Deployment #5

Task: Follow each step of this deployment lab to accomplish your deployment 5. Create a secure Jenkins architecture with your current VPC you made during class.

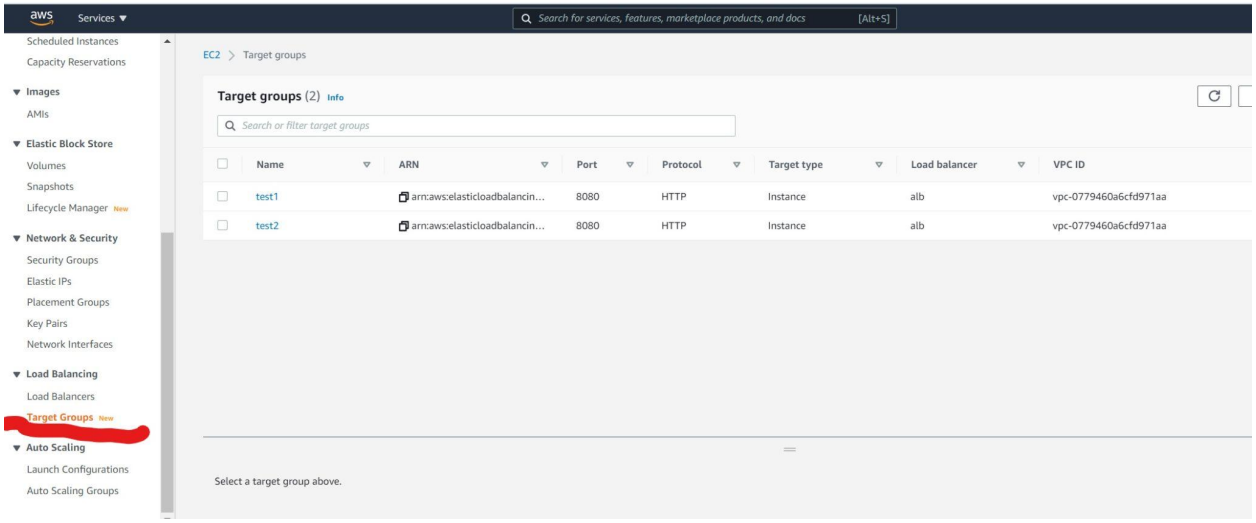
Step 1: Configure Jenkins on the EC2 in the private subnet.

Tips: AWS changed the way you'll need to download and install Jenkins on the EC2. You will need to download java-openjdk11 and epel from {amazon-linux-extras}. Then add Jenkins to the local repository, add the key to the repository and finally install Jenkins:

```
sudo amazon-linux-extras install java-openjdk11
sudo amazon-linux-extras install epel
sudo wget -O /etc/yum.repos.d/jenkins.repo \
https://pkg.jenkins.io/redhat-stable/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
sudo yum upgrade
sudo yum install epel-release java-11-openjdk-devel
sudo yum install jenkins
sudo systemctl start jenkins
```

Step 2: Create a Target Group and Load Balancer

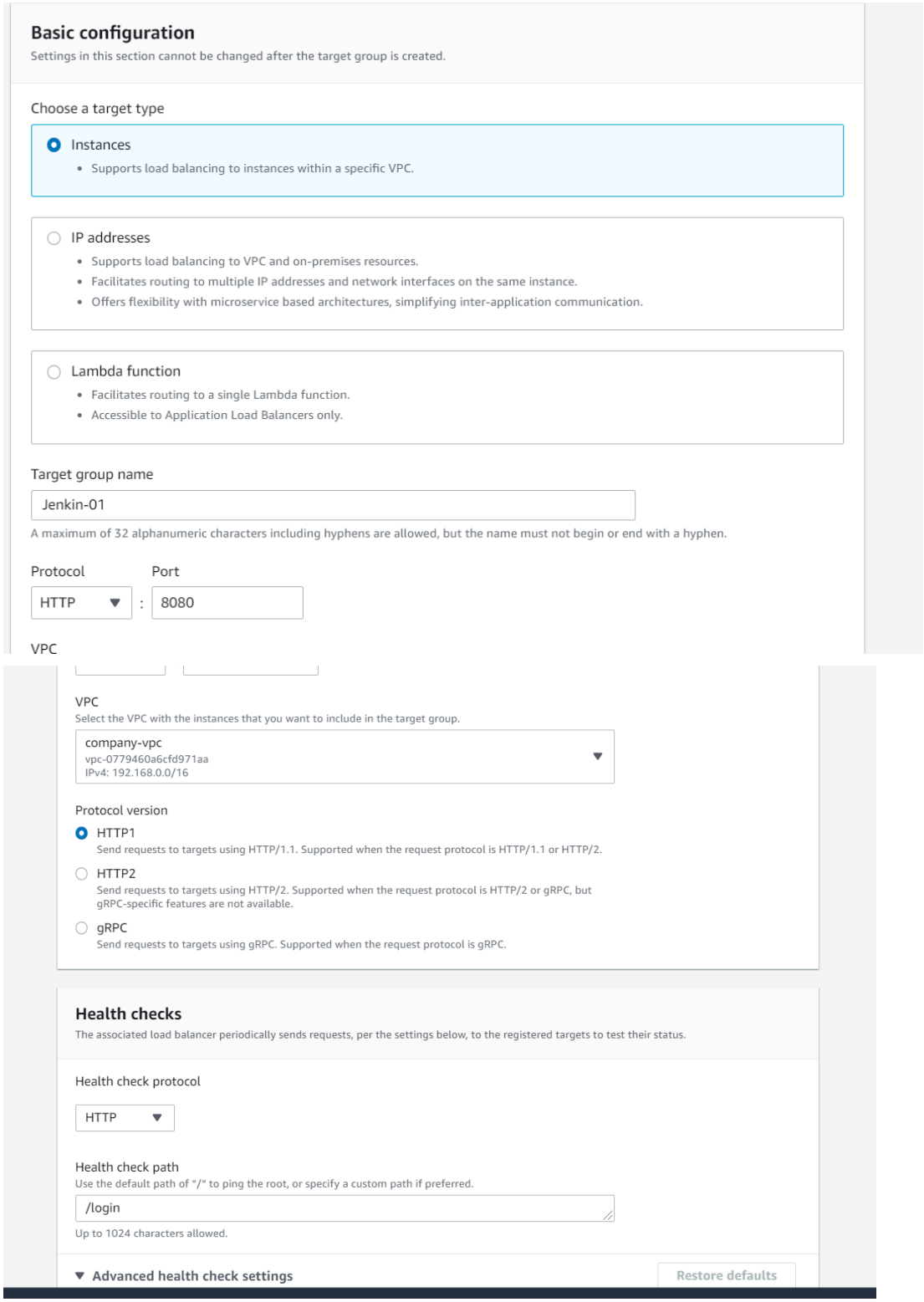
Click on Target Group under Load Balancing:

