuoY8xRIM1SrycqqWKL=,type:str]"
102        },
103        {
104            "Name": "ENC[AES256_GCM,data:uNCZ5tP/IXuPcwHD38KzfE32,iv:+rB4jNPw21tTdpEV7Sa1kSiXJ1gHnSCAGx/RAMEe+W=,tag:FFgveZAI3C3P06VIDu0aQ==,type:str]",
105            "Hours": "ENC[AES256_GCM,data:BUo=,iv:Behwn5V3y6n7ajtYqFKMdc9utSyDATSjuR/bJ8YCbws=,tag:MT4iULSgLARwZ2XN0uA1gQ==,type:str]",
106            "Minutes": "ENC[AES256_GCM,data:Xg==,iv:1S05ffQo7eQnKi+1HL8l1dHzGEqsfieyXucy+uRTmKs=,tag:JJLz+SwBmqeZ6u9EMmyQMA==,type:str]",
107            "seconds": "ENC[AES256_GCM,data:+A==,iv:8XpAO41DZSYQw7G6LF6HD0bqJUXsAOSJdx1t01rUHI=,tag:+dtmxg8a/rUPAhcP3UTAEq==,type:str]"
108        },
109        {
110            "Name": "ENC[AES256_GCM,data:cFGrDo8FnRjymPy8TJJW7aE6CQ==,iv:sn4cUNjmWcJD9ad3PcQFMnk0XeYDibG0ad+7D51Ha/E=,tag:eAIT9hAwnL0uhmbpqRGZRQ==,type:str]
111            "Hours": "ENC[AES256_GCM,data:4v8=,iv:6m9L4Pf/I4jTuV4efvQRb7Y5ZM0VxCmm02ZFfQxqQUM=,tag:Tr+CoVibbJJkRn+h+R40pw==,type:str]",
112            "Minutes": "ENC[AES256_GCM,data:JQ==,iv:xBARUEFm/J3v+8wWlyfNDh0WkmSwDIBSX/CZaBQLENS=,tag:dEmZyPfvb/kshWfwie0yGA==,type:str]",
113            "seconds": "ENC[AES256_GCM,data:NA==,iv:9wMrbJA37LQMTfKcEv/pkCK+9YYF6u15vKMYicw7g9w=,tag:hAXhJU+MAJ4XLY7ips1+7g==,type:str]"
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+arhTMHfxx22HRiukbTQdTJB1c1WaqC/KjUp+JX1NR16mPnE0uIBiibwrRIu+1nzQrheGqcQuKN3WzVzVQ0U7B7tBtoaDnHQ999eliZArI
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 ^o

```

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

```

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

```

* 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. ^o

• 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 sesion 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.

✓ Emojis 🍌 [↗](#)

: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 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.

:key-vault:Comandos para cifrado. [🔗](#)

```
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+EpHlo9uOWGo9cYqx8tFkIjvK8dj8HG
wVEgUg2d9mZIXnSt88AYD+u7tygk4SiiYk27/fot7g=,iv:uogTkdcg68dSq7FzF5lif6mJemAcxkh1Z0KXpXAR/4=,tag:aRsTbIAkHe
bwHmWb4CHLJw==,type:str]",
    "GetProductCategoriesFunctionKey": "ENC[AES256_GCM,data:tvsvMoY6ISURIdYFRivvqJ3trAA0b17yyKR
iJLoe0hxC676sE60I2hnq5vT6cjNNarSPLee/YtDw=,iv:E0opTwnjSWghyCD4kp4DHSe26/IS/dwJagVMwJqWvag=,tag:5ZsYkgu8b+f
5+DEJEmcL/A==,type:str]",
    "UpdateProductCategoryFunctionKey": "ENC[AES256_GCM,data:rQ2M4ytr50yEwigUuLKVP5Q5uHL64dEhbn
KY278GFMzZdiCEdbBmeUMDmwKSUHVgMp+7DCu3bQPQ=,iv:/Yqp79ZUzhPb/9EAS7QN88v50yq7RE3xHMcBQiZj5HE=,tag:dVegt0uu+I
0/md4L1J7yzQ==,type:str]"
  },
  "Auth0": {
    "Domain": "ENC[AES256_GCM,data:zH+k4jAh7p0AuJvj+Tf7gMknFANryYmbmKYQKW9qlwar,iv:Qjtc3HXKkIHb
JDFx6ZTCXz0dGt236cGC8f4SHP/Zb2I=,tag:zAZ4fqCLafsvnsK8615M6Q==,type:str]",
    "ClientId": "ENC[AES256_GCM,data:EEeAjQ53smfNufR05kviIdagAsY4Z5e0VcX01r0LKw=,iv:HT7sZG3R+
9f50IZdJtPuYwZmmRyOBEx1Gwkc9hbCbAU=,tag:Yyg+jRsmW96G7bi3cBZiXw==,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:DKKLPUbVnu17e1oLw2EVHaAsHfsh5/DuJyAeCdIig3Wdt0Pz+gw6Vs7ytr-jwg
ypHVKT9z07hMBXGGPrmiQJYzoQbLc8Q3yPPJS11E8ceIiU1VLF28M1LC1nY+5/nRXksZ11gsbjPraJ30bdtvR70rasjMhU+2Yy1mm/hdMF
iQA=,iv:oeCmYVfc+Nly70prJUBHhrXVoj1+uaZmhXc5wIamEFs=,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 práctica de shell code se debe usar <, sin necesidad del pipeline como tubería de filtrado en salida (|).

:key-vault:Comandos de descifrado. [🔗](#)

```
1 sops --decrypt env.json
```

- **sops**: Herramienta de cifrado.
- **--decrypt**: Parámetro o argumento para descifrar secretos.
- **env.json**: Archivo o estructura a cifrar.

✅ 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: