

Caso Práctico 1: Infraestructura Global y Seguridad en AWS

Tarea 1 (50 puntos): Utilizando Cloud Shn script (Bash + AWS Cli) que muestre cada una de las regiones de AWS junto con sus zonas de disponibilidad asociadas. El debes crear u

La primera parte fue fácil de resolver, unas cuantas búsquedas en internet bastaron para dar con el comando **describe-regions**. A partir de este punto empezaron a salir las preguntas en base a los ejemplos dados en la página de referencia.

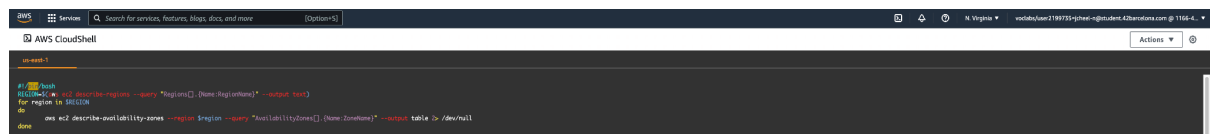
El Script contiene dos comandos diferentes: el primero para listar las “**region**”; el segundo para las “**Availabilities-zones**”.

Se emplea una variable **\$REGIONS** donde almacenar las diferentes líneas de comando. Para ello, fue necesario filtrar la tabla de **Regions** por columna **RegionName** de con la flag **—output** en modo texto (*text*).

Una vez guardado el nombre de las regiones en **\$REGIONS** con un bucle **for** y la flag **describe-availability-zones** para listar las **Availabilities-zone**. La flag **—region** permite listar las **Availabilities-zones** de una región concreta.

De esta manera con la variable definida **\$REGIONS** se utiliza el nombre para mostrar las **Availabilities-zone** de las zonas disponibles.

Este comando hace un **—query** de las **ZoneName**, cómo puede dar error este comando por temas de permisos, se utiliza la redirección de errores para no mostrar los mensajes por pantalla.

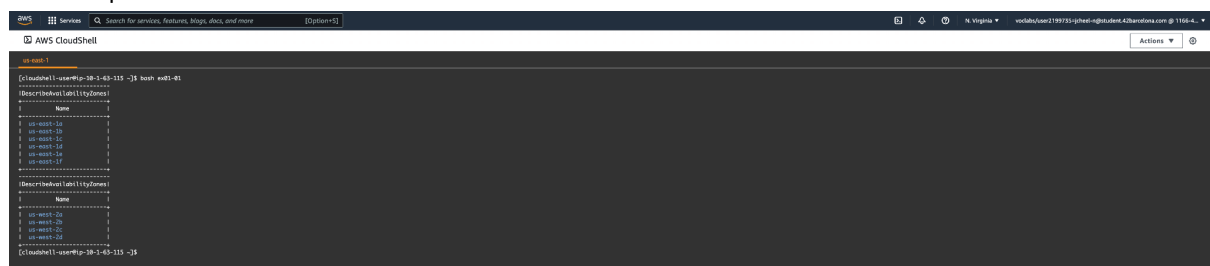


```

aws
Services Search for services, features, blogs, docs, and more [Options]
AWS CloudShell Actions
vs code
# aws
# aws ec2 describe-regions --query "Regions[0].RegionName" --output text
for region in $(cat $REGIONS)
do
  aws ec2 describe-availability-zones --region $region --query "AvailabilityZones[*].ZoneName" --output table > /dev/null
done

```

Como output obtenemos esto:



```

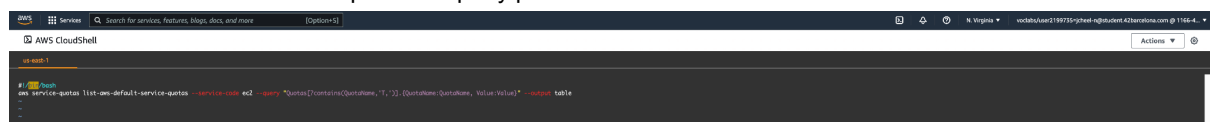
aws
Services Search for services, features, blogs, docs, and more [Options]
AWS CloudShell Actions
vs code
[cloudshell] user@ip-10-1-45-115 ~$ aws ec2 describe-availability-zones
(DescribeAvailabilityZones)
+-----+
| Name |
+-----+
| us-east-1a |
| us-east-1b |
| us-east-1c |
| us-east-1d |
| us-east-1e |
| us-east-1f |
+-----+
(DescribeAvailabilityZones)
+-----+
| Name |
+-----+
| us-east-2a |
| us-east-2b |
| us-east-2c |
| us-east-2d |
+-----+
[cloudshell] user@ip-10-1-45-115 ~$

```

Tarea 2 (50 puntos): Crea otro script que muestre el número máximo de instancias EC2 de tipo T que tenemos permitido crear en la región de North Virginia (Pista: AWS Cli permite consultar las “**quotas**” que tenemos asignadas por cada servicio).

Para este segundo ejercicio fue más complicado encontrar el comando que mostraba la tabla así que entre varios compañeros encontramos la documentación y conseguimos hacer un comando bastante simple.

El comando se compone de **service-quotas** y **list-aws-default-service-quotas** para mostrar los quotas de un servicio en concreto además emplea un **—query** para filtrar la información necesaria.



```

aws
Services Search for services, features, blogs, docs, and more [Options]
AWS CloudShell Actions
vs code
# aws
# aws service-quotas list-aws-default-service-quotas --service-code ec2 --query "Quotas[?contains(QuotaName, 'T_')].QuotaName,QuotaValue" --output table

```

OUTPUT

```
AWS Services Search for services, features, blogs, docs, and more [Distro#1] | AWS CloudShell Actions
```

```
aws bash -l  
[CloudShell-user@ip-10-1-45-115 ~]$ ls  
lsattr default servicequotas  
    QuotaUsage      Billing  
AWS Standard (A, C, D, H, M, R, S, T, X) Spot Instance Requests  S O I  
Running On-Demand Standard (A, C, D, H, M, F, R, S, T, X) Instances  S O I  
[CloudShell-user@ip-10-1-45-115 ~]$
```