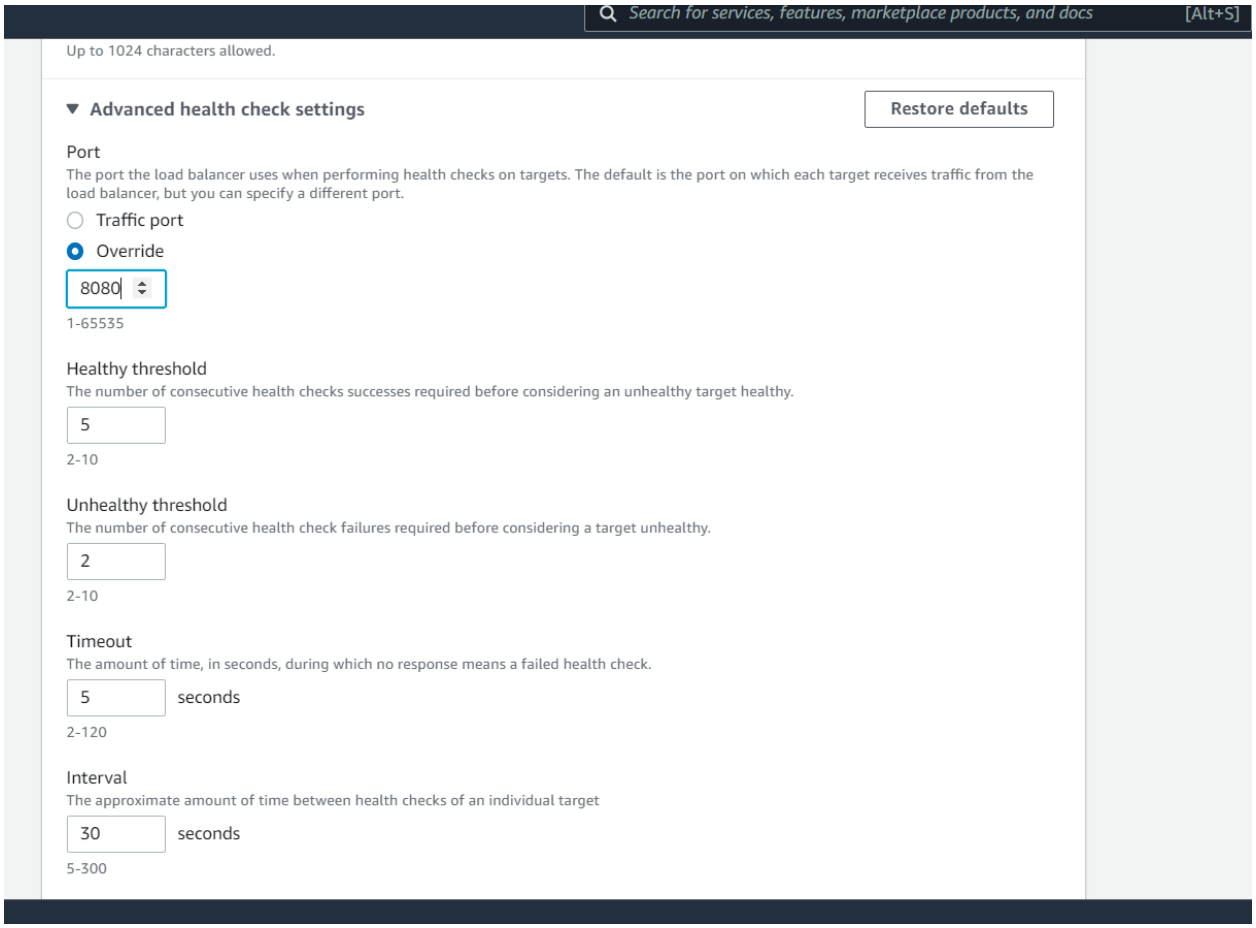


Create Target Group:

- Select instance
- Create a Target name
- Select HTTP 8080
- Select your VPC
- Select HTTP1
- Select HTTP

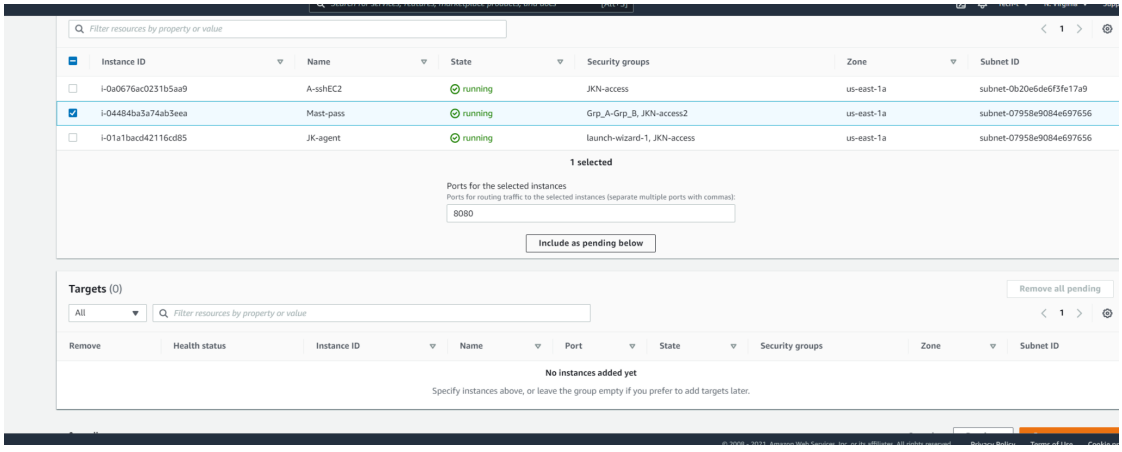
- Enter /login
- Select advance health check settings: select override and enter 8080





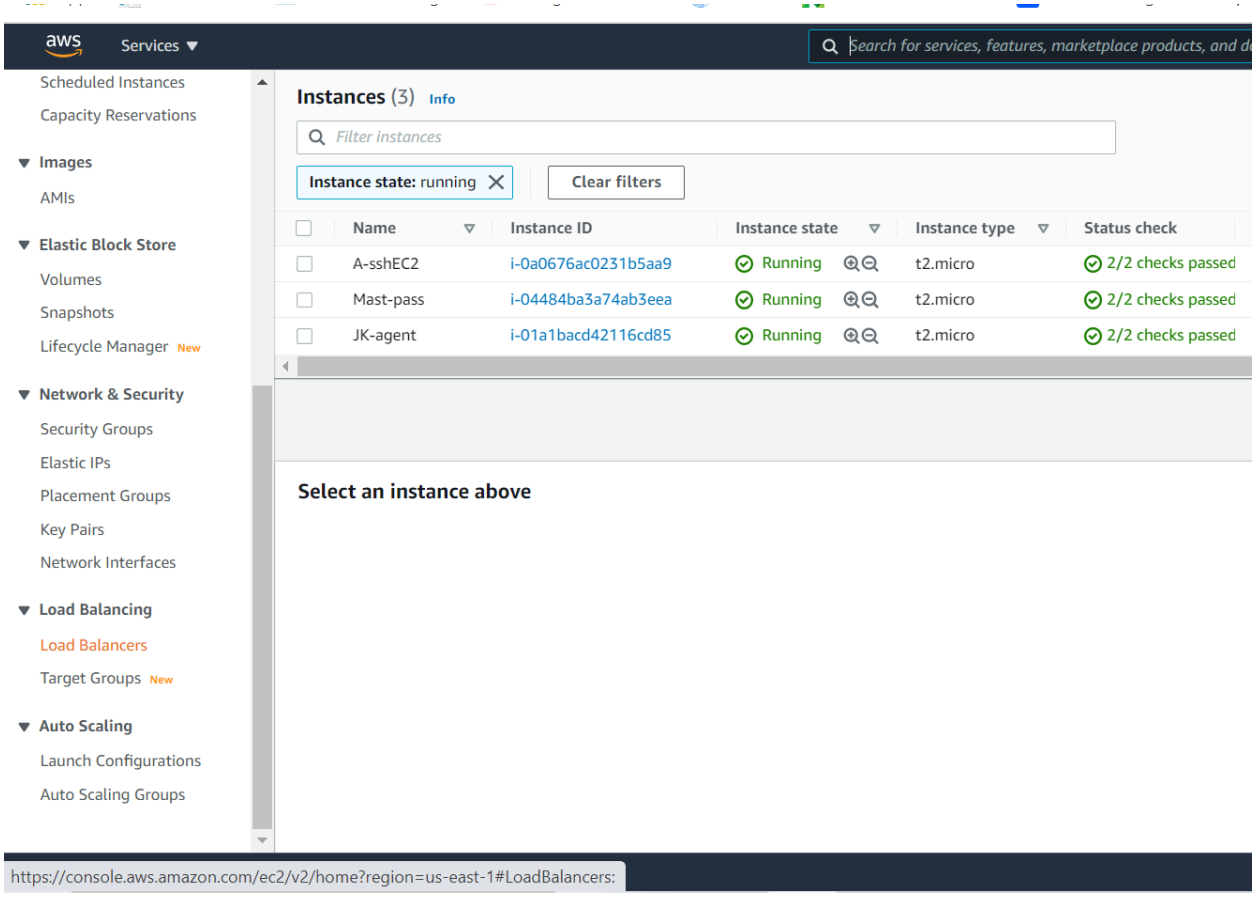
Select the next page:

