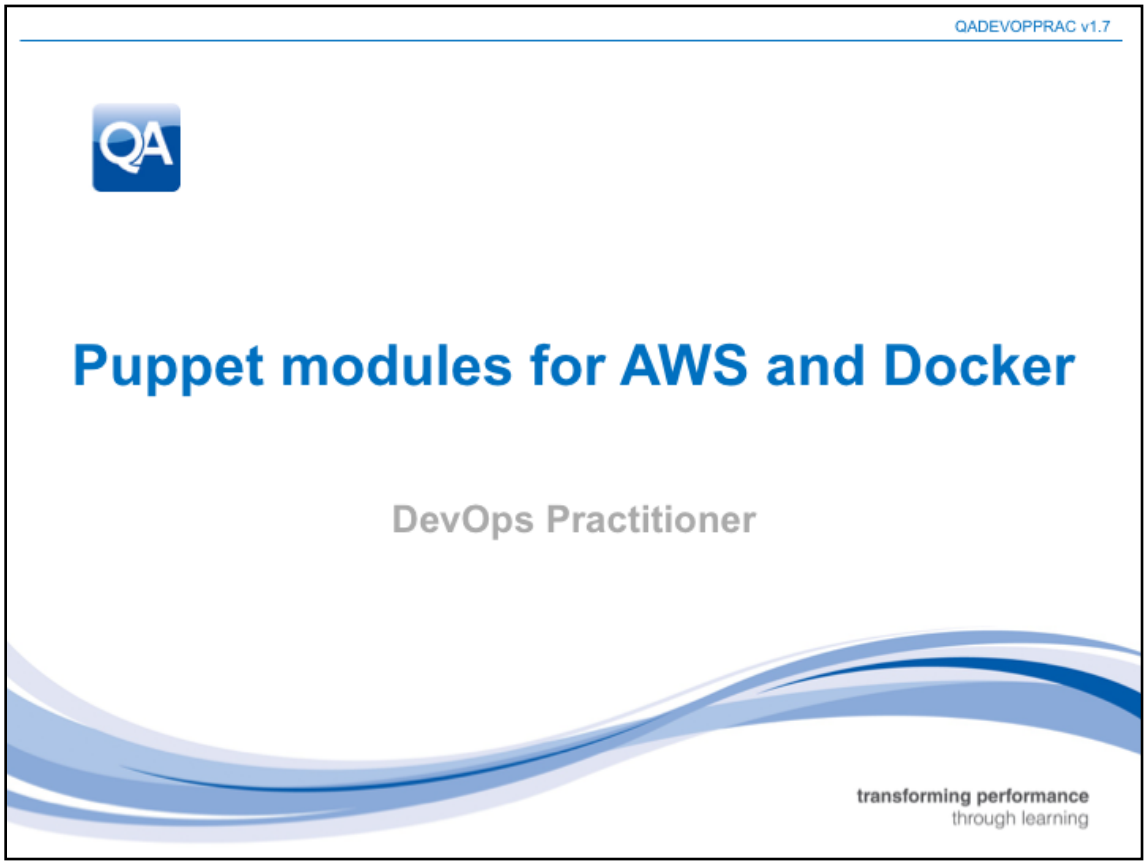
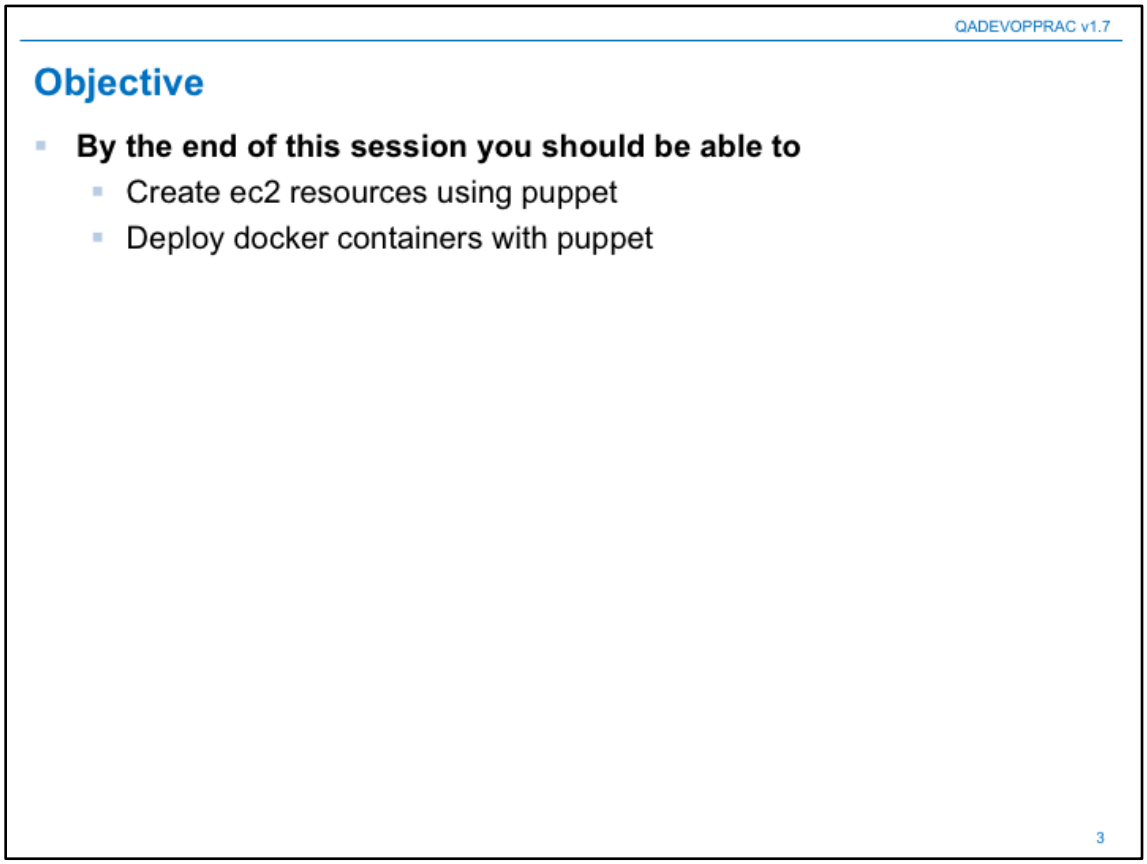
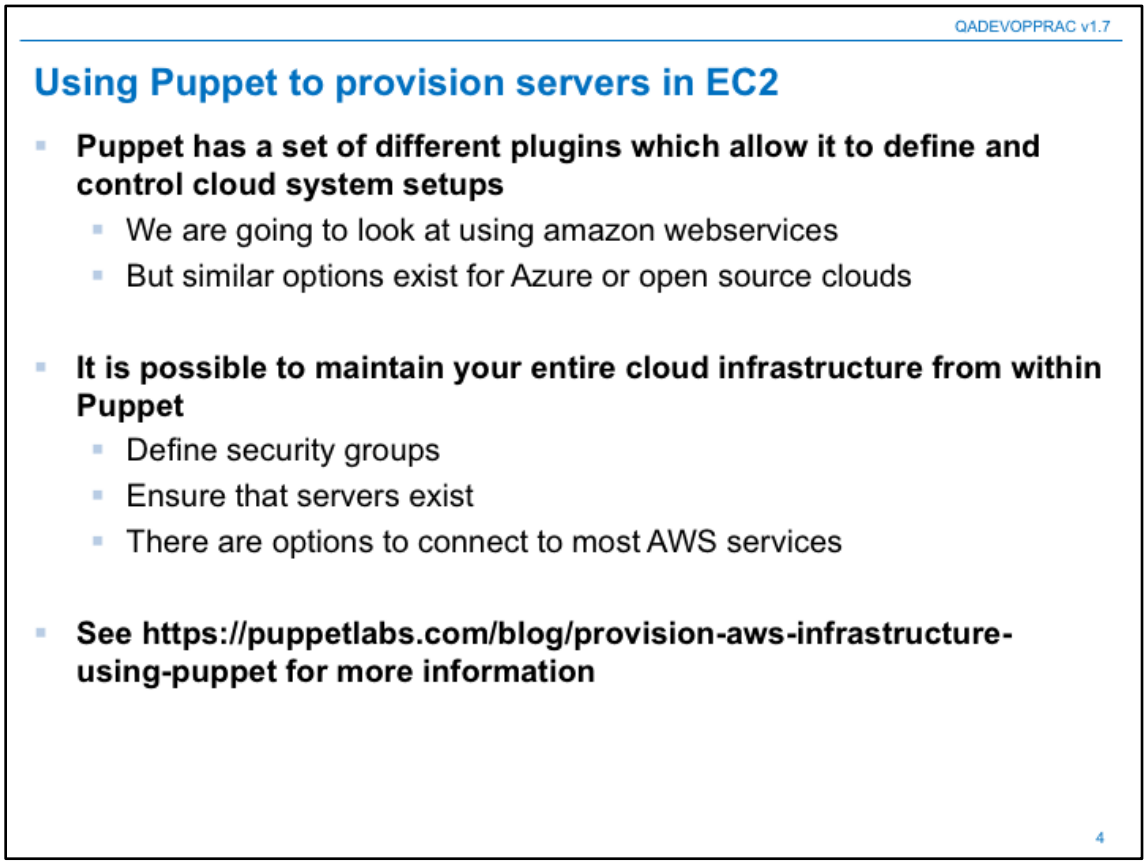

