

Deploying NGINX server in cloudshell using kubernetes

Using AWS CloudShell:

Access AWS CloudShell

Log into the AWS Management Console.

Click on the CloudShell icon at the top right of the console.

```
sudo yum -y update
```

Package	Architecture	Version	Repository	Size
amazon-linux-repo-cdn	x86_64	2023.5.20241010.0_amzn2023	amazonlinux	16 k
contalnerng	x86_64	1.7.22.1-amzn2023.0.2	amazonlinux	36 k
iproute	x86_64	6.18.0-319_amzn2023.0.1	amazonlinux	880 k
iproute-tc	x86_64	6.18.0-319_amzn2023.0.1	amazonlinux	434 k
libxxbcommon	x86_64	1.6.0-2_amzn2023.0.1	amazonlinux	145 k
system-release	noarch	2023.6.20241010.0_amzn2023	amazonlinux	28 k
xkeyboard-config	noarch	2.41-1_amzn2023.0.1	amazonlinux	922 k
Installing dependencies:				
kernel-libbpf	x86_64	6.1.112-122.189.amzn2023	amazonlinux	155 k
Transaction Summary				
Install 1 Package				
Upgrade 7 Packages				

```
curl -o kubectl
```

```
https://amazon-eks.s3.us-west-2.amazonaws.com/1.21.14/2022-11-01/bin/linux/amd64/kubectl
```

```
chmod +x ./kubectl
```

```
sudo mv ./kubectl /usr/local/bin
```

```
[cloudshell-user@ip-10-130-69-10 ~]$ curl -o kubectl https://amazon-eks.s3.us-west-2.amazonaws.com/1.21.14/2022-11-01/bin/linux/amd64/kubectl
% Total    % Received % Xferd  Average Speed   Time   Time     Current
          Dload  Upload   Total Spent    Left  Speed
100  325    0  325    0     0  1481      0 --:--:-- --:--:-- 1484
[cloudshell-user@ip-10-130-69-10 ~]$ chmod +x ./kubectl
[cloudshell-user@ip-10-130-69-10 ~]$ sudo mv ./kubectl /usr/local/bin
[cloudshell-user@ip-10-130-69-10 ~]$
```

Feedback

```
curl --silent --location
```

```
"https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_$(uname
```

```
-s)_amd64.tar.gz" | tar xz -C /tmp
```

```
sudo mv /tmp/eksctl /usr/local/bin
```

```
[cloudshell-user@ip-10-130-69-10 ~]$ curl --silent --location "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_$(uname -s)_amd64.tar.gz" | tar xz -C /tmp
[cloudshell-user@ip-10-130-69-10 ~]$ sudo mv /tmp/eksctl /usr/local/bin
[cloudshell-user@ip-10-130-69-10 ~]$ eksctl create cluster --name my-cluster --version 1.29 --region us-west-2 --nodegroup-name linux-nodes --node-type t2.micro --nodes 2 --nodes-min 1 --nodes-max 4 --managed
2024-10-23 09:46:23 [i] eksctl version 0.194.0
2024-10-23 09:46:23 [i] using region us-west-2
2024-10-23 09:46:24 [i] skipping us-west-2d from selection because it doesn't support the following instance type(s): t2.micro
2024-10-23 09:46:24 [i] skipping us-west-2t, newer & faster than us-west-2d
```