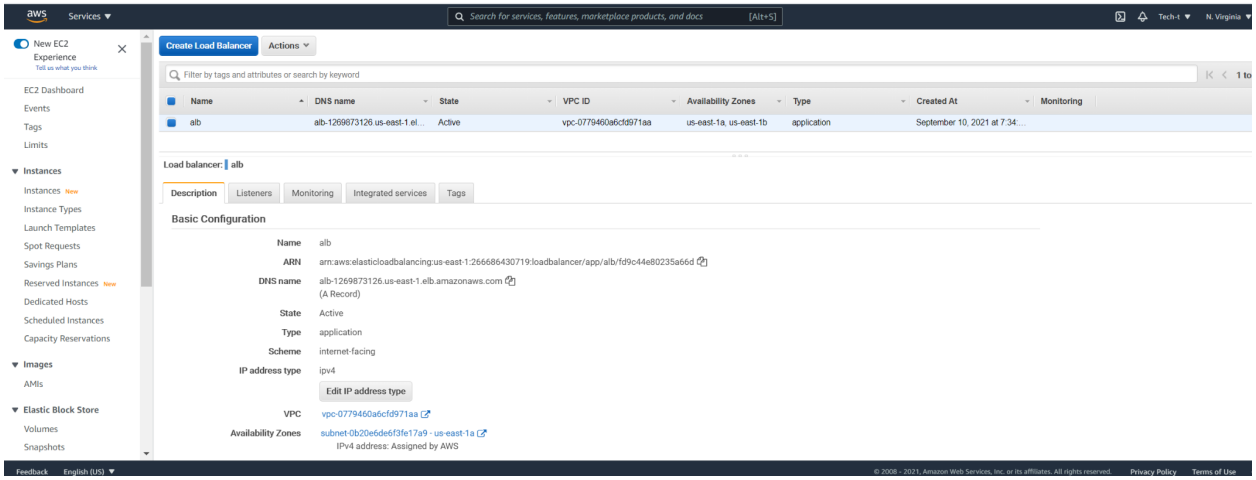
- Select your instance
- Click include as pending below
- finally select create target group

