Immersion Day

Getting Started with Windows Server on Amazon EC2

Rev 2015-01-15

Table of Contents

Overview

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change. Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use.

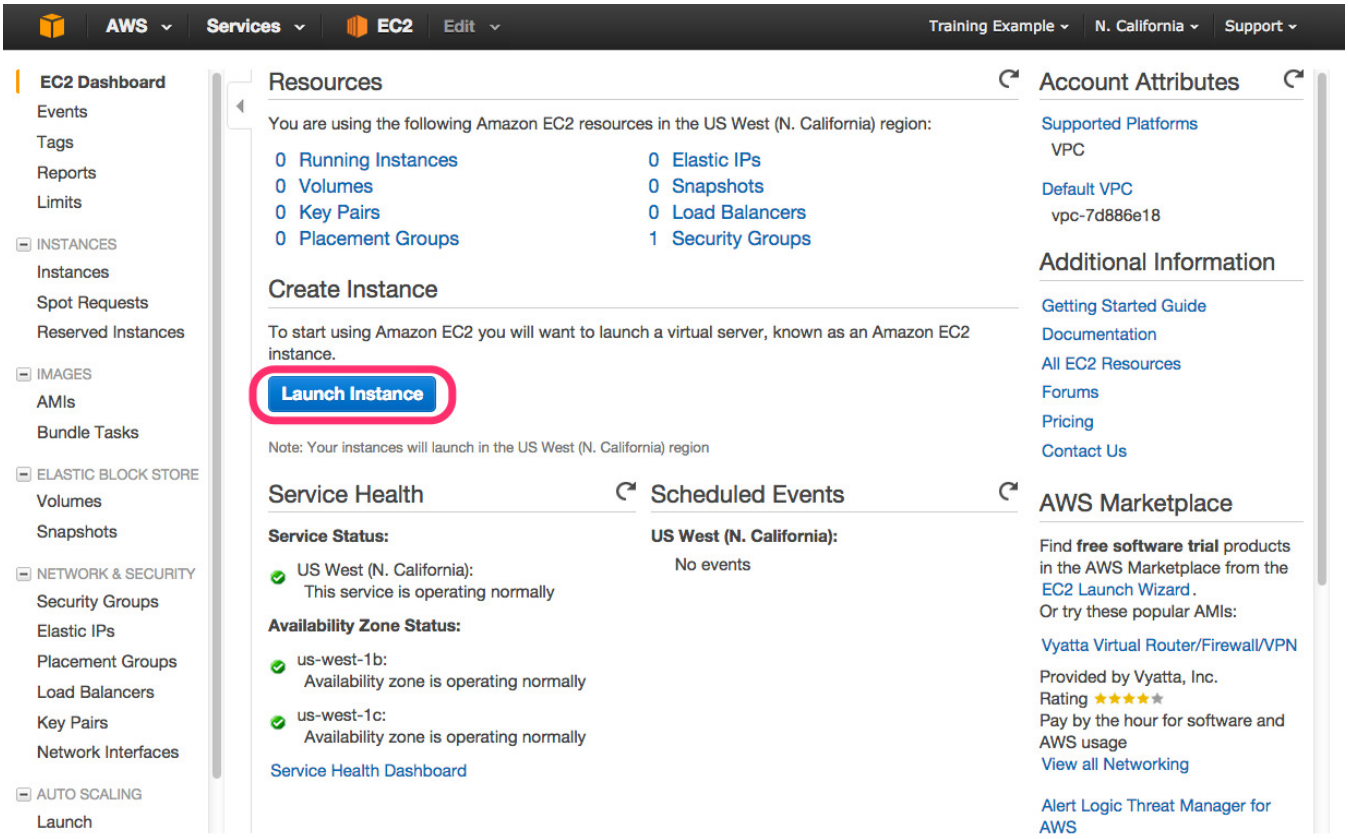
This lab will walk you through launching, configuring, and customizing an EC2 web server using the AWS Management Console.

Launch a Web Server Instance

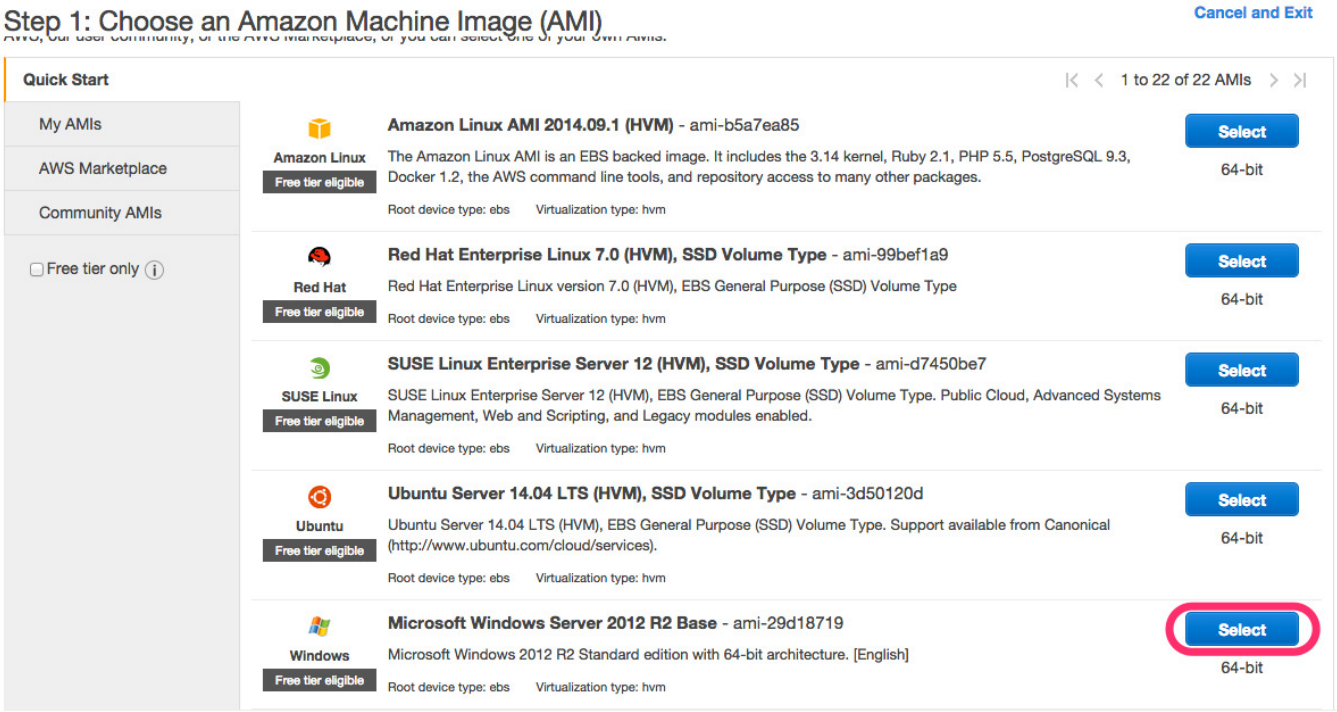
In this example we will launch a Windows Server 2012 instance with the IIS web server installed upon boot.