Create an EKS cluster

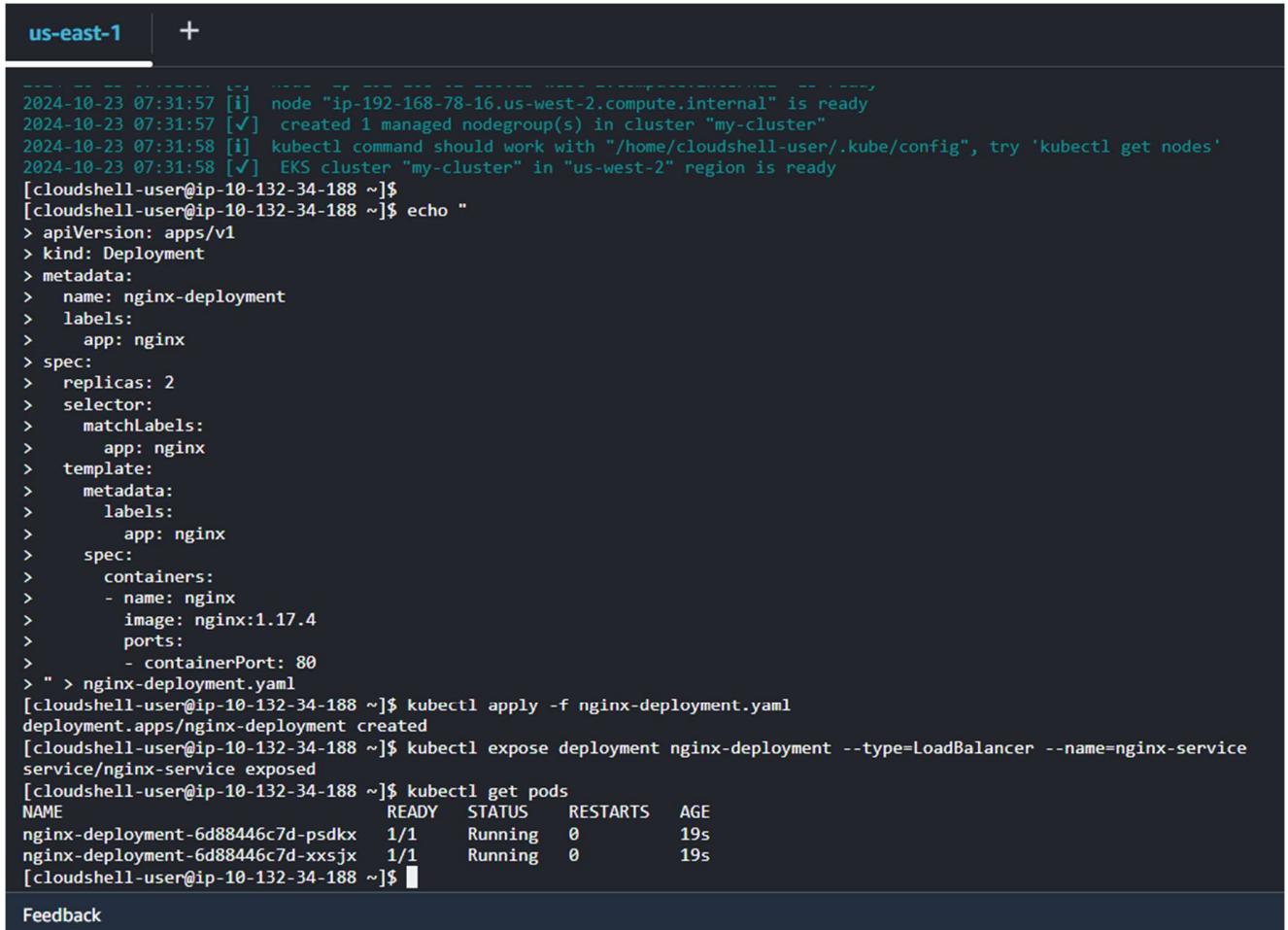
```
eksctl create cluster --name my-cluster --version 1.29 --region us-west-2 --nodegroup-name
linux-nodes --node-type t2.micro --nodes 2 --nodes-min 1 --nodes-max 4 --managed
```

```
kubectl get svc
```

Create a YAML file named nginx-pod.yaml with the following content:

```
kubectl apply -f nginx-pod.yaml
```

```
kubectl get pods
```



