[Home](#) / [AWS](#) / [Guided Lab](#) / Introduction to Amazon Elastic Compute Cloud (EC2)

Introduction to Amazon Elastic Compute Cloud (EC2)

Level: **Fundamental**

Amazon EC2


Amazon Web Services

 Your last attempt on **23-Sep-2025**[View all](#)



Overview

Steps

Validation

 Cloud Architect Compute

Task 1: Sign in to AWS Management Console

 **0h 27m 28s**
left [End Lab](#)[Open Console](#)**Validation****Lab Credentials****User Name** ⓘ

Whiz_User_58170.99117657

Password ⓘ

287b47f2-2052-450f-b

Access Key ⓘ[Help](#)

1. Click on the **Open Console** button, and you will get redirected to AWS Console in a new browser tab.
2. On the AWS sign-in page,
 - Leave the Account ID as default. Never edit/remove the 12 digit Account ID present in the AWS Console. otherwise, you cannot proceed with the lab.
 - Now copy your **User Name** and **Password** in the Lab Console to the **IAM Username and Password** in AWS Console and click on the **Sign in** button.
3. Once Signed In to the AWS Management Console, Make the default AWS Region as **US East (N. Virginia) us-east-1**.

Task 2: Provision Default VPC

1. Navigate to **VPC** by clicking on the **Services** menu in the top, then click on **VPC** or Open the Amazon VPC console via <https://console.aws.amazon.com/vpc/>.
2. Delete the **default VPC** by following the **below steps**:
 - In the navigation pane, choose **Your VPCs**.
 - Select the **VPC** with value as **yes** in **default VPC** column.

Your VPCs (1) Info

Find VPCs by attribute or tag

Last updated 4 minutes ago

Actions

Create VPC

IDR

DHCP option set

Main route table

Main network ACL

Tenancy

Default VPC

Owner ID

[dopt-064356c857fd344...](#)

[rtb-0e0d9f6c4ff8247a9](#)

[acl-02380b13deb1d9499](#)

default

Yes

538376654793

- Go to **actions button** and click on **delete VPC** button.



AKIA4Z6RE4OMEIWQJMMP

Secret Key ⓘ

bJhNp9+kxYhp3INjExwlrkh

Lab Resources —

No Lab Resources Found

Support Documents —

1. [FAQs and Troubleshooting](#)

2. [SSH into EC2 Instance](#)

Need help?

- How to use Hands on Lab
- Troubleshooting Lab
- FAQs

[Submit Feedback](#)[Share](#)

The screenshot shows the 'Your VPCs (1/1)' page in the AWS Management Console. A table lists the VPCs with columns for Name, VPC ID, State, Block Public Access, IPv4 CIDR, and IPv6 CIDR. The first VPC is 'vpc-05ebab04adb266a4e' with state 'Available'. The 'Actions' dropdown menu is open, showing options like 'Create default VPC', 'Create flow log', 'Edit VPC settings', 'Edit CIDRs', 'Manage middlebox routes', 'Manage tags', and 'Delete VPC'. The 'Delete VPC' option is highlighted with a red box. The 'Create VPC' button is also visible in the top right corner.

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
-	vpc-05ebab04adb266a4e	Available	Off	172.31.0.0/16	-

- Check **I acknowledge that I want to delete my default VPC** option.
- Type default delete VPC and click on **Delete button**.



Delete VPC

✔ Will be deleted

This VPC will be deleted permanently and cannot be recovered later:

Name	VPC ID	State
Default VPC	vpc-d7d7b2aa	✔ Available

✔ Will also be deleted

The following 7 resources will also be deleted permanently and cannot be recovered later:

< 1 2 >

Name	Resource ID	State
-	igw-a3864dd9	✔ Availab
-	subnet-79633626	✔ Availab
-	subnet-1de7b03c	✔ Availab
-	subnet-00158931	✔ Availab
-	subnet-f1292bff	✔ Availab

⚠ Warning:

If you delete this default VPC, you can't launch instances in this Region unless you specify a subnet in another VPC or create a new default VPC.