1. Sign into the AWS Management Console and open the Amazon EC2 console at https://console.aws.amazon.com/ec2.

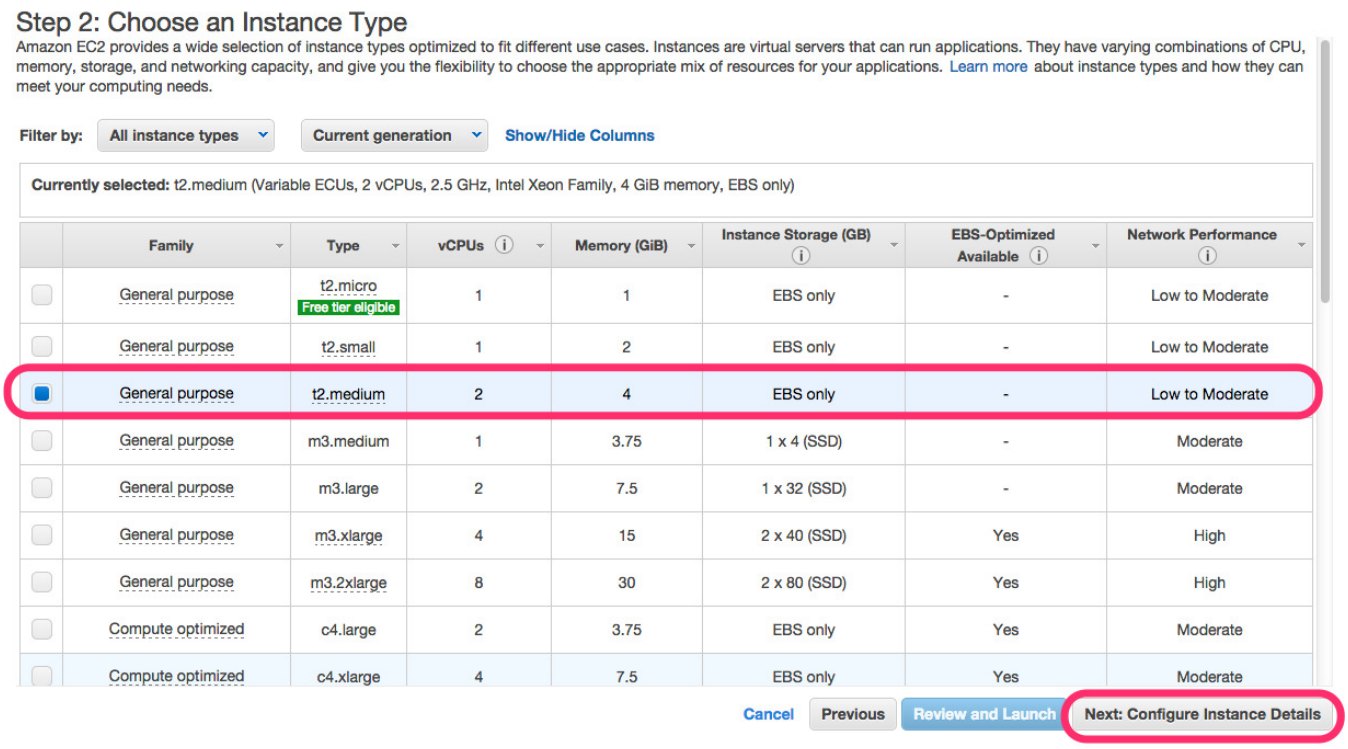
2. Click on **Launch Instance**



3. Scroll down and click **Select** on the Windows Server 2012 R2 Base AMI.



4. In the **Choose Instance Type** tab, select the **t2.medium** instance size and click **Next: Configure Instance Details**



Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

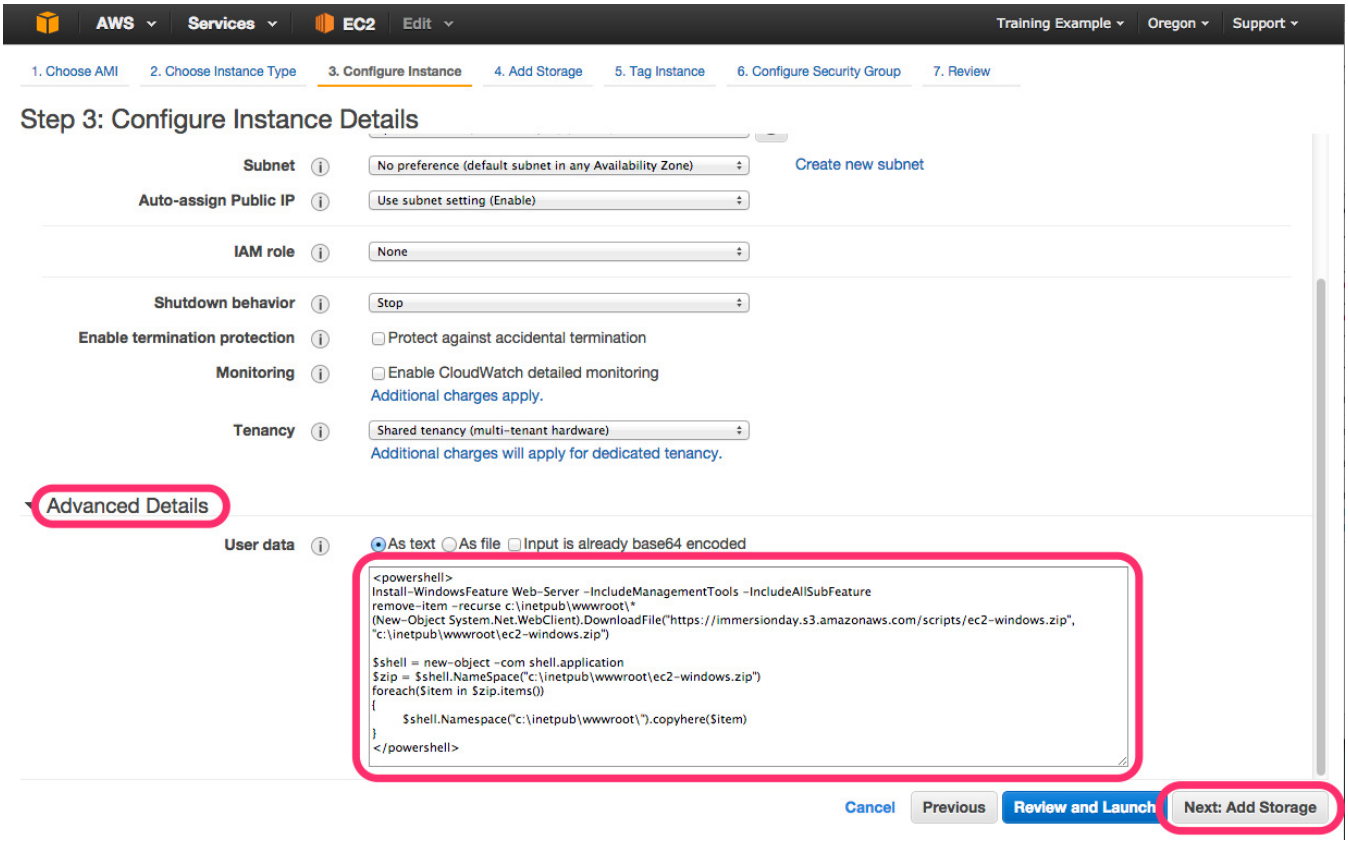
Filter by: All instance types | Current generation | Show/Hide Columns

Currently selected: t2.medium (Variable ECUs, 2 vCPUs, 2.5 GHz, Intel Xeon Family, 4 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
☐	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate
☐	General purpose	t2.small	1	2	EBS only	-	Low to Moderate
☑	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate
☐	General purpose	m3.medium	1	3.75	1 x 4 (SSD)	-	Moderate
☐	General purpose	m3.large	2	7.5	1 x 32 (SSD)	-	Moderate
☐	General purpose	m3.xlarge	4	15	2 x 40 (SSD)	Yes	High
☐	General purpose	m3.2xlarge	8	30	2 x 80 (SSD)	Yes	High
☐	Compute optimized	c4.large	2	3.75	EBS only	Yes	Moderate
☐	Compute optimized	c4.xlarge	4	7.5	EBS only	Yes	Moderate

Cancel | Previous | Review and Launch | Next: Configure Instance Details

5. On the **Configure Instance Details** page, expand the **Advanced Details** section, copy/paste this script (`https://immersionday.s3.amazonaws.com/scripts/ec2-windows-lab-userdata.txt`) into the User Data field (this script will install/start IIS and deploy a simple web page) and click **Next: Add Storage**:

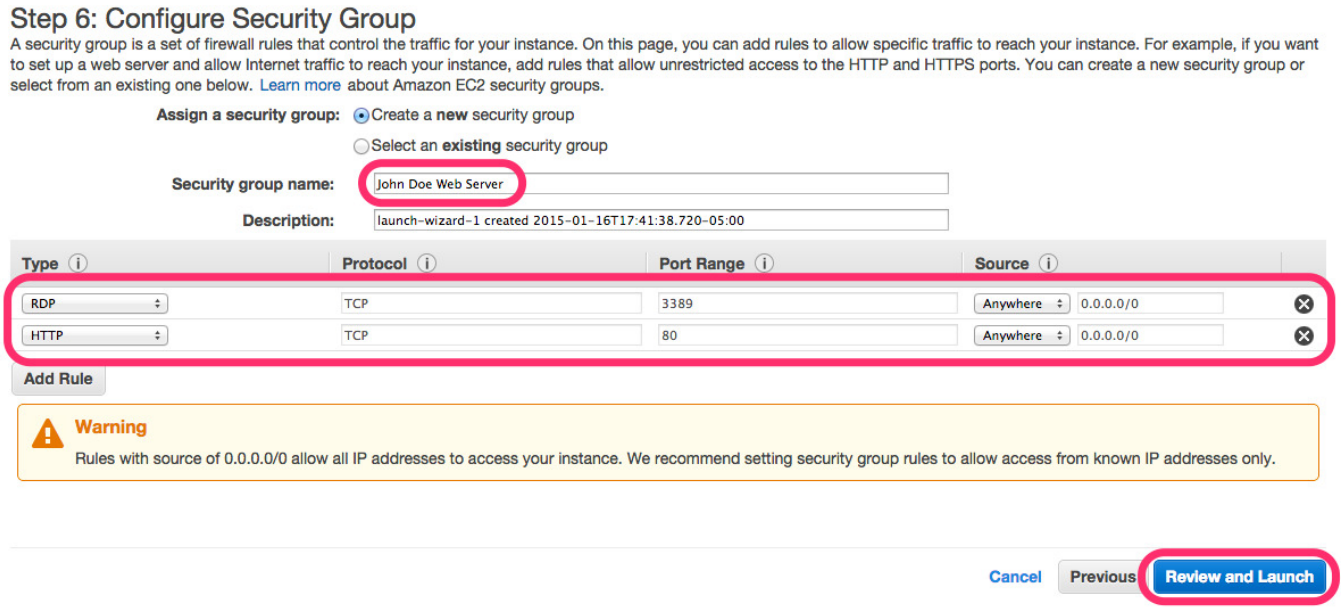


6. On the **Step 4: Add Storage** screen, Click **Next: Tag Instance** to accept the default Storage Device Configuration and move to the Step 5: Tag Instance screen.

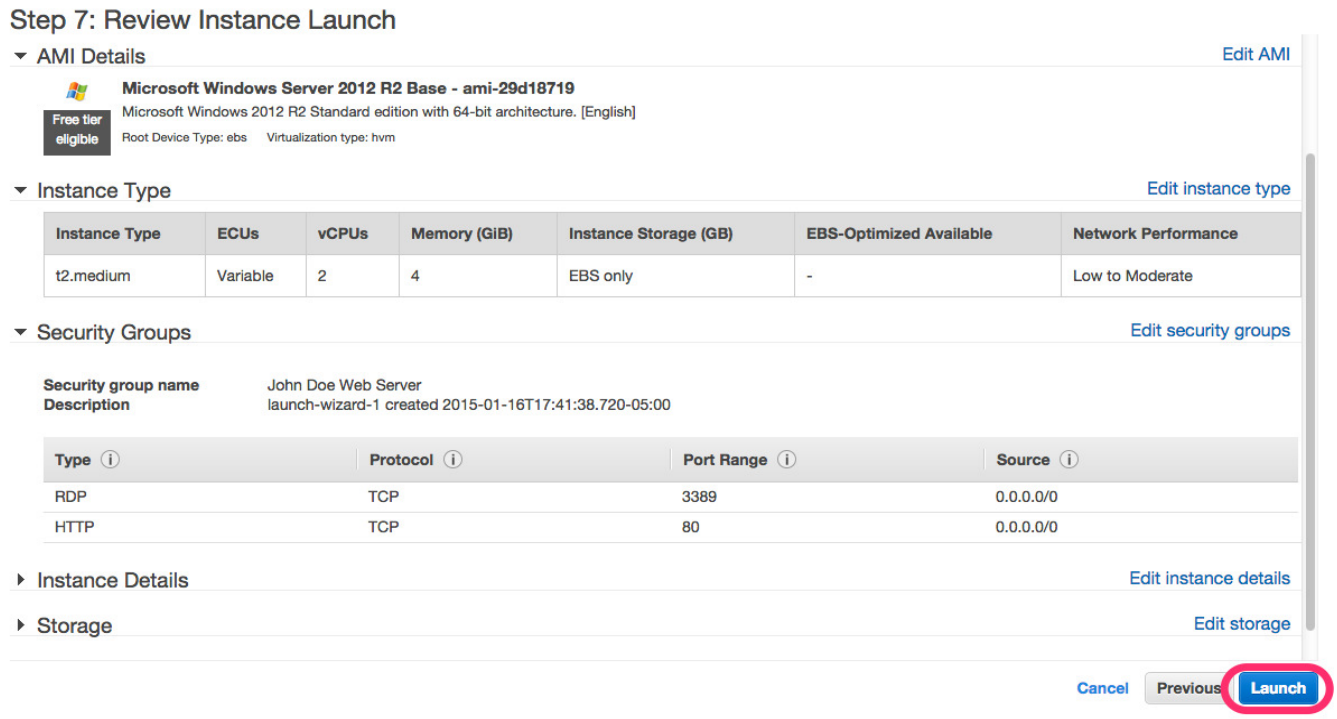
   Next, choose a “friendly name” for your instance. This name, more correctly known as a tag, will appear in the console once the instance launches. It makes it easy to keep track of running machines in a complex environment. Name yours according to this format: “[Your Name] Web Server”.

   Then click **Next: Configure Security Group**.