The screenshot shows a CloudShell terminal window titled "us-east-1". The terminal output is as follows:

```
2024-10-23 07:31:57 [i] node "ip-192-168-78-16.us-west-2.compute.internal" is ready
2024-10-23 07:31:57 [✓] created 1 managed nodegroup(s) in cluster "my-cluster"
2024-10-23 07:31:58 [i] kubectl command should work with "/home/cloudshell-user/.kube/config", try 'kubectl get nodes'
2024-10-23 07:31:58 [✓] EKS cluster "my-cluster" in "us-west-2" region is ready
[cloudshell-user@ip-10-132-34-188 ~]$ 
[cloudshell-user@ip-10-132-34-188 ~]$ echo "
> apiVersion: apps/v1
> kind: Deployment
> metadata:
>   name: nginx-deployment
>   labels:
>     app: nginx
> spec:
>   replicas: 2
>   selector:
>     matchLabels:
>       app: nginx
>   template:
>     metadata:
>       labels:
>         app: nginx
>     spec:
>       containers:
>         - name: nginx
>           image: nginx:1.17.4
>         ports:
>           - containerPort: 80
> " > nginx-deployment.yaml
[cloudshell-user@ip-10-132-34-188 ~]$ kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx-deployment created
[cloudshell-user@ip-10-132-34-188 ~]$ kubectl expose deployment nginx-deployment --type=LoadBalancer --name=nginx-service
service/nginx-service exposed
[cloudshell-user@ip-10-132-34-188 ~]$ kubectl get pods
NAME             READY   STATUS    RESTARTS   AGE
nginx-deployment-6d88446c7d-psdkx  1/1     Running   0          19s
nginx-deployment-6d88446c7d-xxsjx  1/1     Running   0          19s
[cloudshell-user@ip-10-132-34-188 ~]$ 
```

Feedback

You can see the pods thus , ngnix server is successfully deployed on kubernetes
Now we have to monitor using Nagios:

Step 4: Set Up Nagios for Monitoring

1. Install Nagios Core:

- Install Nagios Core on a separate machine or server
- Download the latest version of Nagios Core from the official website

Follow the installation instructions provided in the Nagios documentation

2. Configure Nagios to Monitor the Nginx Deployment:

- Add the necessary Nagios plugins to monitor HTTP services
- Configure Nagios to check the health of the Nginx pods by adding a command and service check in the Nagios configuration files.

Steps for Installing Nagios

1. Create EC2 Instance:

Launch an Amazon Linux EC2 instance named nagios-host.

Instances (1/1) Info		Last updated	Connect	Instance state	Actions	Launch instances	
		less than a minute ago		All states			
<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input checked="" type="checkbox"/>	nagios-host	i-070c8a6a7fe4e9adc	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b ec2-35

2. Configure Security Group:

Open inbound rules for HTTP, HTTPS, SSH, and ICMP

Inbound rules		Outbound rules	Tags
Inbound rules (5)			
<input type="checkbox"/>	Name	Security group rule...	IP version
<input type="checkbox"/>	-	sgr-0f04080bb5a20ec67	IPv6
<input type="checkbox"/>	-	sgr-0c18651ca52b0fa07	IPv4
<input type="checkbox"/>	-	sgr-0fbea99b148b82e7d	IPv4
<input type="checkbox"/>	-	sgr-0eb2ab2ee65b63cbf	IPv4
<input type="checkbox"/>	-	sgr-0b7d558557382e...	IPv4

3. SSH into EC2 Instance:

- Connect to the instance via SSH or EC2 Instance Connect.

EC2 Instance Connect Session Manager SSH client EC2 serial console

⚠ Port 22 (SSH) is open to all IPv4 addresses

Port 22 (SSH) is currently open to all IPv4 addresses, indicated by **0.0.0.0/0** in the inbound rule in [your security group](#). For increased security, consider restricting access to only the EC2 Instance Connect service IP addresses for your Region: 18.206.107.24/29. [Learn more](#).

Instance ID
 i-070c8a6a7fe4e9adc (nagios-host)

Connection Type

Connect using EC2 Instance Connect
Connect using the EC2 Instance Connect browser-based client, with a public IPv4 or IPv6 address.