QADEVOPPRAC v1.7
Puppet modules for AWS and Docker
DevOps Practitioner
transforming performance
through learning

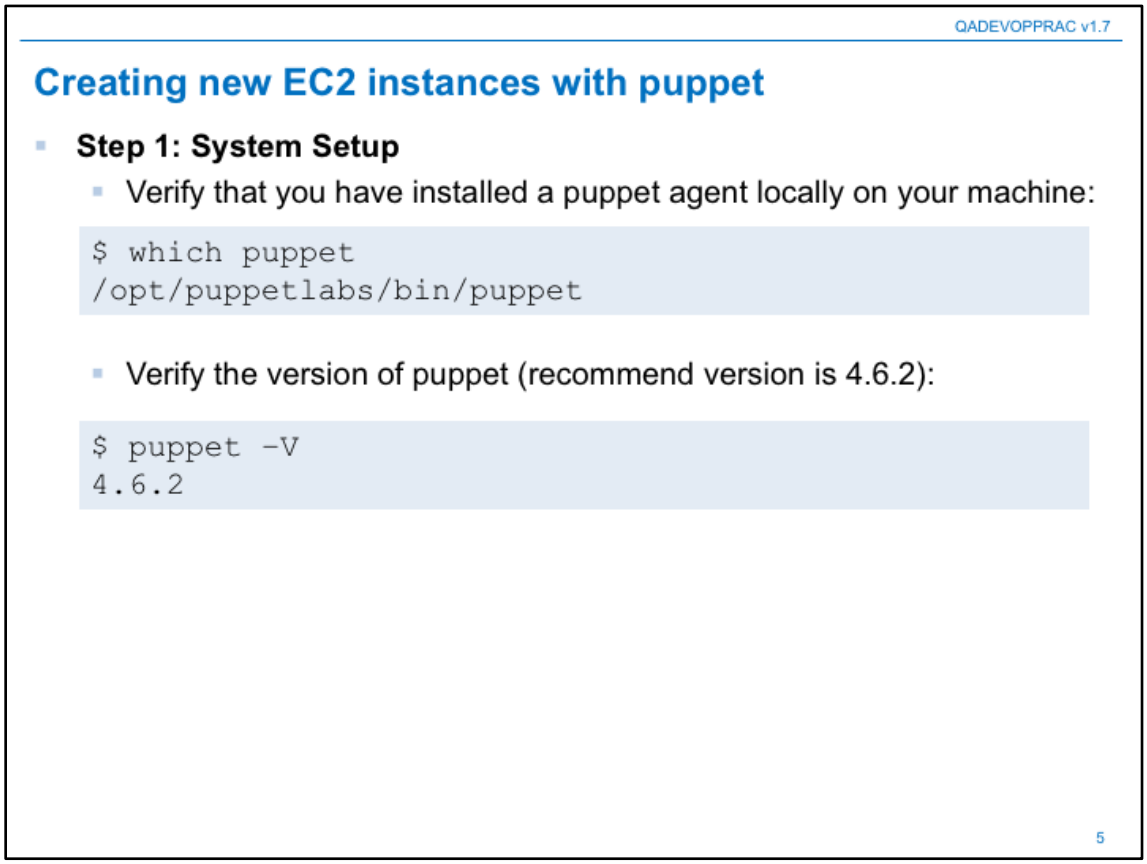
Outline

- **Puppet and EC2**
  - Controlling your infrastructure with Puppet
  - Creating EC2 instances
- **Puppet and Docker**
  - The Docker module
  - Launching Docker containers