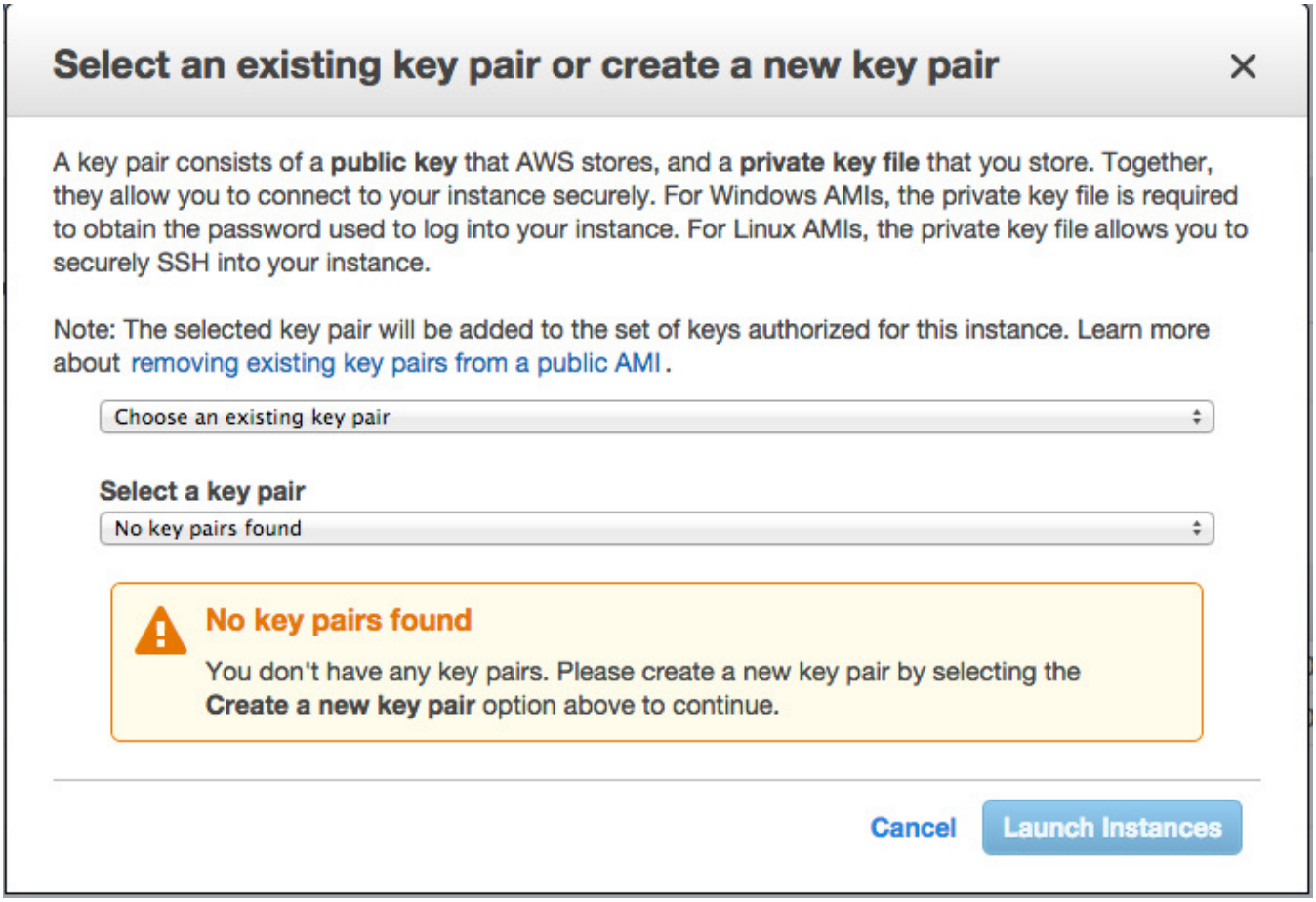
7. You will be prompted to create a new security group, which will be your firewall rules. On the assumption that we are building out a Web server, name this security group according to this format “[Your Name] Web Server”, and open ports 3389 and 80. Click the **Review and Launch** button after configuring the security group.



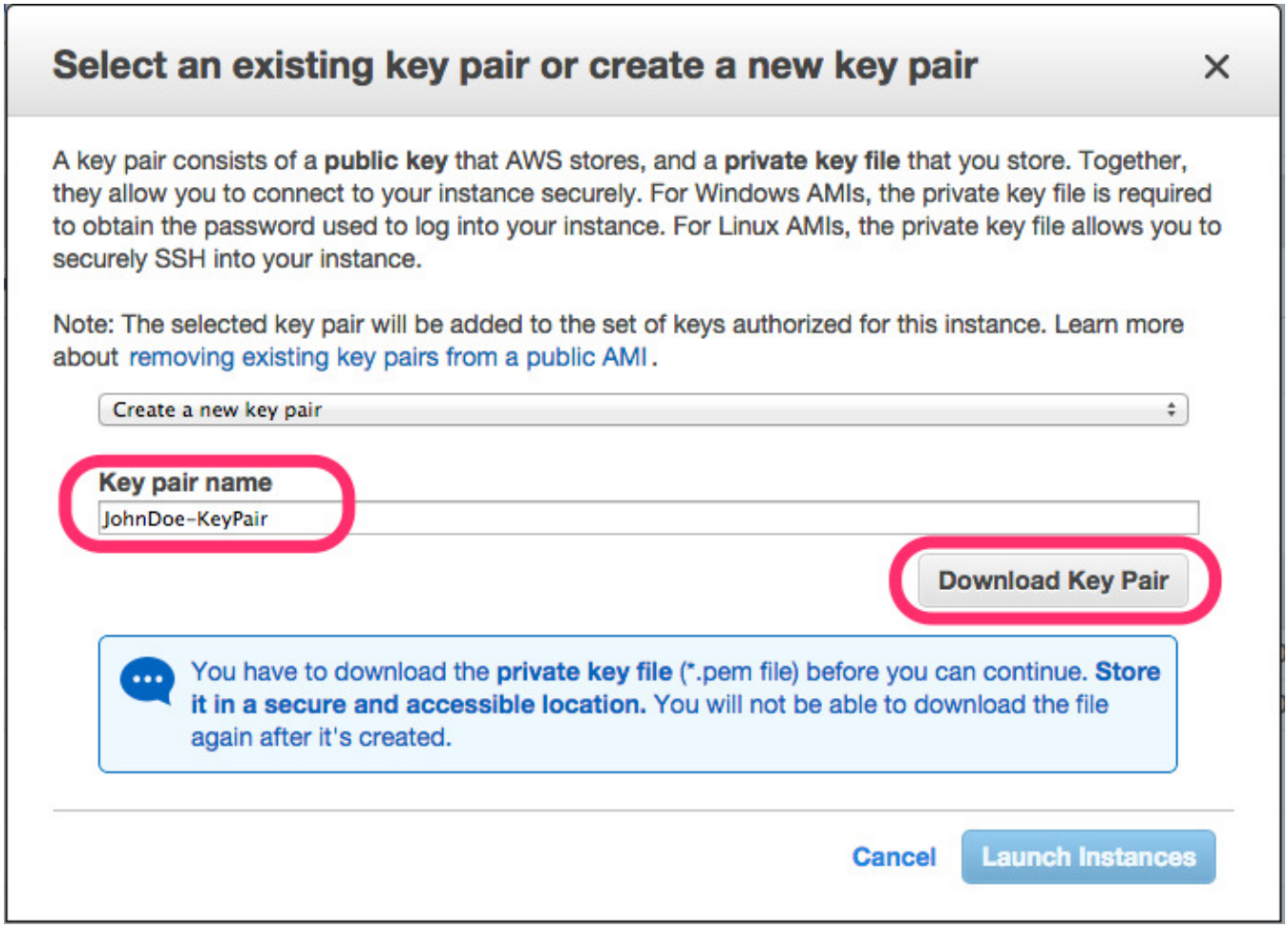
8. Review your choices, and then click **Launch**.