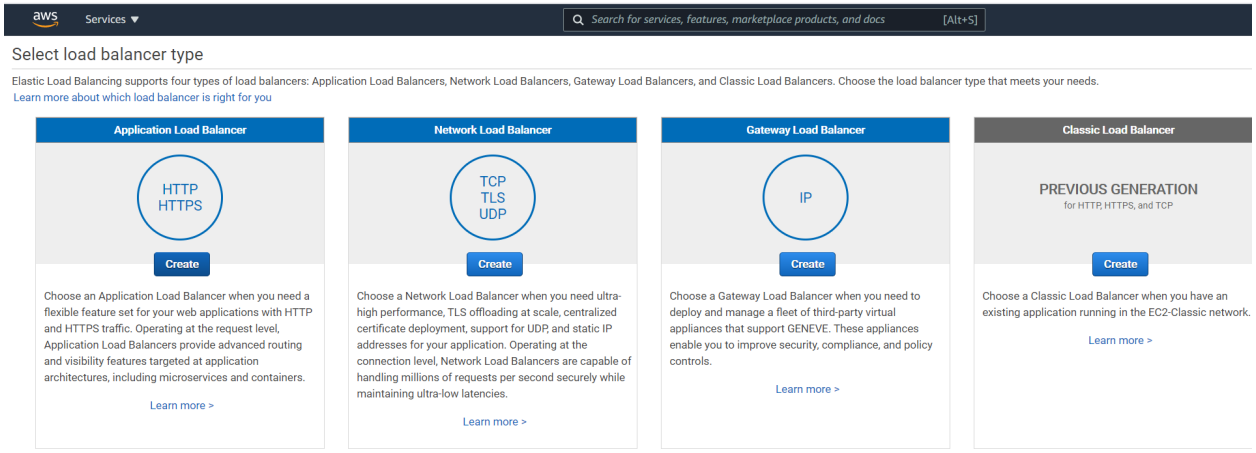


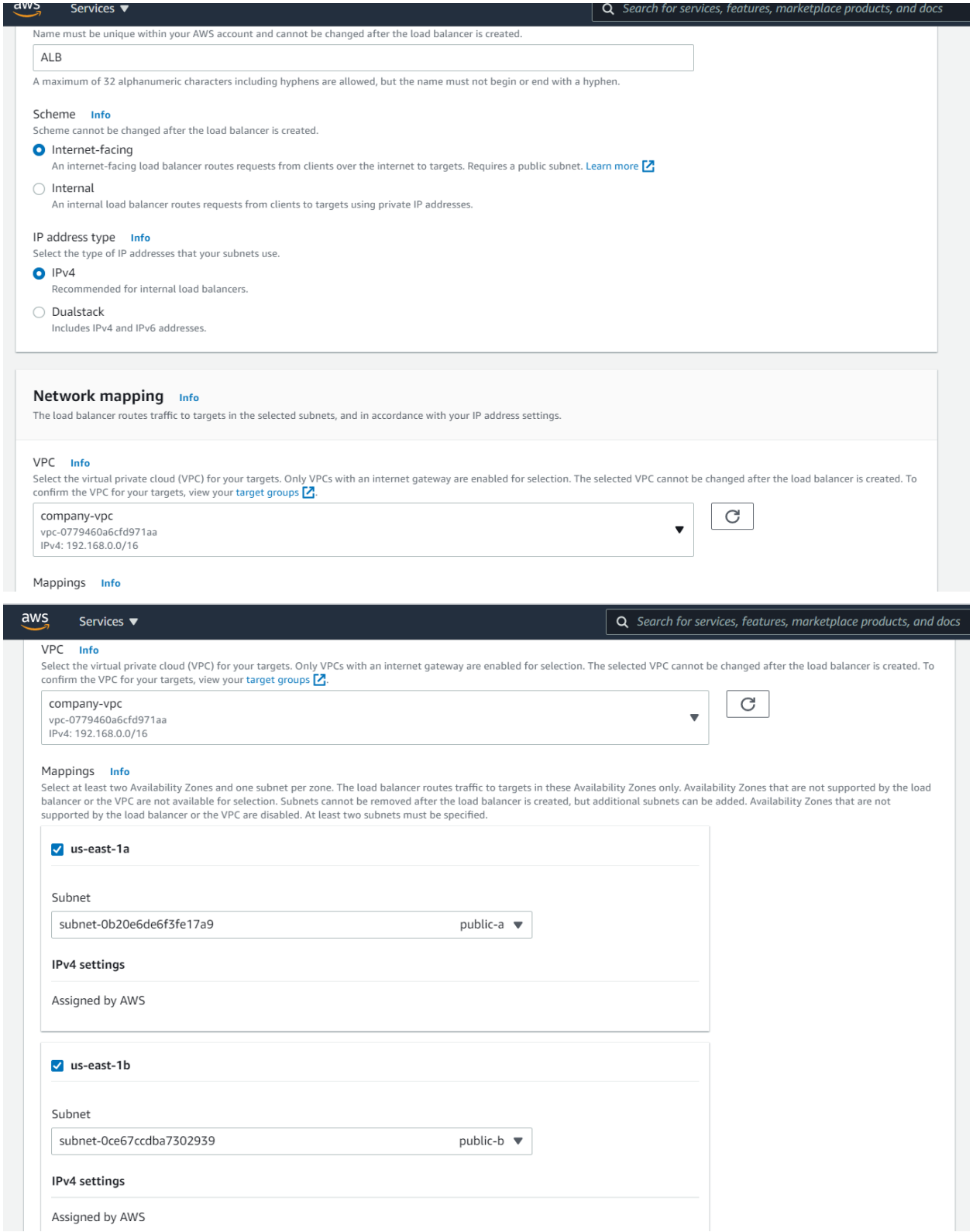
<u>Now create your ALB:</u>

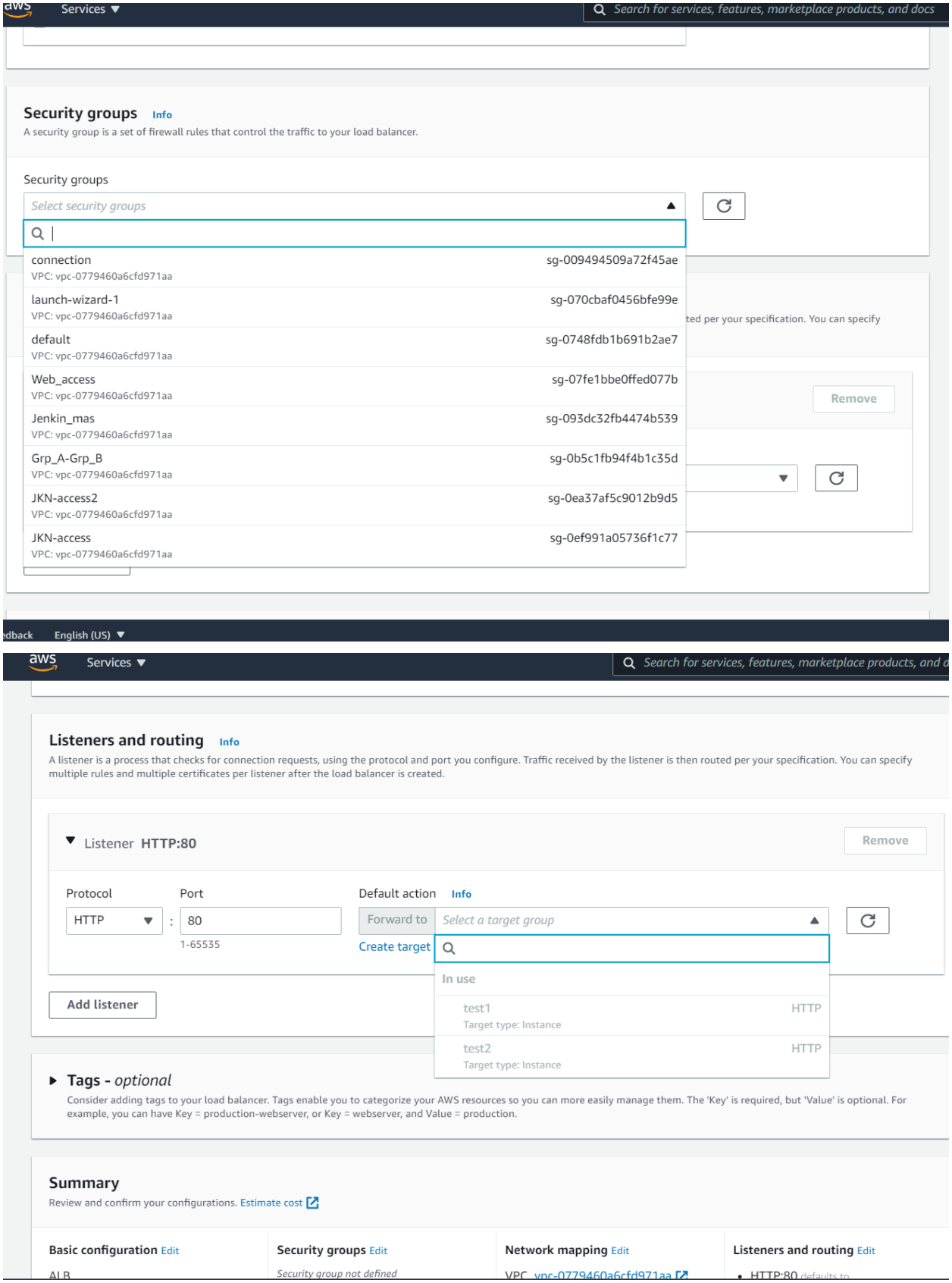
- Select create load balancer
- Select create application load balancer
- Name your load balancer
- Select select internet facing
- Select IPv4
- Select your VPC
- Select two AZ's and two public subnet
- Select select the security group for the ALB
- Select HTTP and your target group
- Finally select create load balancer




Services
Search for services, features, marketplace products, and docs [Alt+S]
Tech-t
N. Virginia
New EC2 Experience
Tell us what you think
EC2 Dashboard
Events
Tags
Limits
Instances
Instances New
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances New
Dedicated Hosts
Scheduled Instances
Capacity Reservations
Images
AMIs
Elastic Block Store
Volumes
Snapshots
Create Load Balancer
Actions
Filter by tags and attributes or search by keyword
Name
DNS name
State
VPC ID
Availability Zones
Type
Created At
Monitoring
alb
alb-1269873126.us-east-1.el...
Active
vpc-0779460a6cfd971aa
us-east-1a, us-east-1b
application
September 10, 2021 at 7:34:...
Load balancer: alb
Description
Listeners
Monitoring
Integrated services
Tags
Basic Configuration
Name alb
ARN arn:aws:elasticloadbalancing:us-east-1:266686430719:loadbalancer/app/alb/fd9c44e80235a66d
DNS name alb-1269873126.us-east-1.elb.amazonaws.com (A Record)
State Active
Type application
Scheme internet-facing
IP address type ipv4
Edit IP address type
VPC vpc-0779460a6cfd971aa
Availability Zones subnet-0b20e6de6f3fe17a9 · us-east-1a
IPv4 address: Assigned by AWS
Feedback
English (US)
© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use


Services
Search for services, features, marketplace products, and docs [Alt+S]
Select load balancer type
Elastic Load Balancing supports four types of load balancers: Application Load Balancers, Network Load Balancers, Gateway Load Balancers, and Classic Load Balancers. Choose the load balancer type that meets your needs.
Learn more about which load balancer is right for you
Application Load Balancer
HTTP
HTTPS
Create
Choose an Application Load Balancer when you need a flexible feature set for your web applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.
Learn more >
Network Load Balancer
TCP
TLS
UDP
Create
Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your application. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.
Learn more >
Gateway Load Balancer
IP
Create
Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls.
Learn more >
Classic Load Balancer
PREVIOUS GENERATION
for HTTP, HTTPS, and TCP
Create
Choose a Classic Load Balancer when you have an existing application running in the EC2-Classic network.
Learn more >


Services
Search for services, features, marketplace products, and docs
Name must be unique within your AWS account and cannot be changed after the load balancer is created.
ALB
A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.
Scheme Info
Scheme cannot be changed after the load balancer is created.
Internet-facing
An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. Learn more
Internal
An internal load balancer routes requests from clients to targets using private IP addresses.
IP address type Info
Select the type of IP addresses that your subnets use.
IPv4
Recommended for internal load balancers.
Dualstack
Includes IPv4 and IPv6 addresses.
Network mapping Info
The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.
VPC Info
Select the virtual private cloud (VPC) for your targets. Only VPCs with an internet gateway are enabled for selection. The selected VPC cannot be changed after the load balancer is created. To confirm the VPC for your targets, view your target groups.
company-vpc
vpc-0779460a6cfd971aa
IPv4: 192.168.0.0/16
Mappings Info
aws
Services
Search for services, features, marketplace products, and docs
VPC Info
Select the virtual private cloud (VPC) for your targets. Only VPCs with an internet gateway are enabled for selection. The selected VPC cannot be changed after the load balancer is created. To confirm the VPC for your targets, view your target groups.
company-vpc
vpc-0779460a6cfd971aa
IPv4: 192.168.0.0/16
Mappings Info
Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection. Subnets cannot be removed after the load balancer is created, but additional subnets can be added. Availability Zones that are not supported by the load balancer or the VPC are disabled. At least two subnets must be specified.
us-east-1a
Subnet
subnet-0b20e6de6f3fe17a9
public-a
IPv4 settings
Assigned by AWS
us-east-1b
Subnet
subnet-0ce67ccdba7302939
public-b
IPv4 settings
Assigned by AWS


Services
Search for services, features, marketplace products, and docs
Security groups Info
A security group is a set of firewall rules that control the traffic to your load balancer.
Security groups
Select security groups
connection
VPC: vpc-0779460a6cfd971aa
sg-009494509a72f45ae
launch-wizard-1
VPC: vpc-0779460a6cfd971aa
sg-070cbaf0456bfe99e
default
VPC: vpc-0779460a6cfd971aa
sg-0748fdb1b691b2ae7
Web_access
VPC: vpc-0779460a6cfd971aa
sg-07fe1bbe0ffed077b
Jenkin_mas
VPC: vpc-0779460a6cfd971aa
sg-093dc32fb4474b539
Grp_A-Grp_B
VPC: vpc-0779460a6cfd971aa
sg-0b5c1fb94f4b1c35d
JKN-access2
VPC: vpc-0779460a6cfd971aa
sg-0ea37af5c9012b9d5
JKN-access
VPC: vpc-0779460a6cfd971aa
sg-0ef991a05736f1c77
Remove
English (US)
Services
Search for services, features, marketplace products, and
Listeners and routing Info
A listener is a process that checks for connection requests, using the protocol and port you configure. Traffic received by the listener is then routed per your specification. You can specify multiple rules and multiple certificates per listener after the load balancer is created.
Listener HTTP:80
Remove
Protocol
HTTP
Port
80
1-65535
Default action Info
Forward to
Select a target group
Create target
In use
test1
Target type: Instance
HTTP
test2
Target type: Instance
HTTP
Add listener
Tags - optional
Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them. The 'Key' is required, but 'Value' is optional. For example, you can have Key = production-webserver, or Key = webserver, and Value = production.
Summary
Review and confirm your configurations. Estimate cost
Basic configuration Edit
Security groups Edit
Security group not defined
Network mapping Edit
Listeners and routing Edit