Connect using EC2 Instance Connect Endpoint
Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

Public IPv4 address
 35.171.182.168

IPv6 address

—

Username
Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ec2-user.

ec2-user

4. Update Package Indices and Install Required Packages

Commands -

```
sudo yum update
```

```
sudo yum install httpd php
```

```
sudo yum install gcc glibc glibc-common
```

```
sudo yum install gd gd-devel
```

```

./m/
[ec2-user@ip-172-31-38-122 ~]$ sudo yum update
Last metadata expiration check: 0:17:45 ago on Wed Oct 23 10:04:49 2024.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-38-122 ~]$ sudo yum install httpd php
Last metadata expiration check: 0:17:56 ago on Wed Oct 23 10:04:49 2024.
Dependencies resolved.

=====
  Package           Architecture      Version       Repository
=====
Installing:
  httpd            x86_64          2.4.62-1.amzn2023
  php8_3           x86_64          8.3.10-1.amzn2023.0.1
Installing dependencies:
  apr              x86_64          1.7.2-2.amzn2023.0.2
  apr-util         x86_64          1.6.3-1.amzn2023.0.1
  generic-logos-httd
  httpd-core       noarch          18.0.0-12.amzn2023
  httpd-filesystem
  httpd-tools      x86_64          2.4.62-1.amzn2023
  libbrotli        x86_64          2.4.62-1.amzn2023
  libsslodium      x86_64          1.0.9-4.amzn2023
  libxml2          x86_64          1.0.19-4.amzn2023
  libxslt          x86_64          1.1.34-5.amzn2023
  mailcap          noarch          2.1.49-3.amzn2023.0.3
  nginx-filesystem

```

```

Complete!
[ec2-user@ip-172-31-38-122 ~]$ sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:19:01 ago on Wed Oct 23 10:04:49 2024.
Package glibc-2.34-52.amzn2023.0.11.x86_64 is already installed.
Package glibc-common-2.34-52.amzn2023.0.11.x86_64 is already installed.
Dependencies resolved.

=====
  Package           Architecture      Version       Repository
=====
Installing:
  gcc              x86_64          11.4.1-2.amzn2023.0.2
Installing dependencies:
  annobin-docs     noarch          10.93-1.amzn2023.0.1
  annobin-plugin-gcc
  cpp              x86_64          10.93-1.amzn2023.0.1
  gc               x86_64          11.4.1-2.amzn2023.0.2
  glibc-devel      x86_64          8.0.4-5.amzn2023.0.2
  glibc-headers-x86
  guile2          x86_64          2.34-52.amzn2023.0.11
  kernel-headers   noarch          2.34-52.amzn2023.0.11
  libmpc          x86_64          2.2.7-2.amzn2023.0.3
  libtinfo         x86_64          6.1.112-122.189.amzn2023
  libxml2          x86_64          1.2.1-2.amzn2023.0.2
  libxcrypt-devel  x86_64          2.4.7-1.amzn2023.0.3
  make             x86_64          4.4.33-7.amzn2023

```

Transaction Summary

```

Complete!
[ec2-user@ip-172-31-38-122 ~]$ sudo yum install gd gd-devel
Last metadata expiration check: 0:19:57 ago on Wed Oct 23 10:04:49 2024.
Dependencies resolved.

=====
  Package           Architecture      Version       Repository
=====
Installing:
  gd               x86_64          2.3.3-5.amzn2023.0.3
  gd-devel         x86_64          2.3.3-5.amzn2023.0.3
Installing dependencies:
  brotli          x86_64          1.0.9-4.amzn2023.0.2
  brotli-devel    x86_64          1.0.9-4.amzn2023.0.2
  bzip2-devel     x86_64          1.0.8-6.amzn2023.0.2
  cairo            x86_64          1.17.6-2.amzn2023.0.1
  cmake-filesystem
  fontconfig       x86_64          3.22.2-1.amzn2023.0.4
  fontconfig-devel
  fonts-filesystem
  freetype         x86_64          2.13.94-2.amzn2023.0.2
  freetype-devel   x86_64          2.13.94-2.amzn2023.0.2
  glib2-devel     x86_64          2.74.7-689.amzn2023.0.2
  google-noto-fonts-common
  google-noto-sans-vf-fonts
  graphite2        x86_64          20201206-2.amzn2023.0.2
  graphite2-devel  x86_64          20201206-2.amzn2023.0.2
  harfbuzz         x86_64          1.3.14-7.amzn2023.0.2

```

5.Create a New Nagios User

Commands -

```
sudo adduser -m nagios
```

```
sudo passwd nagios
```

```
Complete!
[ec2-user@ip-172-31-38-122 ~]$ sudo adduser -m nagios
[ec2-user@ip-172-31-38-122 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
BAD PASSWORD: The password contains the user name in some form
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-38-122 ~]$
```

i-070c8a6a7fe4e9adc (nagios-host)