9. Now you need to create a public/private keypair. When this instance launches, you will connect to it via Remote Desktop using the credentials for “administrator”. For Windows instances, EC2 automatically generates a password and encrypts with your public key. To decrypt the encrypted password, you will use your private key. Let’s create a new public/private keypair.



10. Enter a name for the keypair using the following format: [YourName]-KeyPair and click **Download Key Pair**.

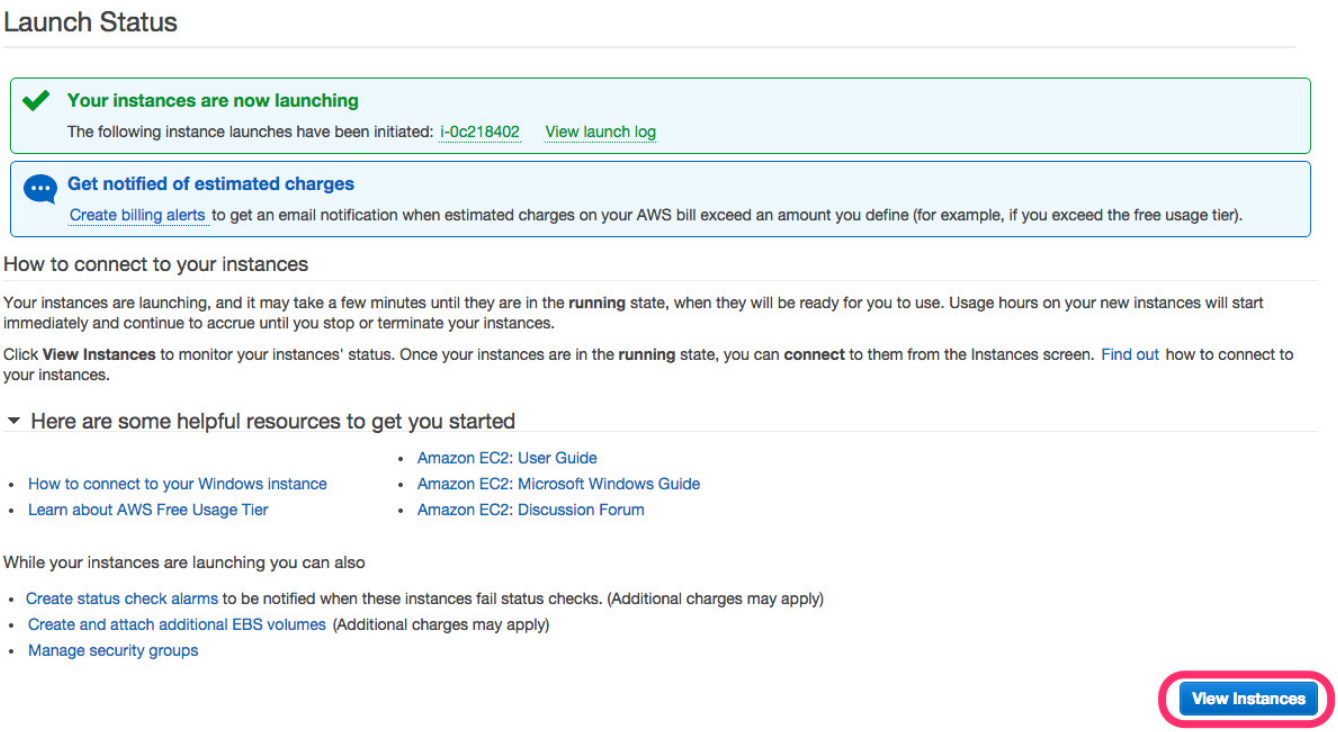


11. Your browser will download the private portion of the keypair to your PC. It will have a name like *JohnDoe-KeyPair.pem*. Note the location of this file because you will need it later to decrypt the administrator password.

12. Now click the Launch Instances button to launch your Windows web server.

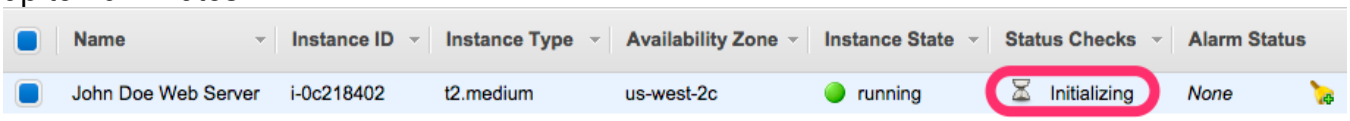
13.

14. The next screen will confirm that your instance is now launching. Click the **View Instances** button. Once your instance has launched, you will see the "[Your Name] Web Server" instance as well as the Availability Zone the instance is in and its publicly routable DNS name.

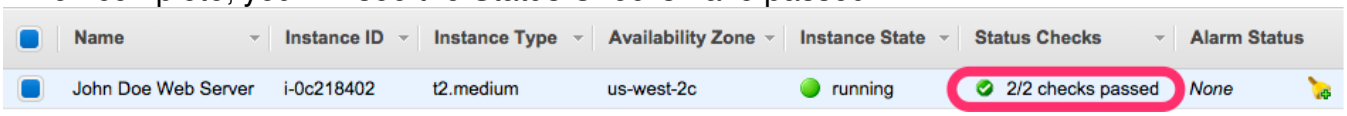


Browse the Web Server

1. Wait for the instance to pass the Status Checks. For Windows instances, this could take up to 20 minutes.



When complete, you will see the Status Checks have passed.