☒

I acknowledge that I want to delete my default VPC.

To confirm deletion, type *delete default vpc* in the field:

Cancel

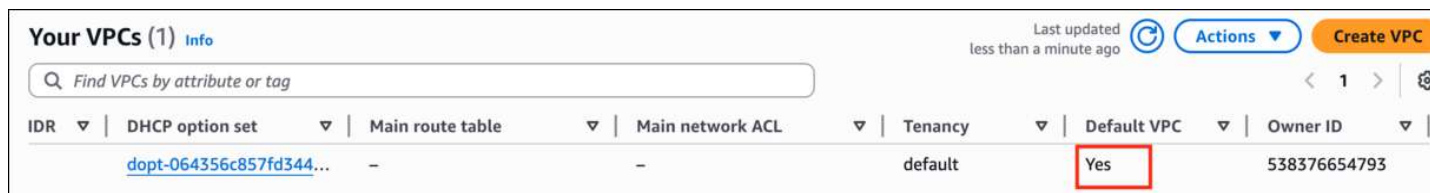
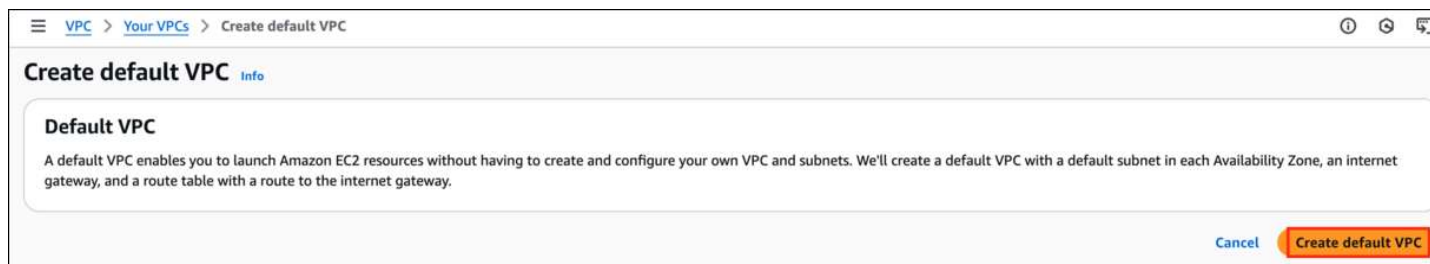
Delete



3. Now to provision **Default VPC** again, Refresh your console go to **actions** and click **Create default VPC**



4. Click **Create default VPC button** and your default VPC will get created



Task 3 : Launch an EC2 Instance with desired specifications



1. Ensure you are in the **US East (N. Virginia) us-east-1** Region to begin launching an EC2 instance in the Amazon cloud.
2. Navigate to **EC2** by clicking on the **Services** menu in the top, then click on **EC2** in the **Compute** section.
3. Click on the **Instances** option on the left panel, and then click on the **Launch Instances** button.

4. Name : Enter **MyEC2Server**

Name and tags [Info](#)

Name

MyEC2Server

Add additional tags

5. Select **Amazon Linux 2023 AMI** from the dropdown.

Recents

My AMIs

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Linux

SUSE

Debian

debian

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI

ami-0150ccaf51ab55a51 (64-bit (x86), uefi-preferred) / ami-0cd4eb0ae8def650 (64-bit (Arm), uefi)

Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2023 (kernel-6.1) is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