Public IPs: 35.171.182.168 Private IPs: 172.31.38.122

6.create a New User Group

Commands -

```
sudo groupadd nagcmd
```

Add Users to the Group

Commands -

```
sudo usermod -a -G nagcmd nagios
```

```
sudo usermod -a -G nagcmd apache
```

```
[ec2-user@ip-172-31-38-122 ~]$ sudo usermod -a -G nagcmd nagios
[ec2-user@ip-172-31-38-122 ~]$ sudo usermod -a -G nagcmd apache
```

Create a Directory for Nagios Downloads

Commands -

```
mkdir ~/downloads
```

```
cd ~/downloads
```

```
[ec2-user@ip-172-31-38-122 ~]$ mkdir ~/downloads
[ec2-user@ip-172-31-38-122 ~]$ cd ~/downloads
```

Download Nagios and Plugins Source Files

Commands -

```
wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
```

```
wget https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
```

```
[ec2-user@ip-172-31-38-122 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
--2024-10-23 10:29:12-- https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
Resolving assets.nagios.com (assets.nagios.com) ... 45.79.49.120, 2600:3c00::f03c:92ff:fef7:45ce
Connecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11333414 (11M) [application/x-gzip]
Saving to: 'nagios-4.4.6.tar.gz'

nagios-4.4.6.tar.gz          100%[=====] 13.5 MB/s

2024-10-23 10:29:13 (13.5 MB/s) - 'nagios-4.4.6.tar.gz' saved [11333414/11333414]
```

```
[ec2-user@ip-172-31-38-122 downloads]$ wget https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
--2024-10-23 10:30:20-- https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org) ... 45.56.123.251
Connecting to nagios-plugins.org (nagios-plugins.org)|45.56.123.251|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2782610 (2.7M) [application/x-gzip]
Saving to: 'nagios-plugins-2.3.3.tar.gz'

nagios-plugins-2.3.3.tar.gz      100%[=====] 7.65 MB/s

2024-10-23 10:30:21 (7.65 MB/s) - 'nagios-plugins-2.3.3.tar.gz' saved [2782610/2782610]
```

```
2024-10-23 10:30:21 (7.65 MB/s) - 'nagios-plugins-2.3.3.tar.gz' saved [2782610/2782610]
```

```
[ec2-user@ip-172-31-38-122 downloads]$ tar zxvf nagios-4.4.6.tar.gz
nagios-4.4.6/
nagios-4.4.6/.gitignore
nagios-4.4.6/.travis.yml
nagios-4.4.6/CONTRIBUTING.md
nagios-4.4.6/Changelog
nagios-4.4.6/INSTALLING
nagios-4.4.6/LEGAL
nagios-4.4.6/LICENSE
nagios-4.4.6/Makefile.in
nagios-4.4.6/README.md
nagios-4.4.6/THANKS
nagios-4.4.6/UPGRADING
nagios-4.4.6/aclocal.m4
nagios-4.4.6/autoconf-macros/
nagios-4.4.6/autoconf-macros/.gitignore
nagios-4.4.6/autoconf-macros/CHANGELOG.md
nagios-4.4.6/autoconf-macros/LICENSE
nagios-4.4.6/autoconf-macros/LICENSE.md
nagios-4.4.6/autoconf-macros/README.md
nagios-4.4.6/autoconf-macros/add_group_user
nagios-4.4.6/autoconf-macros/ax_nagios_get_distrib
nagios-4.4.6/autoconf-macros/ax_nagios_get_files
```

Extract the Nagios Source File

Commands -

```
tar zxvf nagios-4.4.6.tar.gz
```

```
cd nagios-4.4.6
```

Run the Configuration Script

Commands -

```
./configure --with-command-
```

```
group=nagcmd
```

```
nagios-4.4.6/xdata/xsddefault.h
[ec2-user@ip-172-31-38-122 downloads]$ cd nagios-4.4.6
[ec2-user@ip-172-31-38-122 nagios-4.4.6]$ ./configure --with-command-group=nagcmd
checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking whether make sets $(MAKE)... yes
checking whether ln -s works... yes
checking for strip... /usr/bin/strip
checking how to run the C preprocessor... gcc -E
checking for grep that handles long lines and -e... /usr/bin/grep
checking for egrep... /usr/bin/grep -E
checking for ANSI C header files... yes
checking whether time.h and sys/time.h may both be included... yes
checking for sys/wait.h that is POSIX.1 compatible... yes
checking for sys/types.h... yes
checking for sys/stat.h... yes
checking for stdlib.h... yes
```