Summary

Review and confirm your configurations. Estimate cost

Basic configuration Edit

ALB

- Internet-facing
- IPv4

Security groups Edit

Security group not defined

Network mapping Edit

VPC vpc-0779460a6cfd971aa
company-vpc

- us-east-1a
  subnet-0b20e6de6f3fe17a9
  public-a
- us-east-1b
  subnet-0ce67ccdba7302939
  public-b

Listeners and routing Edit

- HTTP:80 defaults to
  Target group not defined

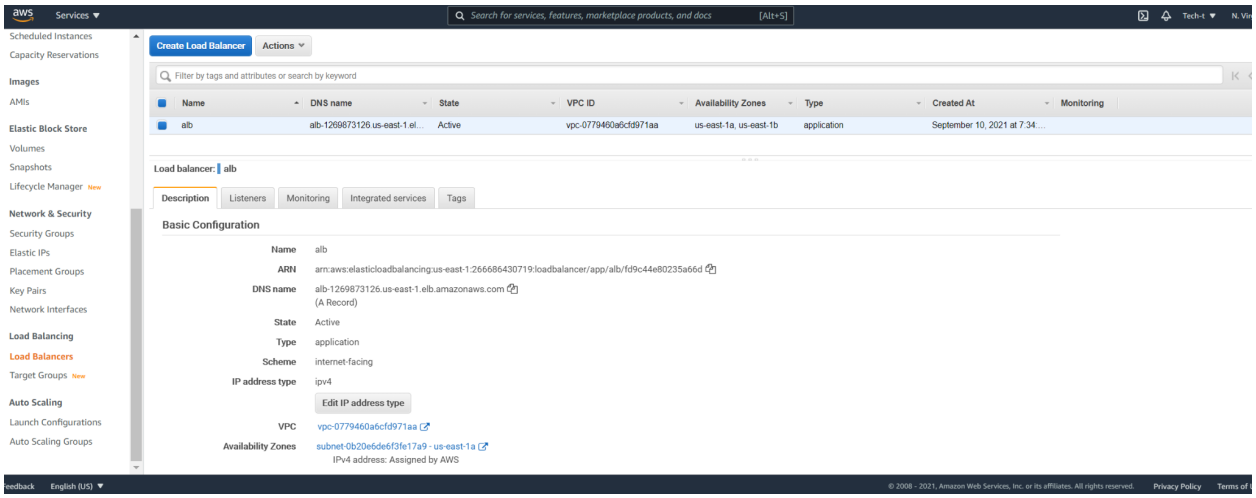
Tags Edit

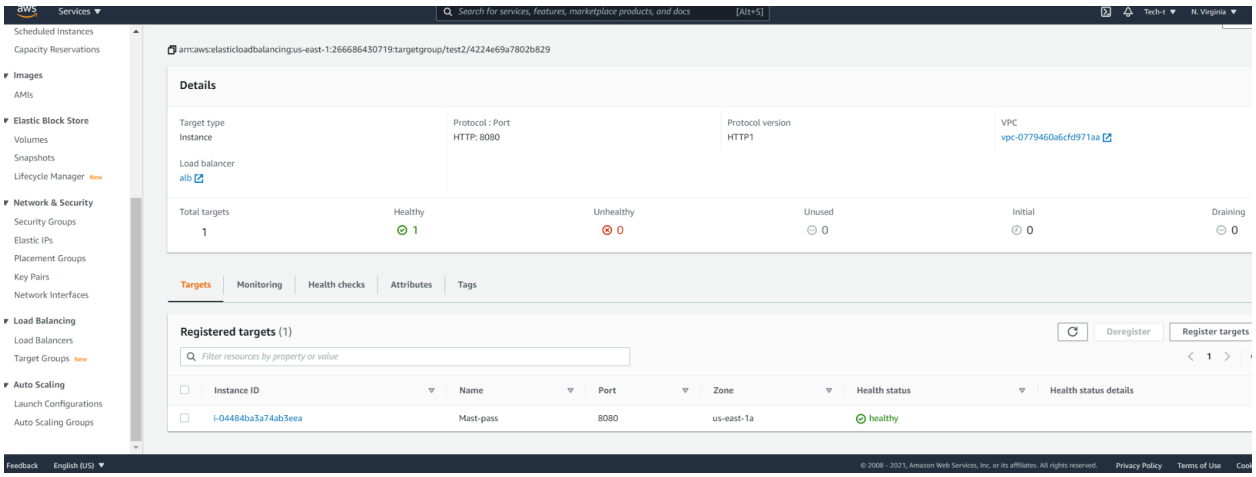
None

Attributes

Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.

Cancel | Create load balancer

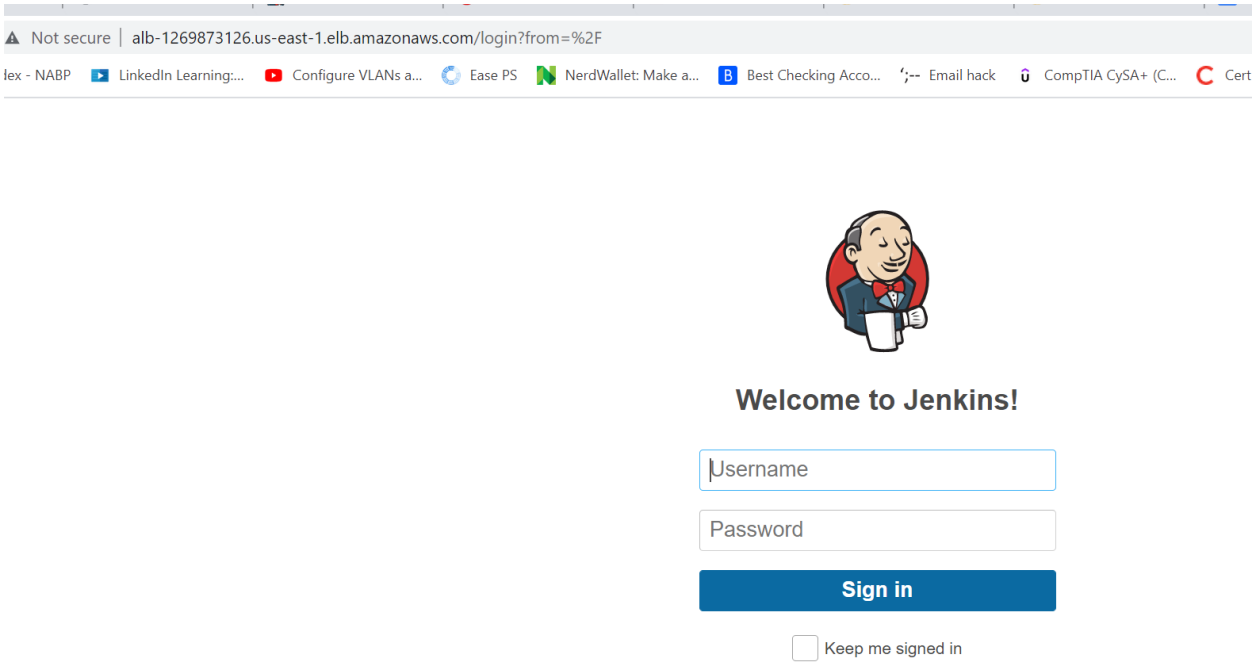




The ALB takes a few minutes to set up. Once the ALB is set, you will see the status is active and the target group is healthy in the Target Group section .