Amazon Linux 2023 AMI 2023.8.20250707.0 x86_64 HVM kernel-6.1

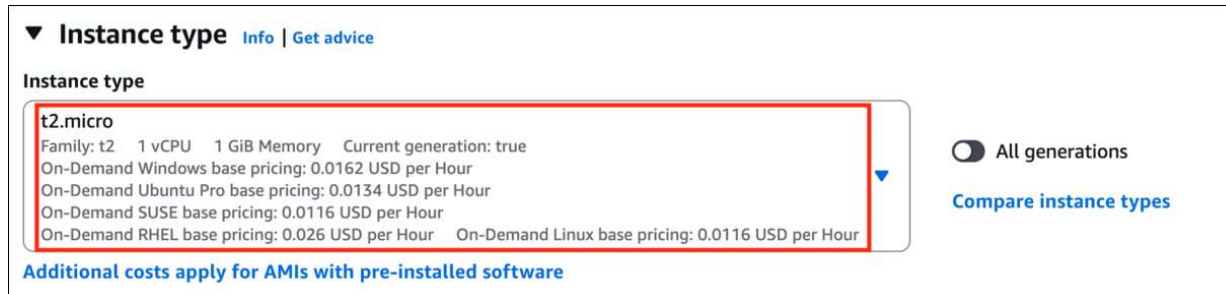
Architecture	Boot mode	AMI ID	Publish Date	Username	
64-bit (x86)	uefi-preferred	ami-0150ccaf51ab55a51	2025-07-08	ec2-user	<div>Verified provider</div>



Note: if there are two AMI's present for **Amazon Linux 2023 kernel-6.1 AMI**.

6. An instance type in AWS refers to a virtual server configuration that determines the computing resources, such as CPU, memory, and storage, available to an instance. It is the basic building block for creating an EC2 instance in the AWS cloud.

- For **Instance Type**: Select **t2.micro**



- t2.micro is an instance type in AWS that comes with 1 vCPU, and 1GB memory and is suitable for low-traffic web servers, small development environments, and other lightweight applications.

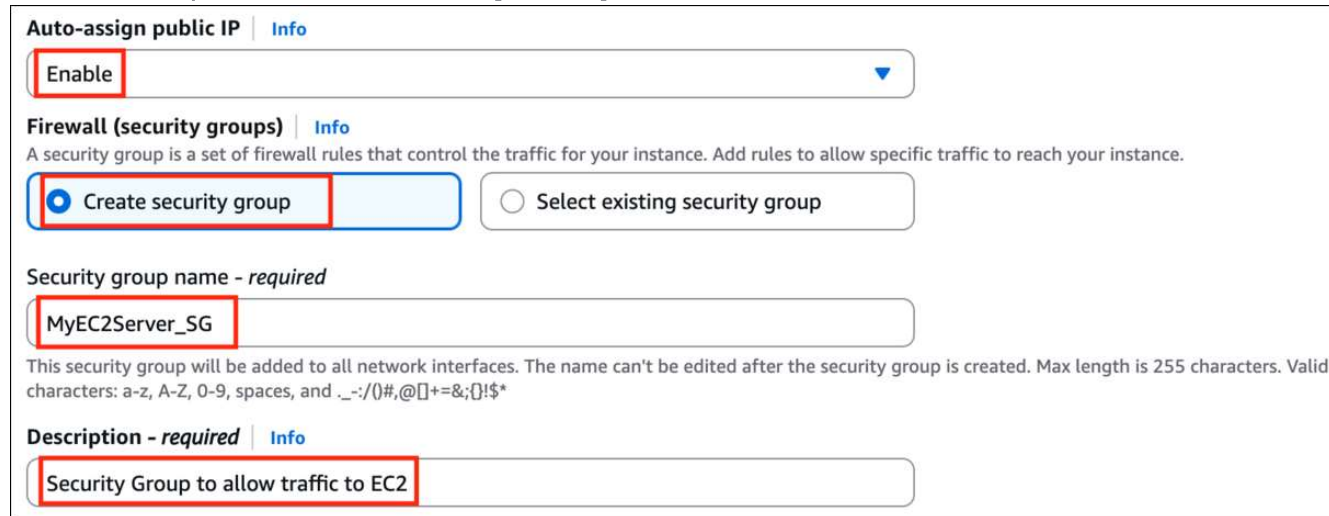
7. AWS key pair is a secure pair of keys used for login and access to EC2 instances. It includes a public key placed on the instance and a private key kept on the user's local computer, used for authentication to prevent unauthorized access.