Compile the Source Code

Commands –

make all

```
* Support Notes ****
you have questions about configuring or running Nagios,
please make sure that you:
- Look at the sample config files
- Read the documentation on the Nagios Library at:
  https://library.nagios.com

Before you post a question to one of the mailing lists,
so make sure to include pertinent information that could
help others help you. This might include:
- What version of Nagios you are using
- What version of the plugins you are using
- Relevant snippets from your config files
- Relevant error messages from the Nagios log file

For more information on obtaining support for Nagios, visit:
  https://support.nagios.com
*****
joy.

[ec2-user@ip-172-31-38-122 nagios-4.4.6]$
```

Install Binaries, Init Script, and Sample Config Files

Commands -

./sudo make install

sudo make install-init

sudo make install-config

sudo make install-commandmode

```
[ec2-user@ip-172-31-38-122 nagios-4.4.6]$ sudo make install
cd ./base && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/base'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagios /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagiostats /usr/local/nagios/bin
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.6/base'
cd ./cgi && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/cgi'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/cgi'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/sbin
for file in *.cgi; do \
    /usr/bin/install -c -s -m 775 -o nagios -g nagios $file /usr/local/nagios/sbin; \
done
make install-config
- This installs sample config files in /usr/local/nagios/etc

make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.6'
[ec2-user@ip-172-31-38-122 nagios-4.4.6]$ sudo make install-init
/usr/bin/install -c -m 755 -d /root -g root /lib/systemd/system
/usr/bin/install -c -m 755 -o root -g root startup/default-service /lib/systemd/system/nagios.service
[ec2-user@ip-172-31-38-122 nagios-4.4.6]$ sudo make install-config
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc/objects
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg /usr/local/nagios/etc/nagios.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/cgi.cfg /usr/local/nagios/etc/cgi.cfg
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /usr/local/nagios/etc/resource.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/templates.cfg /usr/local/nagios/etc/objects/templates.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/commands.cfg /usr/local/nagios/etc/objects/commands.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/contacts.cfg /usr/local/nagios/etc/objects/contacts.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/timeperiods.cfg /usr/local/nagios/etc/objects/timeperiods.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/localhost.cfg /usr/local/nagios/etc/objects/localhost.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/switch.cfg /usr/local/nagios/etc/objects/switch.cfg
```

```
c2-user@ip-172-31-38-122 nagios-4.4.6]$ sudo make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagcmd -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw
```

* External command directory configured ***

```
c2-user@ip-172-31-38-122 nagios-4.4.6]$
```

Edit the Config File

Commands -

```
sudo nano /usr/local/nagios/etc/objects/commands.cfg sudo
```

```
nano /usr/local/nagios/etc/objects/services.cfg
```

```
define command {
    command_name  notify-host-by-email
    command_line   /usr/bin/printf "%b" "***** Nagios *****\n\nNotification Type: $NOTIFICATIONTYPE$\nHost: $HOSTNAME$\nState: $HOSTSTATE$\nAddress: $HOSTADDRESS$\n\n"
}

define command {
    command_name  notify-service-by-email
    command_line   /usr/bin/printf "%b" "***** Nagios *****\n\nNotification Type: $NOTIFICATIONTYPE$\n\nService: $SERVICEDESC$\nHost: $HOSTALIAS$\nAddress: $HOSTADDRESS$\n\n"
}

define command {
    command_name  check_nginx
    command_line   $USER1$/check_http -I $HOSTADDRESS$ -p 80
}

#####
#
```

```
GNU nano 5.8
define service {
    use          generic-service
    host_name    localhost
    service_description Nginx
    check_command   check_nginx
    check_interval  0.5
    retry_interval  0.5
    max_check_attempts 3
}
```