Step 3: Configure Jenkins UI using the ALB

Enter the domain name of the ALB to access Jenkins setup page:

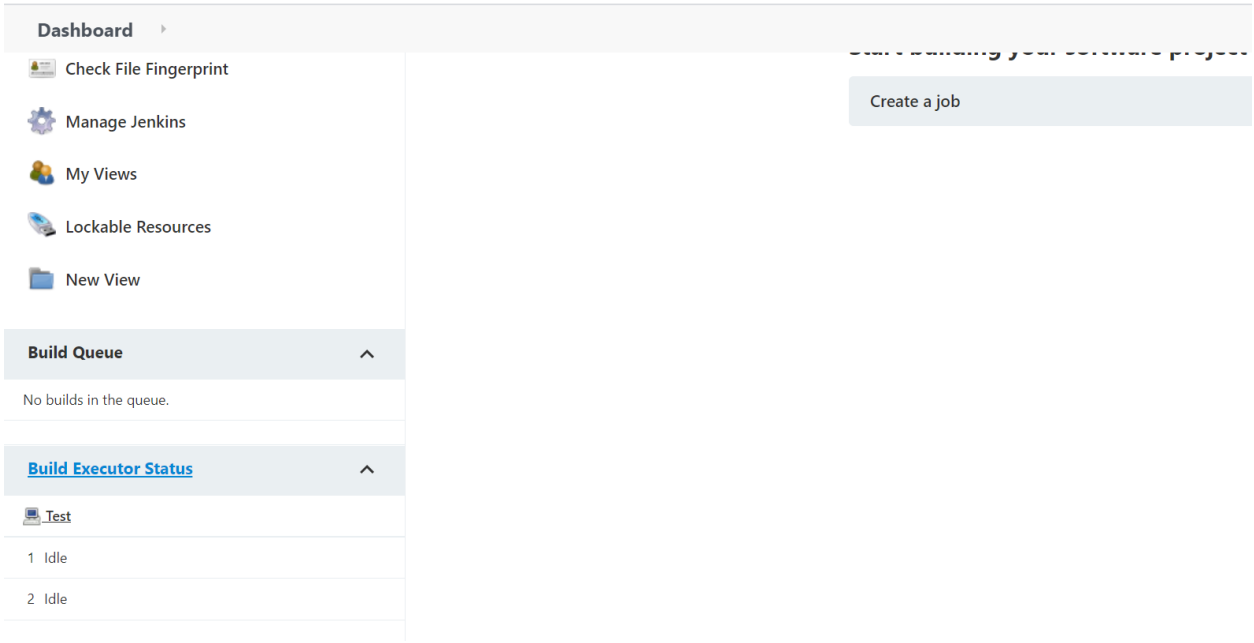


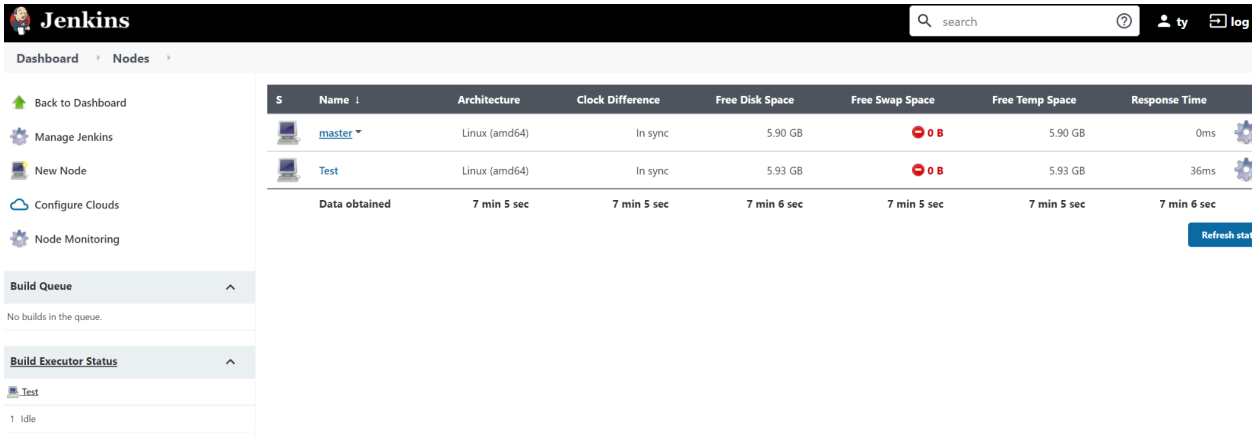
Step 4: **Create** another amazon Linux EC2 inside the same private subnet of the Jenkins master (this EC2 will be the Agent).

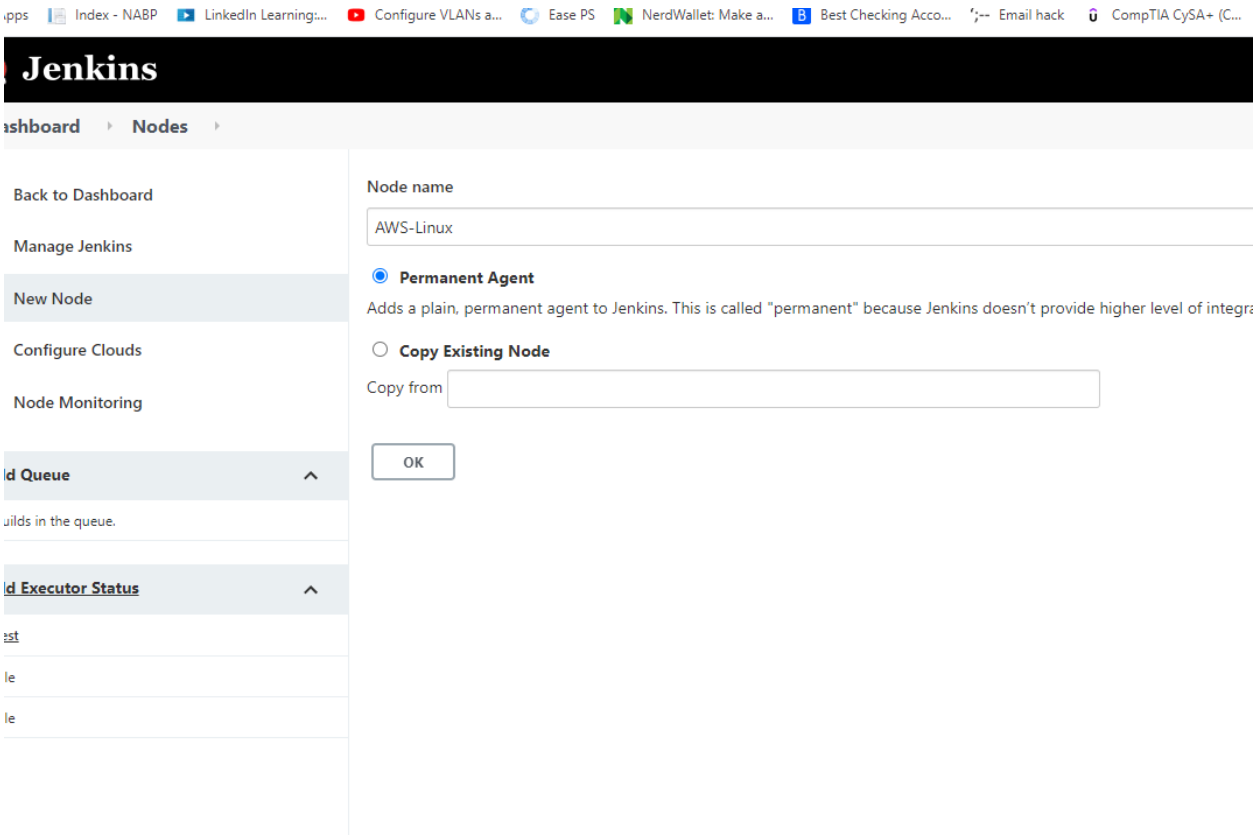
Before you configure Jenkins master to ssh into the agent, use your Jump Server to ssh into Jenkins. Next, ssh into the newly created EC2. Once you are able to ssh into the Amazon Linux instance, move on to the next step.