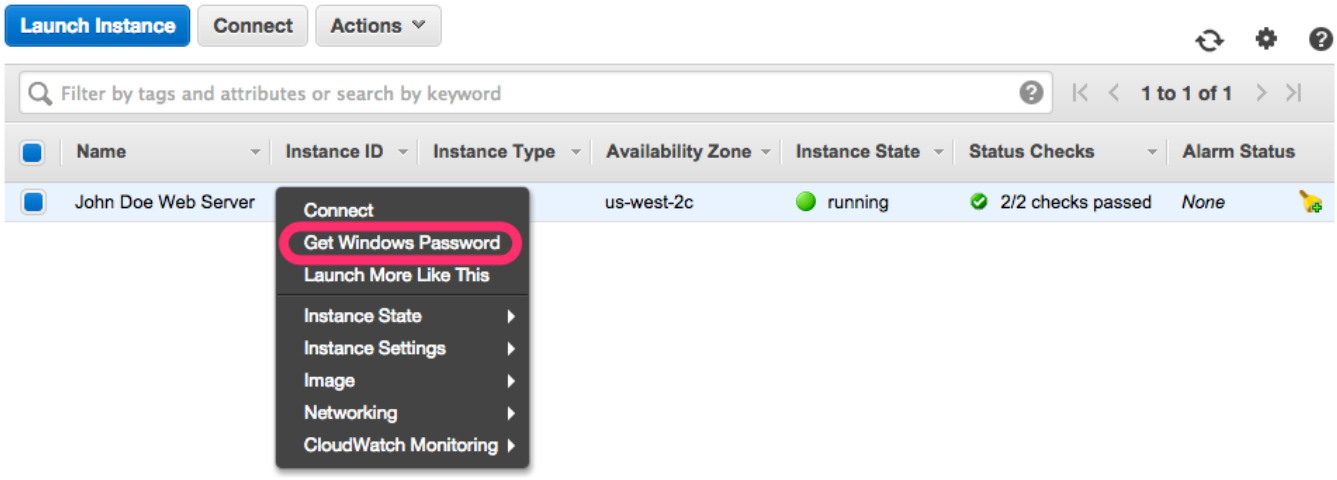
2. Open a new browser tab and browse the Web Server by entering the EC2 instance's Public DNS name into the browser. The EC2 instance's Public DNS name can be found in the console by reviewing the "Public DNS" column. You should see a page that looks similar to this:

Great Job: You have built your first web server!

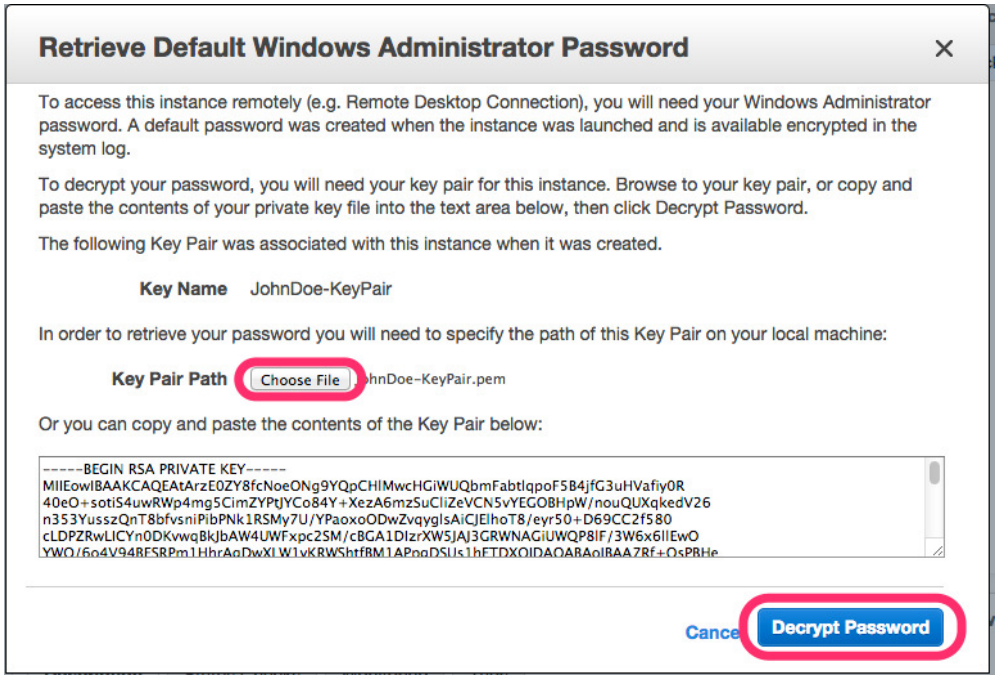
Connecting To Your Windows Instance

To connect to the Windows desktop, we will use a RDP client. If you're using a Windows PC, use the bundled Remote Desktop application. For Mac users, if you don't have a RDP client already installed, download Microsoft Remote Desktop.

1. Retrieve the automatically generated, encrypted Windows password by right clicking your instance and selecting **Get Windows Password**.



2. On the next screen, click the Choose File button and select the private key file that was automatically downloaded earlier when you launched the instance. Then click Decrypt Password to obtain the Administrator password.



3. The decrypted Administrator password should look something like this:

   Note that since only you have the private key, it's important to understand the automatically generated password can only be decrypted by you. So it's important to keep this key secure. Generally, the automatically generated password is changed by the customer after first login. If the automatically generated password is not changed and the private key is lost, there's no way to recover the password.

4. Start your RDP application and connect to the hostname of your instance. The hostname can be found in a couple of different places. For example, in the web console, you'll see hostname listed as the **Public DNS** of the instance.