- For **Key pair(login)**: Select **Create a new key pair** Button
 - Key pair name: **WhizKey**
 - Key pair type: **RSA**
 - Private key file format: **.pem**



8. In **Network Settings**, Click on **Edit** Button:

- Auto-assign public IP: **Enable**
- Select **Create security group**
- Security group name: Enter **MyEC2Server_SG**
- Description: Enter **Security Group to allow traffic to EC2.**



Auto-assign public IP [Info](#)

Enable

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group ☐ Select existing security group

Security group name - required

MyEC2Server_SG

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and . _ - / () # , @ [] + = & ; ' ! \$ *

Description - required [Info](#)

Security Group to allow traffic to EC2

9. We will now add the security group rules. SSH will already be present there.

- For HTTP, Select **Add security group rule** Button
 - Choose Type: Select **HTTP**
 - Source: Select **Anywhere**



Security group name - required

MyEC2Server_SG

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-:/()#,@[]+=&;{}!\$*

Description - required [Info](#)

Security Group to allow traffic to EC2

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0) [Remove](#)

Type	Protocol	Port range	Source type	Source	Description - optional
ssh	TCP	22	Anywhere	0.0.0.0/0	e.g. SSH for admin desktop

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0) [Remove](#)

Type	Protocol	Port range	Source type	Source	Description - optional
HTTP	TCP	80	Anywhere	0.0.0.0/0	e.g. SSH for admin desktop

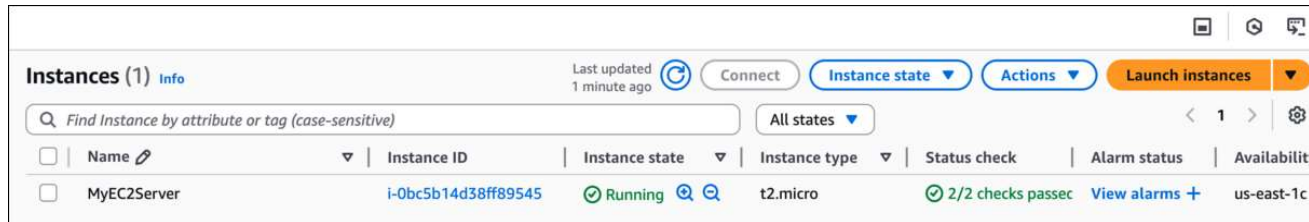
10. A security group is a virtual firewall that controls the inbound and outbound traffic for instances in a particular network in a cloud computing environment. Here we have selected SSH and HTTP rules that will allow incoming SSH and HTTP traffic to instances that are associated with the security group.



11. Proceed with launching the instance while leaving all other settings as default. Simply click on the **Launch Instance** without modifying any other configuration.


12. To view the instance that you have created, choose the **View all Instances** option.

13. **Launch Status:** Once you have initiated the instance launch process, Go to the Instances page from the left menu and wait for your EC2 instance to become "**Running**" while ensuring the health check status is **2/2 checks passed** for optimal performance.



	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
<input type="checkbox"/>	MyEC2Server	i-0bc5b14d38ff89545	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1c

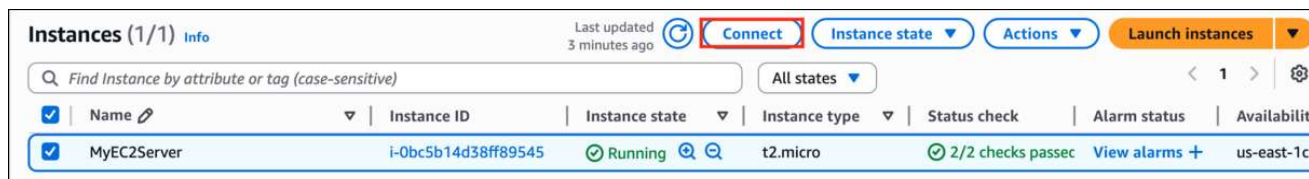
14. **Select** the instance that you have created and **copy** the public IPv4 address within the details section and **paste** it into the editor for later use. An example of this process is depicted in the screenshot provided.



