

CLI automation lab

Dr Peadar Grant

October 23, 2022

In this week's lab we are going to learn how to capture information from the AWS CLI output, rather than doing so by hand.

1 JSON

AWS CLI outputs JSON by default as you have seen so far. Instead of manually copying and storing IDs, we can use our local PowerShell / Bash to capture this information for us. This will help later on when we can write scripts to automate some of this work for us.

1.1 JSON parsing in Powershell

We will use the example of create-vpc.

```
# Run aws ec2 create-vpc but capture the output instead
$Output = aws ec2 create-vpc --cidr-block 10.0.0.0/16
```

```
# View the output
$Output
```

```
# Convert from JSON to PS objects
$VpcInfo = $Output | ConvertFrom-Json
```

```
# Pull out the VpcId using the dot notation
$VpcId = $VpcInfo.Vpc.VpcId
```

1.1.1 Faster version

```
# Run aws ec2 create-vpc, capture output and convert to PS objects
$VpcInfo = aws ec2 create-vpc --cidr-block 10.0.0.0/16 | ConvertFrom-Json
```

```
# Pull out the VpcId using the dot notation
$VpcId = $VpcInfo.Vpc.VpcId
```

1.2 Even faster version

If you know what the output of particular commands look like, and you're interested only in one parameter you can do the store immediately. Consider instead:

```
$VpcId = (aws ec2 create-vpc --cidr-block 10.0.0.0/16 | ConvertFrom-Json).Vpc.VpcId
```

1.3 JSON parsing in Bash

Bash does not natively understand JSON but the `jq` utility does.

2 Lab exercise

Make sure you can login to the AWS Console. Then use the CLI to do the following **without manually copying any IDs or other information!**:

1. Create a VPC LAB_VPC, IP range 10.0.0.0/16. Store the ID as \$VpcId.
2. Create a Subnet LAB_SUBNET_1 within your VPC, IP range 10.0.1.0/24. Store the ID as \$SubnetId.
3. Turn on auto IP address assignment in the subnet.
4. Create an Internet Gateway LAB_GATEWAY. Store the ID as \$GatewayId.
5. Attach the internet gateway to your VPC.
6. Get the route table ID and store it as \$RouteTableId.
7. Alter the route table to send traffic for anywhere 0.0.0.0/0 to the internet gateway.
8. Create a security group LAB_GROUP storing its ID as \$GroupId.
9. Modify the security group to permit traffic inbound on SSH (22) and RDP (3389) protocols from anywhere 0.0.0.0/0.
10. Create an instance LAB_INSTANCE to run Linux.
11. Wait until the instance is running.
12. Get the instance ID and store as \$InstanceId.
13. Get the instance's public IP and store as \$PublicIp.
14. Use SSH (linux) to login to the instance.