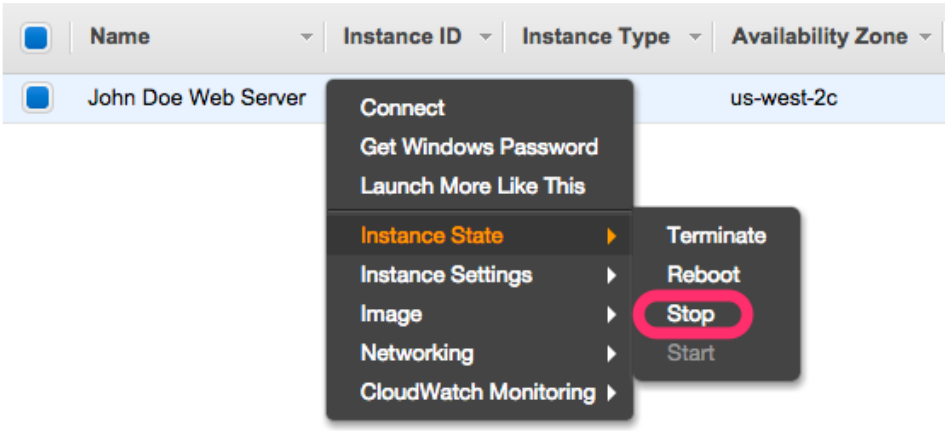
5. In your RDP application, use **Administrator** as the username along with the decrypted password. Once connected, you will have access to the Windows desktop. At this point, feel free to explore Windows. You may want to change the Administrator password to something friendlier or easy to remember (but still secure of course).

Appendix – Additional EC2 Concepts

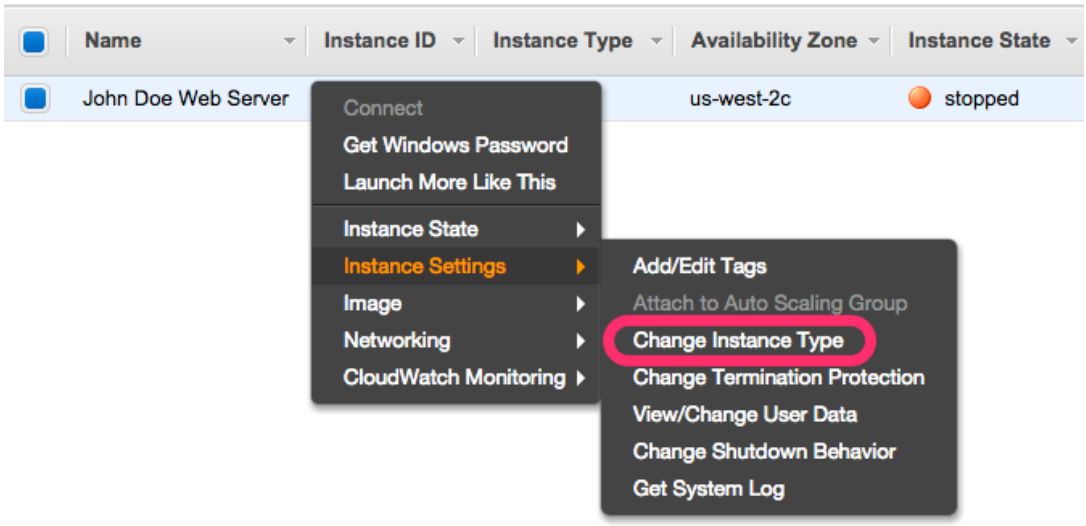
Change the Instance Type

Did you know that you can change the instance type that an AMI is running on? This only works with EBS-backed instances (what we're running here). There is no particular reason to change the instance type in this lab, but the following steps outline how easy it is to do in AWS.

In the AWS Console, select your lab instance, then right-click on it and hover over "Instance State" and select "Stop" (NOT "Terminate"!). Then select "Yes, Stop" to confirm.



After it has stopped, right-click on it again, hover over "Instance Settings" and select "Change Instance Type".



After going through the options and selecting your new instance type, right-click your lab instance and start it again.

Elastic IPs

How do you set up practical DNS names for your web server? Using an address such as http://ec2-75-101-197-112.compute-1.amazonaws.com/ is cumbersome. Setting up a DNS record that points to http://www.yourdomain.com is easy enough – until you reboot the server and the underlying DNS name and IP address both change.

AWS offers Elastic IP Addresses, which are actually NAT addresses that operate at a regional level. That is, an Elastic IP Address works across Availability Zones, within a single region. Assign one to your application as follows:

- Click on the Elastic IPs link in the AWS Console
- Allocate a new address
- Associate the address with a running instance. If you change instances, it's as simple as allocating the address to the new instance.
- Create a DNS "A" record in your own DNS server that points www.yourdomain.com to 75.101.162.40.

Two Important Notes:

1. As long as an Elastic IP address is associated with a running instance, there is no charge for it. However an address that is not associated with a running instance costs $0.01/hour. This prevents address hoarding; however it also means that you need to delete any addresses you create, or you will incur an ongoing charge.
2. Load balancing (covered in the next section) requires CNAME records instead of "A" records. So Elastic IP is not required for load-balanced applications.

Black Belt Booting

There are a number of advanced techniques that offer additional power and flexibility when booting instances. For example, some organizations maintain a series of generic instances, and customize the images upon launch.

Common techniques include:

- Automatically check for updates upon each boot.
- Look in a well-known location, such as in a S3 bucket, for data or a script to tell the instance which packages to load.
- Pass **user data** to the instance to accomplish each of the above, or possibly instead of the other approaches.

How to Pass User Data

The general format looks like this from the command line:

```
aws ec2 run-instances --image-id [ami id] --user-data "user data up to
2048 bytes" ...other params...
```

You can also paste user data into a text field via the AWS Console; however this is usually a form of automation – thus the command line example.