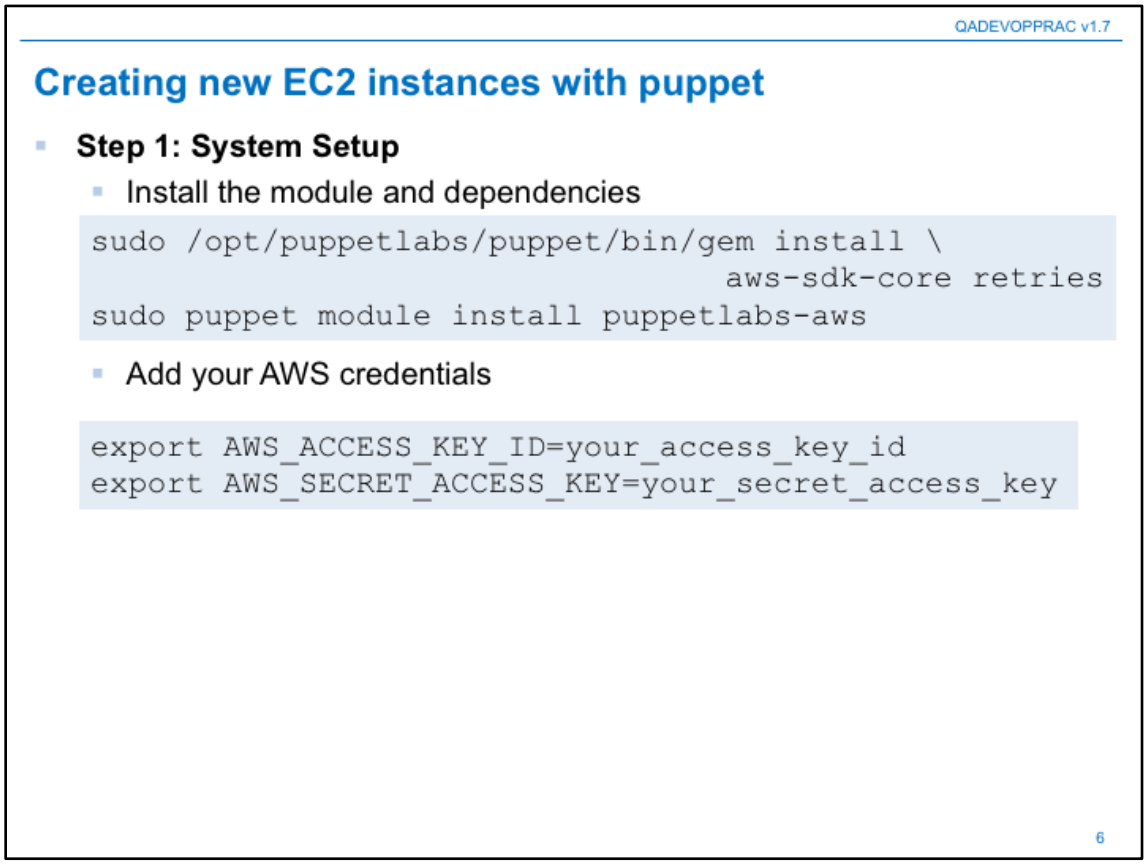

QADEVOPPRAC v1.7
Objective
By the end of this session you should be able to
Create ec2 resources using puppet
Deploy docker containers with puppet
3



https://forge.puppetlabs.com/puppetlabs/aws



