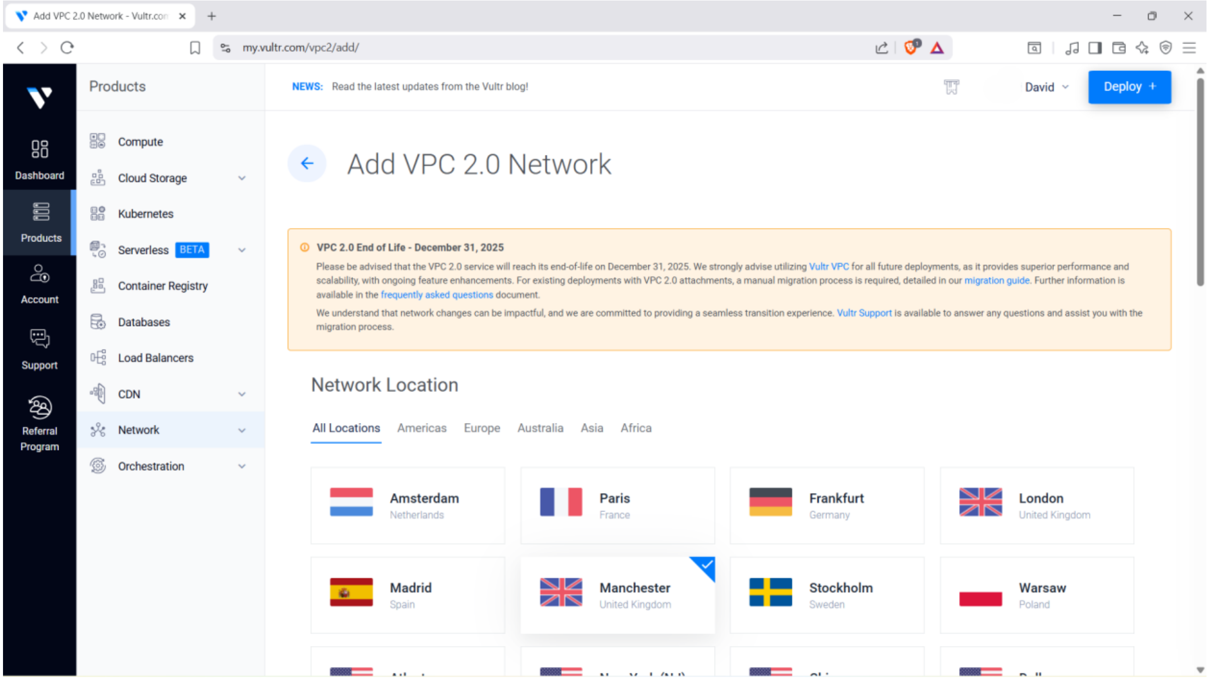
ELASTICSEARCH AND KIBANA SETUP.

Objective: how to setup elasticsearch on vultr cloud platform.

Tools Used

- **Vultr Cloud Platform** – for hosting the virtual private cloud (VPC) and virtual machines.
- **Ubuntu Server** – the operating system used for deployment.
- **Elasticsearch** – the core engine for indexing and searching log data.
- **Kibana** – the visualization and dashboard tool for analyzing Elasticsearch data.
- **PowerShell/SSH** – for remote access and configuration.
- **Firewall Rules & UFW** – for securing access to Elasticsearch and Kibana services.

After creating an account on Vultr, you will have an interface like this.