Configure the Web Interface

Commands –

```
sudo make install-webconf
```

```
[ec2-user@ip-172-31-38-122 nagios-4.4.6]$ sudo make install-webconf
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/httpd/conf.d/nagios.conf
if [ 0 -eq 1 ]; then \
    ln -s /etc/httpd/conf.d/nagios.conf /etc/apache2/sites-enabled/nagios.conf; \
fi

*** Nagios/Apache conf file installed ***

[ec2-user@ip-172-31-38-122 nagios-4.4.6]$ █
```

Create a Nagios Admin Account

Commands –

```
sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

You will be prompted to enter and confirm the password for the nagiosadmin user

```
[ec2-user@ip-172-31-38-122 nagios-4.4.6]$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
[ec2-user@ip-172-31-38-122 nagios-4.4.6]$ █
```

i-070c8a6a7fe4e9adc (nagios-host)

Public IPs: 35.171.182.168 Private IPs: 172.31.38.122

Restart Apache

Commands -

```
sudo systemctl restart httpd
```

Extract the Plugins Source File

Commands –

```
cd ~/downloads
```

```
tar zxf nagios-plugins-2.3.3.tar.gz  
[ec2-user@ip-172-31-38-122 nagios-4.4.6]$ sudo systemctl restart httpd  
[ec2-user@ip-172-31-38-122 nagios-4.4.6]$ cd ~/downloads  
[ec2-user@ip-172-31-38-122 downloads]$ tar zxvf nagios-plugins-2.3.3.tar.gz  
nagios-plugins-2.3.3/  
nagios-plugins-2.3.3/perlmods/  
nagios-plugins-2.3.3/perlmods/Config-Tiny-2.14.tar.gz  
nagios-plugins-2.3.3/perlmods/parent-0.226.tar.gz  
nagios-plugins-2.3.3/perlmods/Test-Simple-0.98.tar.gz  
nagios-plugins-2.3.3/perlmods/Makefile.in  
nagios-plugins-2.3.3/perlmods/version-0.9903.tar.gz  
nagios-plugins-2.3.3/perlmods/Makefile.am  
nagios-plugins-2.3.3/perlmods/Module-Runtime-0.013.tar.gz  
nagios-plugins-2.3.3/perlmods/Module-Metadata-1.000014.tar.gz  
nagios-plugins-2.3.3/perlmods/Params-Validate-1.08.tar.gz  
nagios-plugins-2.3.3/perlmods/Class-Accessor-0.34.tar.gz  
nagios-plugins-2.3.3/perlmods/Try-Tiny-0.18.tar.gz  
nagios-plugins-2.3.3/perlmods/Module-Implementation-0.07.tar.gz  
nagios-plugins-2.3.3/perlmods/Makefile
```

If Error

```
sudo mkdir -p /usr/local/nagios/var/spool/checkresults  
sudo chown -R nagios:nagcmd /usr/local/nagios/var
```

Compile and Install Plugins

Commands -

```
./configure --with-nagios-user=nagios --with-nagios-group=nagios
```

```
make
```

```
sudo make install
```

```
[ec2-user@ip-172-31-38-122 nagios-plugins-2.3.3]$ ./configure --with-nagios-user=nagios --with-nagios-group=nagios  
checking for a BSD-compatible install... /usr/bin/install -c  
checking whether build environment is sane... yes  
checking for a thread-safe mkdir -p... /usr/bin/mkdir -p  
checking for gawk... gawk  
checking whether make sets $(MAKE)... yes  
checking whether to disable maintainer-specific portions of Makefiles... yes  
checking build system type... x86_64-unknown-linux-gnu  
checking host system type... x86_64-unknown-linux-gnu  
checking for gcc... gcc  
checking for C compiler default output file name... a.out  
checking whether the C compiler works... yes  
checking whether we are cross compiling... no  
checking for suffix of executables...  
checking for suffix of object files... o  
checking whether we are using the GNU C compiler... yes  
checking whether gcc accepts -g... yes  
checking for gcc option to accept ISO C89... none needed  
checking for style of include used by make... GNU  
checking dependency style of gcc... gcc3
```