https://linuxacademy.com/howtoguides/posts/show/topic/11889-deploying-ec2-resources-with-puppet

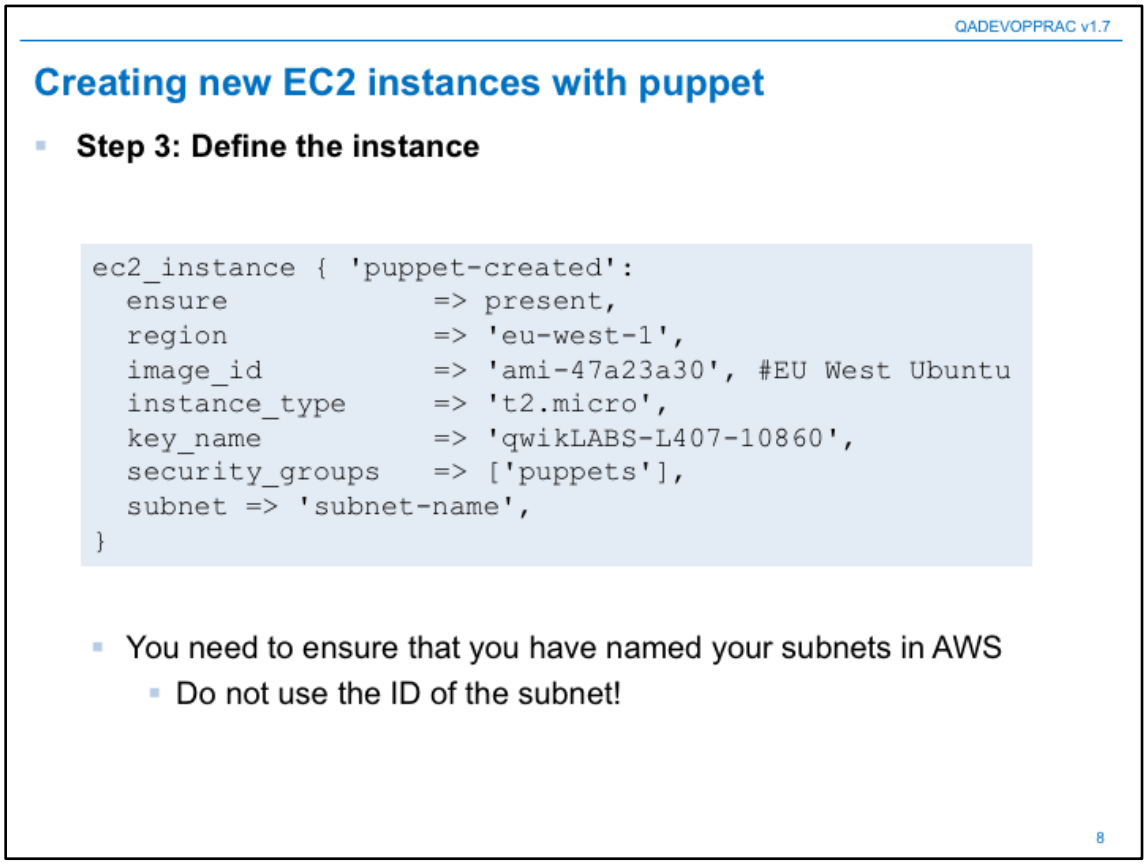



```
export AWS_ACCESS_KEY_ID=lina
export AWS_SECRET_ACCESS_KEY=lina_key
```