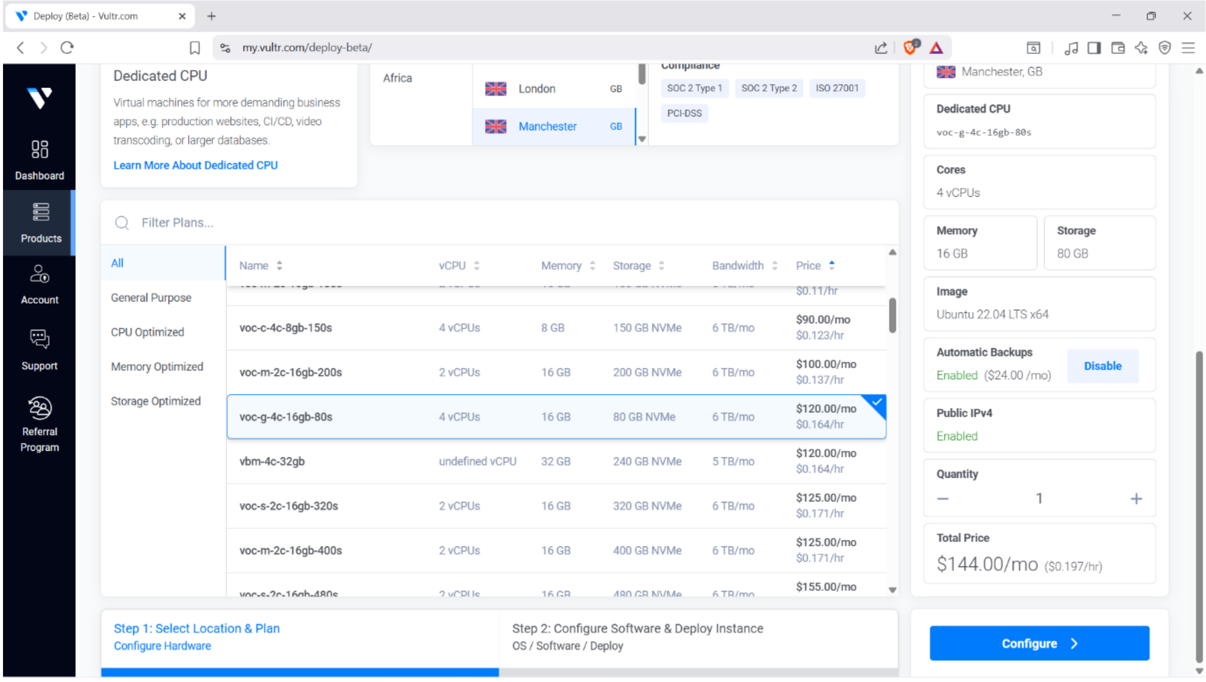
VPC 2.0 End of Life - December 31, 2025

Please be advised that the VPC 2.0 service will reach its end-of-life on December 31, 2025. We strongly advise utilizing Vultr VPC for all future deployments, as it provides superior performance and scalability, with ongoing feature enhancements. For existing deployments with VPC 2.0 attachments, a manual migration process is required, detailed in our migration guide. Further information is available in the frequently asked questions document.

We understand that network changes can be impactful, and we are committed to providing a seamless transition experience. Vultr Support is available to answer any questions and assist you with the migration process.

You need to create a Virtual Private Cloud (VPC) network. Select your network location.

Note: when you create a VPC, all of your Virtual machines created in your vpc must have the same location as that of the VPC.



Deploy (Beta) - Vultr.com

my.vultr.com/deploy-beta/

Dedicated CPU

Virtual machines for more demanding business apps, e.g. production websites, CI/CD, video transcoding, or larger databases.

Learn More About Dedicated CPU

Africa

London GB

Manchester GB

Compliance

SOC 2 Type 1 SOC 2 Type 2 ISO 27001 PCI-DSS

Filter Plans...

All
General Purpose
CPU Optimized
Memory Optimized
Storage Optimized

Name	vCPU	Memory	Storage	Bandwidth	Price
					$0.11/hr
voc-c-4c-8gb-150s	4 vCPUs	8 GB	150 GB NVMe	6 TB/mo	$90.00/mo $0.123/hr
voc-m-2c-16gb-200s	2 vCPUs	16 GB	200 GB NVMe	6 TB/mo	$100.00/mo $0.137/hr
voc-g-4c-16gb-80s	4 vCPUs	16 GB	80 GB NVMe	6 TB/mo	$120.00/mo $0.164/hr
vbm-4c-32gb	undefined vCPU	32 GB	240 GB NVMe	5 TB/mo	$120.00/mo $0.164/hr
voc-s-2c-16gb-320s	2 vCPUs	16 GB	320 GB NVMe	6 TB/mo	$125.00/mo $0.171/hr
voc-m-2c-16gb-400s	2 vCPUs	16 GB	400 GB NVMe	6 TB/mo	$125.00/mo $0.171/hr
voc-s-2c-16gb-480s	2 vCPUs	16 GB	480 GB NVMe	6 TB/mo	$155.00/mo