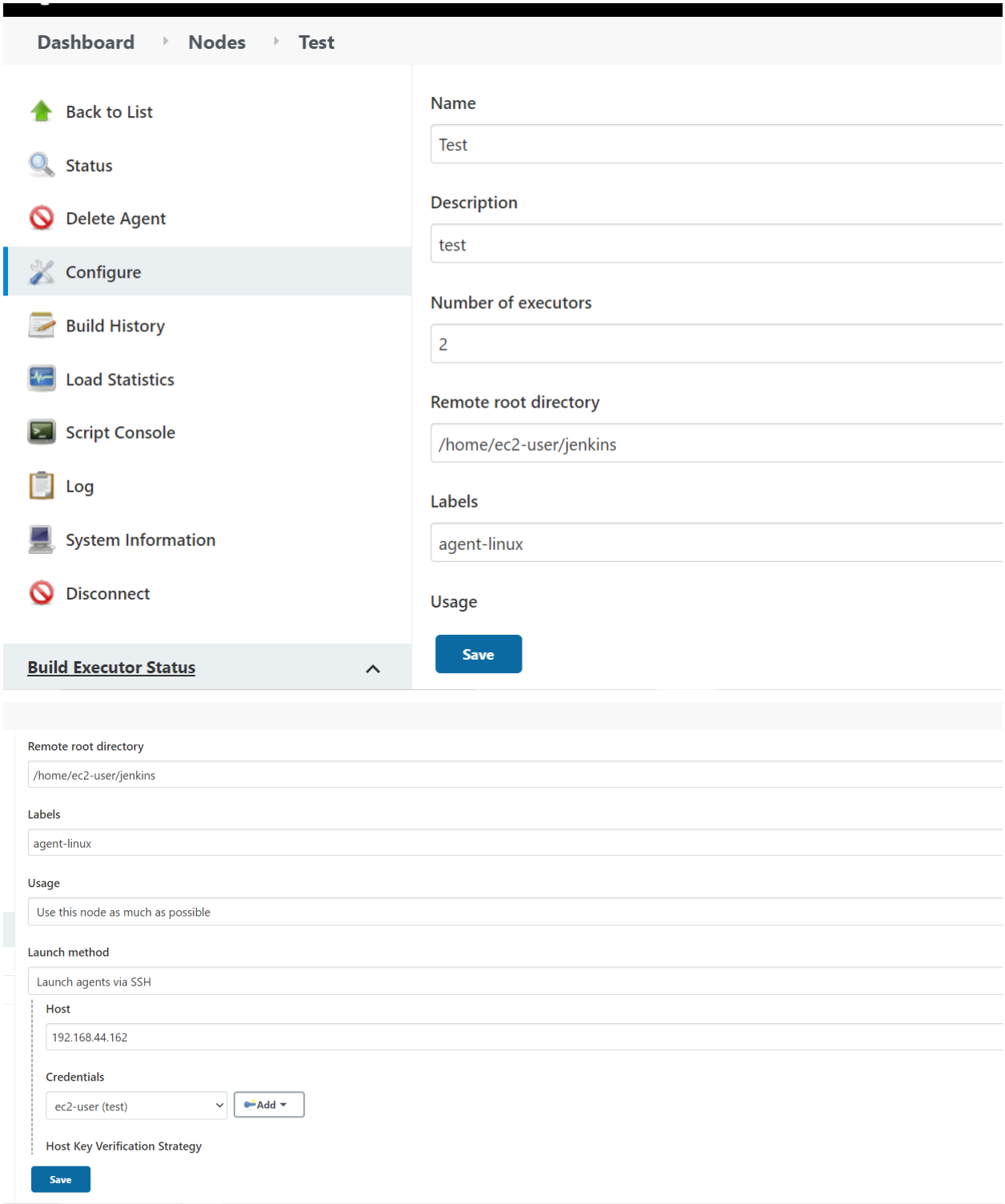
Step 5: Configure the Jenkins master to SSH into the Agent

- Select build executor status
- Select new node
- Create a name for the node and select permanent agent
- Create a name and description
- Enter 2 for executors
- enter {/home/ec2-user/jenkins} for remote root directory
- Create a label
- Select use this node as much as possible
- Select launch agent via ssh
- Enter the private IP address of the Agent
- Add SSH credentials (username: ec2-user | key: the private key you used to ssh into the agent)
- Select non verifying verification strategy
- save and then look at the logs to see if your setup was successful



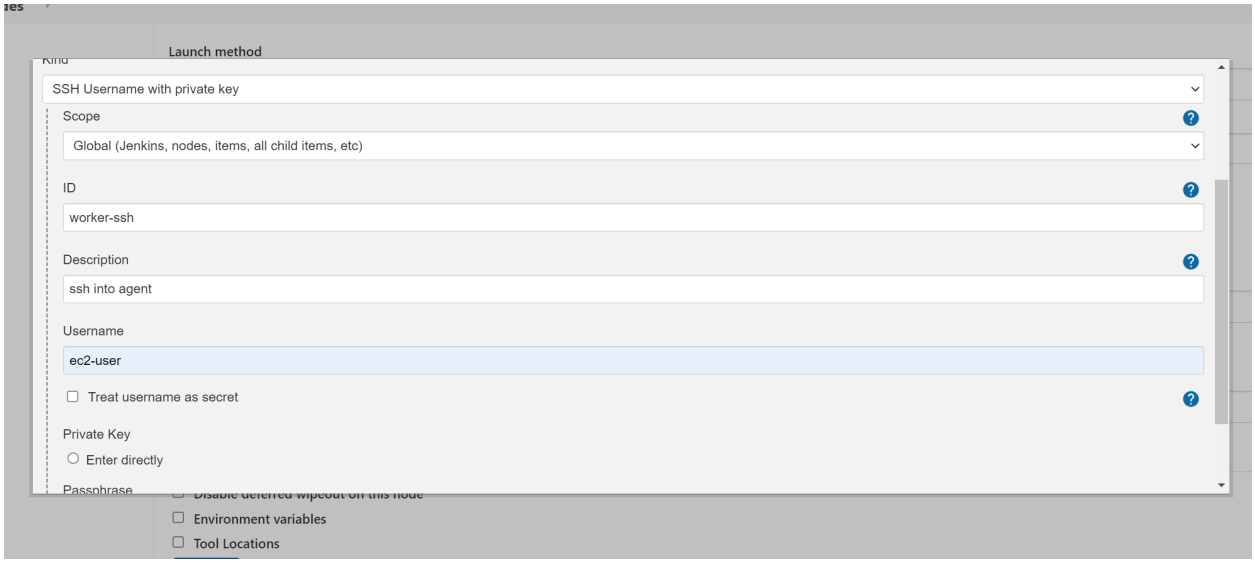

Jenkins
search
ty
log
Dashboard
Nodes
Back to Dashboard
Manage Jenkins
New Node
Configure Clouds
Node Monitoring
Build Queue
No builds in the queue.
Build Executor Status
Test
1 Idle
S
Name ↓
Architecture
Clock Difference
Free Disk Space
Free Swap Space
Free Temp Space
Response Time
master
Linux (amd64)
In sync
5.90 GB
0 B
5.90 GB
0ms
Test
Linux (amd64)
In sync
5.93 GB
0 B
5.93 GB
36ms
Data obtained
7 min 5 sec
7 min 5 sec
7 min 6 sec
7 min 5 sec
7 min 5 sec
7 min 6 sec
Refresh stat


Index - NABP
LinkedIn Learning:...
Configure VLANs a...
Ease PS
NerdWallet: Make a...
Best Checking Acco...
Email hack
CompTIA CySA+ (C...
Jenkins
Nodes
Back to Dashboard
Manage Jenkins
New Node
Configure Clouds
Node Monitoring
Node name
AWS-Linux
Permanent Agent
Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integra
Copy Existing Node
Copy from
OK



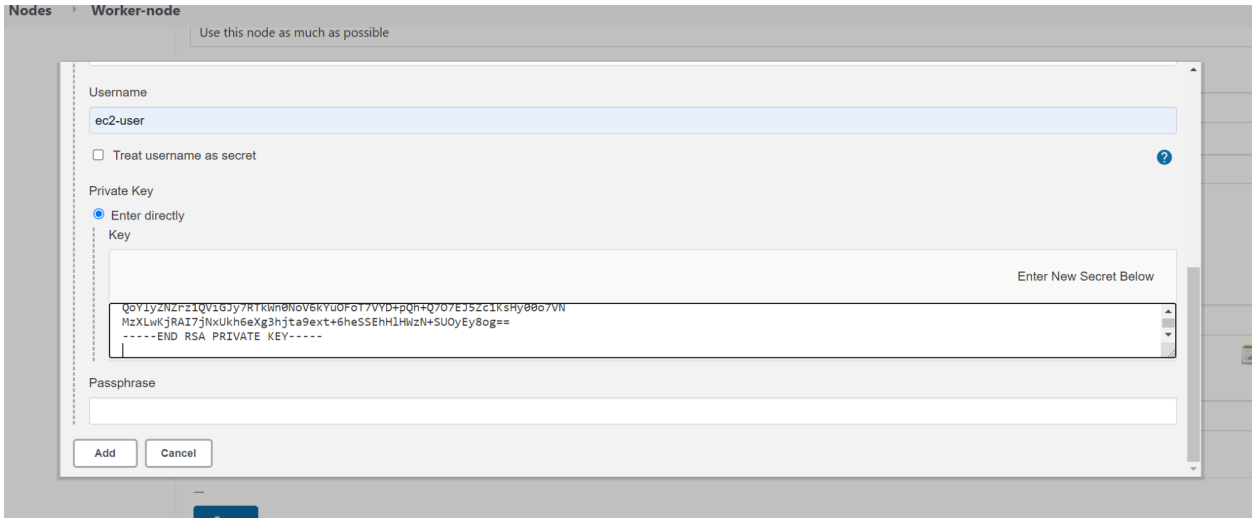
Adding SSH credentials:

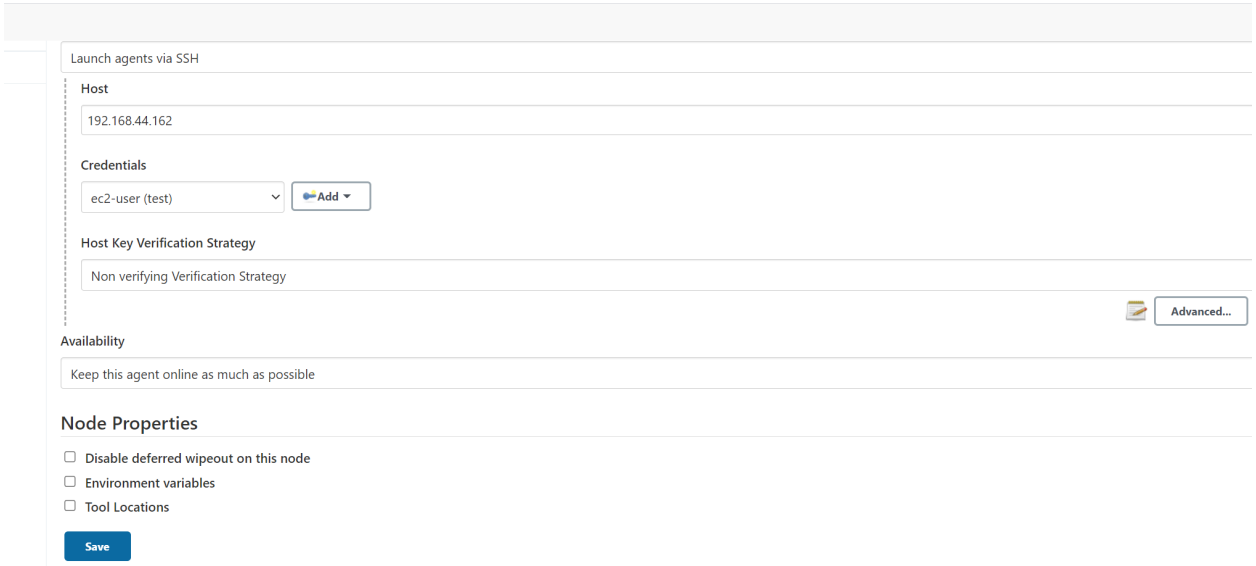
Choose SSH Username with private key

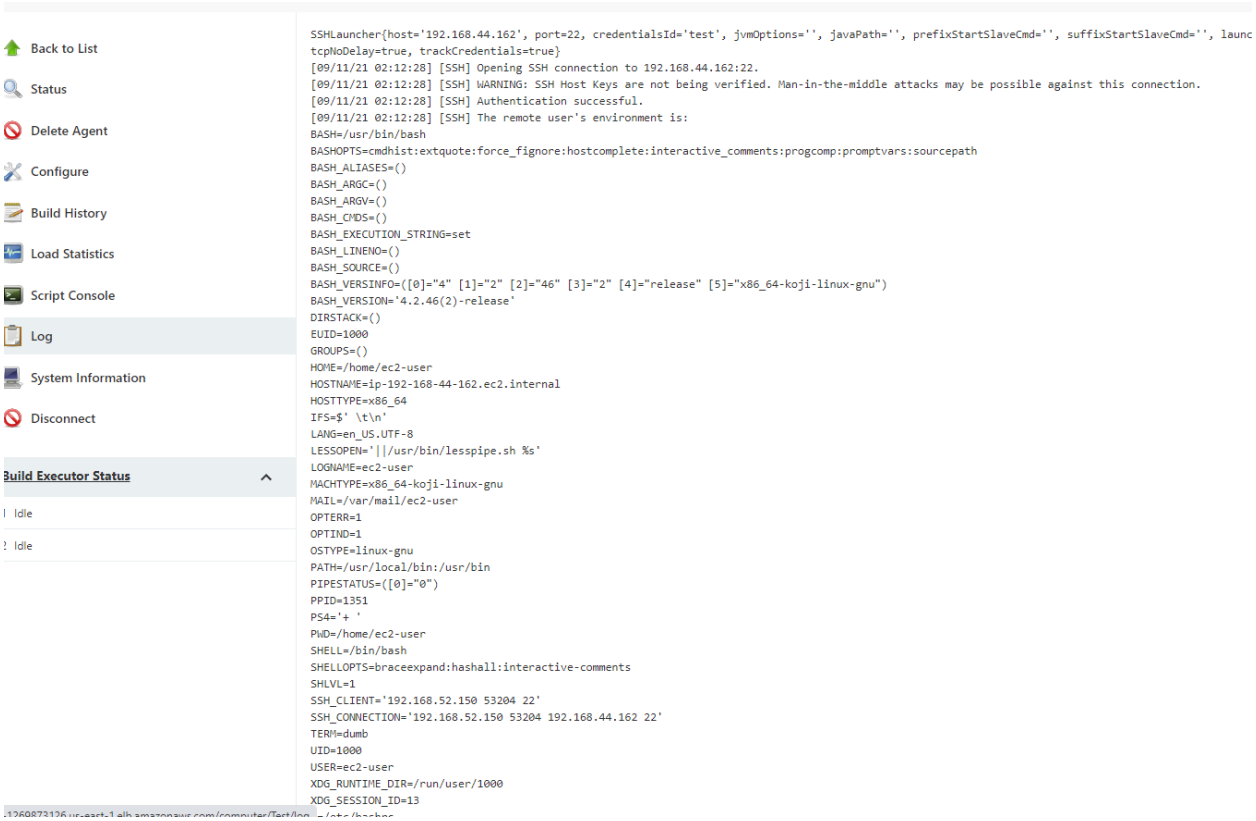


Enter username and select [ENTER DIRECTLY]

- No passphrase for the key




Launch agents via SSH
Host
192.168.44.162
Credentials
ec2-user (test)
Add
Host Key Verification Strategy
Non verifying Verification Strategy
Advanced...
Availability
Keep this agent online as much as possible
Node Properties
Disable deferred wipeout on this node
Environment variables
Tool Locations
Save


Back to List
Status
Delete Agent
Configure
Build History
Load Statistics
Script Console
Log
System Information
Disconnect
Build Executor Status
Idle
Idle
SSHLauncher{host='192.168.44.162', port=22, credentialsId='test', jvmOptions='', javaPath='', prefixStartSlaveCmd='', suffixStartSlaveCmd='', launc
tcpNoDelay=true, trackCredentials=true}
[09/11/21 02:12:28] [SSH] Opening SSH connection to 192.168.44.162:22.
[09/11/21 02:12:28] [SSH] WARNING: SSH Host Keys are not being verified. Man-in-the-middle attacks may be possible against this connection.
[09/11/21 02:12:28] [SSH] Authentication successful.
[09/11/21 02:12:28] [SSH] The remote user's environment is:
BASH=/usr/bin/bash
BASHOPTS=cmdhist:extquote:force_fignore:hostcomplete:interactive_comments:progcomp:promptvars:sourcepath
BASH_ALIASES=()
BASH_ARGC=()
BASH_ARGV=()
BASH_CMDS=()
BASH_EXECUTION_STRING=set
BASH_LINENO=()
BASH_SOURCE=()
BASH_VERSINFO=([0]="4" [1]="2" [2]="46" [3]="2" [4]="release" [5]="x86_64-koji-linux-gnu")
BASH_VERSION='4.2.46(2)-release'
DIRSTACK=()
EUID=1000
GROUPS=()
HOME=/home/ec2-user
HOSTNAME=ip-192-168-44-162.ec2.internal
HOSTTYPE=x86_64
IFS=$' \t\n'
LANG=en_US.UTF-8
LESSOPEN='||/usr/bin/lesspipe.sh %s'
LOGNAME=ec2-user
MACHTYPE=x86_64-koji-linux-gnu
MAIL=/var/mail/ec2-user
OPTERR=1
OPTIND=1
OSTYPE=linux-gnu
PATH=/usr/local/bin:/usr/bin
PIPESTATUS=([0]="0")
PPID=1351
PS4='+ '
PWD=/home/ec2-user
SHELL=/bin/bash
SHELLOPTS=braceexpand:hashall:interactive-comments
SHLVL=1
SSH_CLIENT='192.168.52.150 53204 22'
SSH_CONNECTION='192.168.52.150 53204 192.168.44.162 22'
TERM=dumb
UID=1000
USER=ec2-user
XDG_RUNTIME_DIR=/run/user/1000
XDG_SESSION_ID=13