Creating new EC2 instances with puppet

- **Step 2: Create a security group**
  - We need port 8080 open to access the hosted services

```
ec2_securitygroup { 'puppets':
  ensure      => present,
  region      => 'eu-west-1',
  description => 'ports required for puppets',
  ingress     => [{
    protocol  => 'tcp',
    port      => 80,
    cidr      => '0.0.0.0/0',
  },{
    protocol  => 'tcp',
    port      => 22,
    cidr      => '0.0.0.0/0',
  },{
    protocol  => 'tcp',
    port      => 8080,
    cidr      => '0.0.0.0/0',
  }],
}
```



You can add user data (a bootstrap script) with the user_data property

```
user_data          => template('agent-setup.sh.erb'),
```

Creating new EC2 instances with puppet

- **Step 4: Tell puppet to run the manifest**
  - Have a look at your dashboard

```
root@ip-172-31-42-219:~/puppetscripts# puppet apply ec2-helloworld.pp
Notice: Compiled catalog for agent1.eu-west-1.compute.internal in environment production in 0.03 seconds
Warning: Ambiguous subnet name '' resolves to subnets subnet-32d32059 (vpc: vpc-3dd32056), subnet-33d32058 (vpc: vpc-3dd32056), subnet-3cd32057 (vpc: vpc-3dd32056) - using subnet-32d32059
Notice: /Stage[main]/Main/Ec2_instance[puppet-created]/ensure: changed absent to running
Notice: Finished catalog run in 14.67 seconds
root@ip-172-31-42-219:~/puppetscripts#
```

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status
puppet-created	i-ac26c314	t2.micro	eu-west-1a	running	Initializing	None
Puppet Agent	i-ae2d1b0f	t2.medium	eu-west-1c	running	2/2 checks...	None
Puppet Master	i-4d2016ec	m3.large	eu-west-1c	running	2/2 checks...	None

9

QADEVOPPRAC v1.7

Puppet and Docker Configs

- **We can add Docker information to the puppet manifest file**
  - There's a module to help

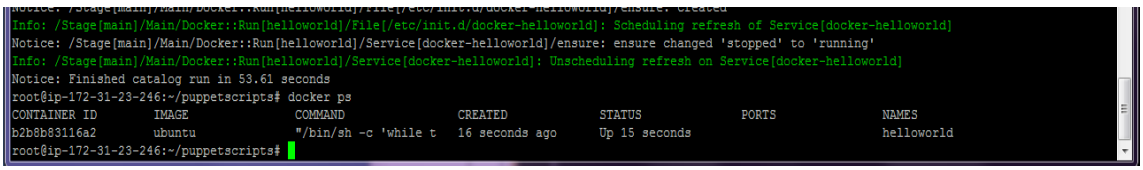
```
$ sudo puppet module install garethr-docker
```