Details		Status and alarms	Monitoring	Security	Networking	Storage	Tags
▼ Instance summary							
Instance ID	i-0bc5b14d38ff89545			Public IPv4 address	34.238.247.133 open address		
IPv6 address	-			Instance state	Running		
				Private IPv4 addresses	172.31.90.120		
				Public DNS	ec2-34-238-247-133.compute-1.amazonaws.com open address		

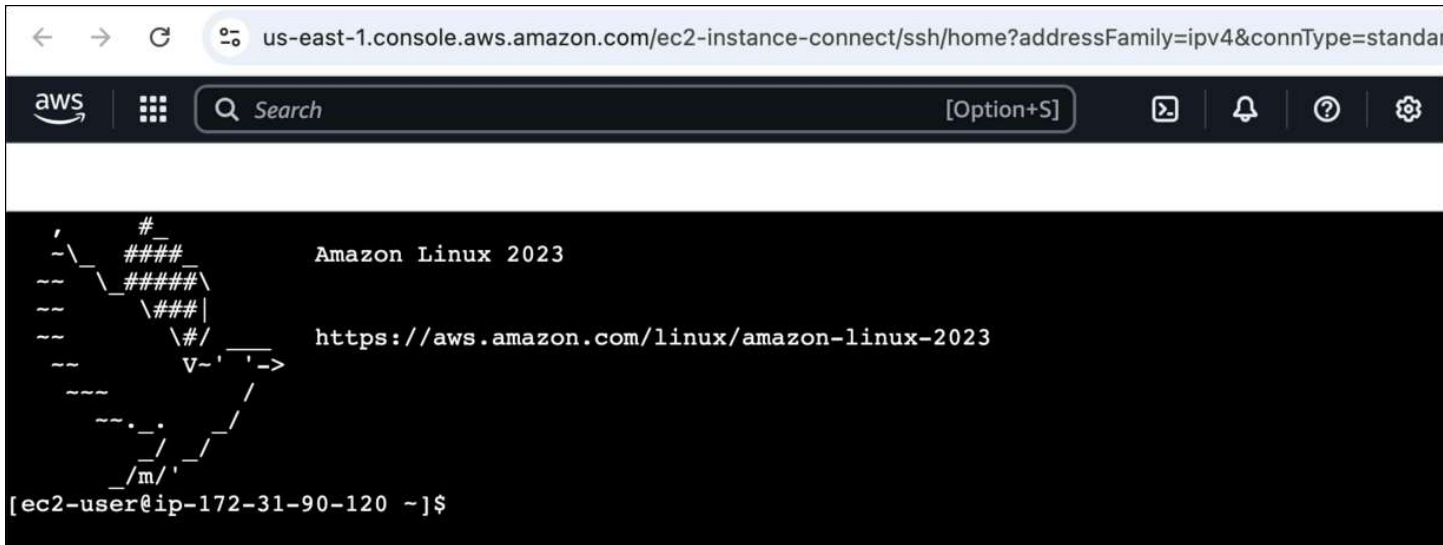
Task 4 : SSH into EC2 Instance using the key pair

1. Select your EC2 instance (**MyEC2Server**) and click on the **Connect** button.

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
<input checked="" type="checkbox"/>	MyEC2Server	i-0bc5b14d38ff89545	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1c

2. Select **EC2 Instance Connect** option and click on **Connect** button. Keep everything else as default.
3. A new tab will open in the browser where you can execute the Linux Commands.
4. Please follow the steps in [SSH into EC2 Instance](#) for more options to SSH.



```
us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?addressFamily=ipv4&connType=standar

aws Search [Option+S]

Amazon Linux 2023

https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-172-31-90-120 ~]$
```

Task 5: Install an Apache Server on the instance

In this task, our goal is to configure an Amazon EC2 instance to run an Apache Web Server and verify its functionality by accessing the web server via a web browser using the instance's public IPv4 address.