```
[ec2-user@ip-172-31-38-122 nagios-plugins-2.3.3]$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
Nagios Core 4.4.6
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2020-04-28
License: GPL

Website: https://www.nagios.org
Reading configuration data...
Error: Cannot open main configuration file '/home/ec2-user/downloads/nagios-plugins-2.3.3/usr/local/nagios/etc/nagios.cfg' fo
[ec2-user@ip-172-31-38-122 nagios-plugins-2.3.3]$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.4.6
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2020-04-28
License: GPL

Website: https://www.nagios.org
Reading configuration data...
  Read main config file okay...
  Read object config files okay...

Running pre-flight check on configuration data...
Checking objects...
```

Start Nagios

Commands -

```
sudo chkconfig --add nagios
```

```
sudo chkconfig nagios on
```

```
sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

```
sudo systemctl start nagios
```

Check Nagios Status: sudo

systemctl status nagios

```
Total Warnings: 0
Total Errors: 0

Things look okay - No serious problems were detected during the pre-flight che
[ec2-user@ip-172-31-38-122 nagios-plugins-2.3.3]$ sudo service nagios start
Redirecting to /bin/systemctl start nagios.service
[ec2-user@ip-172-31-38-122 nagios-plugins-2.3.3]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.6
    Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset:
      Active: active (running) since Wed 2024-10-23 11:14:47 UTC; 3min 45s ago
        Docs: https://www.nagios.org/documentation
     Main PID: 70052 (nagios)
       Tasks: 6 (limit: 1112)
      Memory: 2.8M
         CPU: 62ms
      CGroup: /system.slice/nagios.service
              ├─70052 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
              ├─70053 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/v
              ├─70054 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/v
              ├─70055 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/v
              ├─70056 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/v
              └─70057 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 23 11:14:47 ip-172-31-38-122.ec2.internal nagios[70052]: qh: echo service
Oct 23 11:14:47 ip-172-31-38-122.ec2.internal nagios[70052]: qh: help for the
Oct 23 11:14:47 ip-172-31-38-122.ec2.internal nagios[70052]: wproc: Successful
```

Nagios Page will be visible:

Username : nagiosadmin

Password : your password

The screenshot shows the Nagios Core 4.4.6 dashboard. At the top right, it says "Nagios® Core™ Version 4.4.6 April 28, 2020 Check for updates". A message box at the top right says "A new version of Nagios Core is available! Visit nagios.org to download Nagios 4.5.6." On the left, there's a sidebar with links for General, Current Status, Problems, Reports, and System. The main area has sections for Get Started, Latest News, and Quick Links.

The screenshot shows the "Service Status Details For All Hosts" page. It displays a table of services with columns for Host, Service, Status, Last Check, Duration, Attempt, and Status Information. Services listed include Current Load, Current Users, HTTP, PING, Root Partition, Swap Usage, and Total Processes. The table includes filters for Limit Results (set to 100) and sorting by various columns.

Host	Service	Status	Last Check	Duration	Attempt	Status Information
localhost	Current Load	OK	10-23-2024 08:07:19	0d 0h 2m 18s+	1/4	OK - load average: 0.24, 0.10, 0.04
	Current Users	OK	10-23-2024 08:08:02	0d 0h 2m 18s+	1/4	USERS OK - 0 users currently logged in
	HTTP	WARNING	10-23-2024 08:08:45	0d 0h 0m 10s	1/4	HTTP WARNING: HTTP/1.1 401 Unauthorized - 695 bytes in 0.000 second response time
	PING	PENDING	N/A	0d 0h 2m 18s+	1/4	Service check scheduled for Wed Oct 23 08:09:28 UTC 2024
	Root Partition	PENDING	N/A	0d 0h 2m 18s+	1/4	Service check scheduled for Wed Oct 23 08:10:11 UTC 2024
	Swap Usage	PENDING	N/A	0d 0h 2m 18s+	1/4	Service check scheduled for Wed Oct 23 08:10:54 UTC 2024
	Total Processes	PENDING	N/A	0d 0h 2m 18s+	1/4	Service check scheduled for Wed Oct 23 08:11:37 UTC 2024