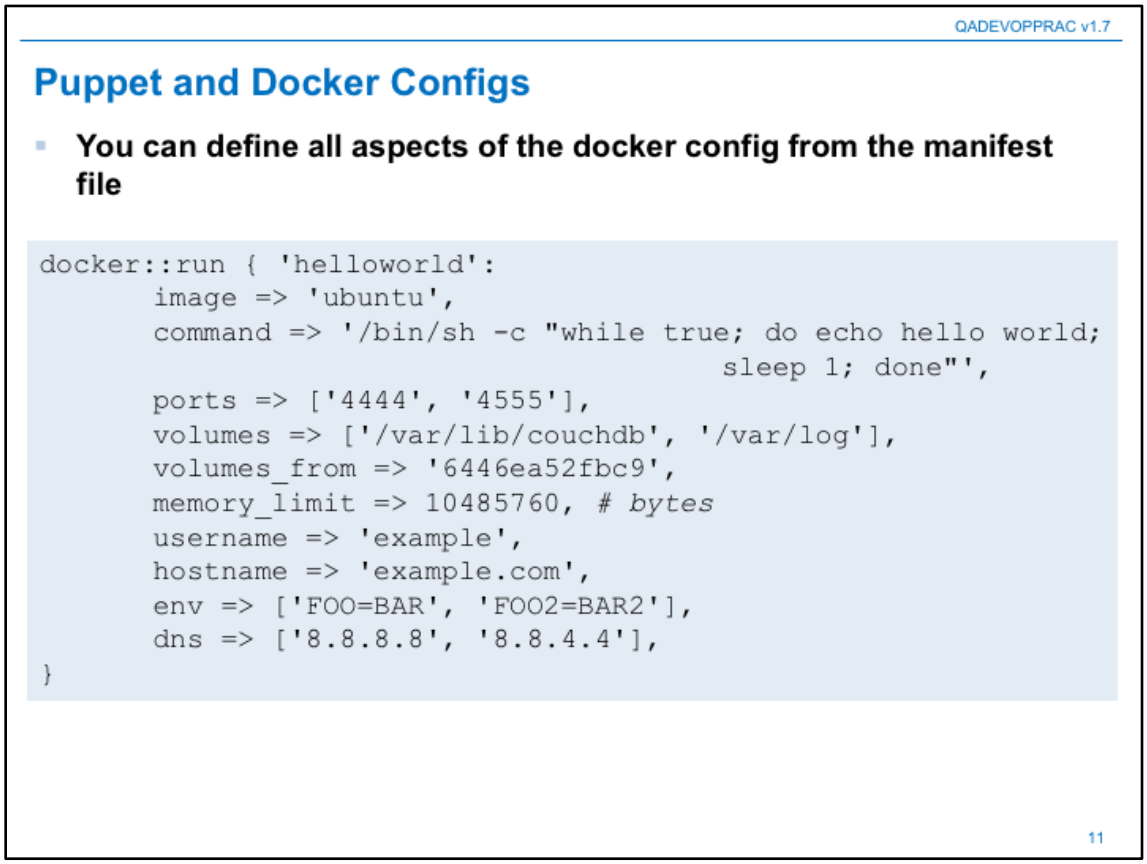
- **Now create the manifest using the docker information:**

```
include 'docker'

docker::image { 'ubuntu':
  ensure => 'absent',
}

docker::run { 'helloworld':
  image   => 'ubuntu',
  command => '/bin/sh -c "while true; do echo hello
                                world; sleep 1; done"',
}
```

When you apply the script you will see the Docker container running:

```
$ puppet apply -t docker-helloworld.pp
```





See: https://puppetlabs.com/blog/simplify-managing-docker-puppet or https://docs.docker.com/articles/puppet/

Exercise

- **Use puppet to automate deploying your docker containers**
  - Create a new module to host some some docker containers
  - Create a new agent
    - (Manually to start with)
  - Add your agent to a group and assign the docker class to it!
- **Stretchers**
  - Look at how you can manage your infrastructure with puppet
  - (And read about whether you should or not)

To read more about Puppet

Documentation:

- https://puppet.com/blog/simplify-managing-docker-puppet
- https://puppet.com/blog/puppet-docker-running-puppet-container-centric-infrastructure
- https://docs.docker.com/engine/admin/puppet/

Summary

- **Puppet and EC2**
  - Controlling your infrastructure with Puppet
  - Creating EC2 instances
- **Puppet and Docker**
  - The Docker module
  - Launching Docker containers