1. Switch to root user:



2. Now run the updates using the following command:

```
dnf update -y
```



3. Once completed, lets install and run an apache server

4. Install the Apache web server:

```
dnf install httpd -y
```



5. Start the web server:

```
systemctl start httpd
```



6. Now Enable httpd:

```
systemctl enable httpd
```



7. Check the webserver status

```
systemctl status httpd
```



8. You can see Active status is running.

9. You can test that your web server is properly installed and started by entering the **public IPv4 address** of your **EC2 instance** in the address bar of a web browser. If your web server is running, then you see the Apache test page. If you don't see the Apache test page, then verify whether you followed the above steps properly and check your inbound rules for the security group that you created.

Task 6 : Create a web page and publish it on the instance

In this task, you will add content to the index.html file using the "echo" command and restart the webserver. Then, you can view the content by entering the public IPv4 address followed by "/index.html" in a web browser, ensuring that the URL protocol is HTTP.

1. To add the contents into index.html file using echo, copy and paste the below command to shell.

```
echo "<html>Hi Whizlabs, I am a public page</html>" > /var/www/html/index.html
```



2. Restart the webserver by using the following command:

```
systemctl restart httpd
```



3. Now enter the file name, **/index.html** after the **public IPv4 Address** which you got when you created the ec2 instance in the browser, and you can see your HTML content.



- Make sure **URL Protocol** is **http** not https.
- Syntax: **http://<Your_Public_IPv4_Address>/index.html**

- Sample URL: **http://52.87.50.168/index.html**
- **Note:** If the index.html page is not loading, try removing **s** from the link, it should be HTTP.



4. If you can see the above text in the browser, then you have successfully completed the lab.

Do You Know?

The **Amazon EC2 P4d instance** is the most powerful EC2 instance type available, with eight NVIDIA A100 Tensor Core GPUs, 1.1 TB of NVMe storage, and 400 Gbps network bandwidth, making it ideal for large-scale machine learning and HPC workloads.

Task 7 : Validation of the Lab



1. Once the lab steps are completed, please click on the **Validation** button on the left side panel.



2. This will validate the resources in the AWS account and displays whether you have completed this lab successfully or not.

3. Sample output :

Check your Validation

If any checks fail , you can use the remaining time in the Lab to work on making the checks pass . Click Validate My Lab again to rerun the checks at any time.

Validate My Lab 

Launch an Amazon EC2 Instance

Check whether an Amazon Linux 2023 AMI Instance is created or not.

Install Apache Web Server

Check whether Apache Web Server is installed and index.html file is added in EC2 Instance or not.

Completion and Conclusion

1. You have successfully created and launched an Amazon EC2 instance
2. You have successfully logged into the EC2 instance using SSH
3. You have successfully installed an Apache server on the instance
4. You have successfully created a webpage and published it.
5. You have successfully validated the lab.



End Lab

- 1. Sign out of AWS Account.
- 2. You have successfully completed the lab.
- 3. Once you have completed the steps, click on **End Lab** from your Whizlabs lab console and wait till the process gets completed.



Categories	Popular Courses	Company	Legal	Support
Training Library	AWS Certified Solutions Architect Associate	About Us	Privacy Policy	Contact Us
Cloud Computing	AWS Certified Cloud Practitioner	Blog	Terms of Use	FAQs
Certifications	Microsoft Azure Exam AZ-204 Certification	Reviews	EULA	
Amazon Web Services (AWS)	Microsoft Azure Exam AZ-900 Certification	Careers	Refund Policy	
Microsoft Azure	Google Cloud Certified Associate Cloud Engineer	Team Account	Programs Guarantee	
Google Cloud	Microsoft Power Platform Fundamentals (PL-900)			
DevOps				
 Security				
Microsoft Power Platform				
Microsoft 365				
Certifications				

Java Certifications

HashiCorp Certified
Terraform Associate
Certification

Snowflake SnowPro
Advanced Architect
Certification

Docker Certified
Associate

Need help? You can  or  **+91 6364678444**

©2025, Whizlabs Software Pvt. Ltd. All rights reserved.