Manchester, GB

Dedicated CPU
voc-g-4c-16gb-80s

Cores
4 vCPUs

Memory
16 GB

Storage
80 GB

Image
Ubuntu 22.04 LTS x64

Automatic Backups
Enabled ($24.00 /mo) Disable

Public IPv4
Enabled

Quantity
– 1 +

Total Price
$144.00/mo ($0.197/hr)

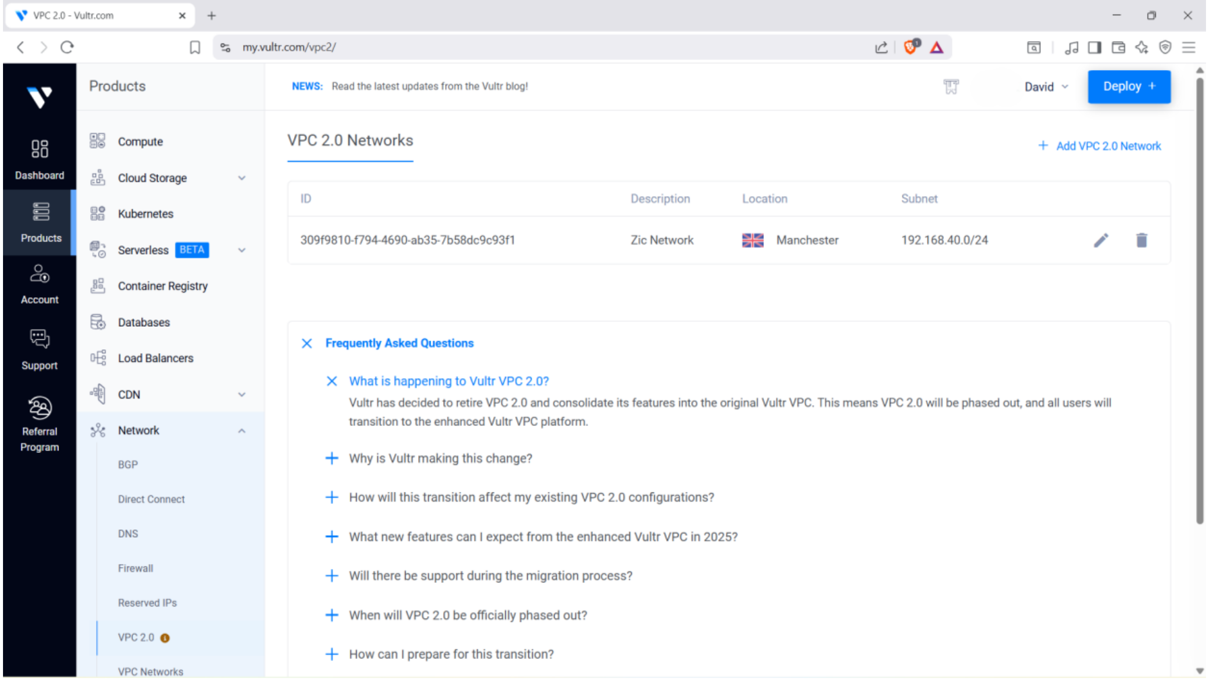
Step 1: Select Location & Plan
Configure Hardware

Step 2: Configure Software & Deploy Instance
OS / Software / Deploy

Configure >

Next step is to configure your IP and give your VPC a name.

You click “deploy” to deploy a new server. The features of the server include the type (dedicated CPU), location (same as that of VPC), the image (ubuntu) and the plan you wish to work with. Always ensure to disable backups and IPv6, as it is not needed for this plan. Ensure to include VPC Network as the server will be under the VPC you created.



VPC 2.0 - Vultr.com

my.vultr.com/vpc2/

Products

Compute
Cloud Storage
Kubernetes
Serverless BETA
Container Registry
Databases
Load Balancers
CDN
Network
BGP
Direct Connect
DNS
Firewall
Reserved IPs
VPC 2.0
VPC Networks

NEWS: Read the latest updates from the Vultr blog!

David Deploy +

VPC 2.0 Networks

+ Add VPC 2.0 Network

ID	Description	Location	Subnet
309f9810-f794-4690-ab35-7b58dc9c93f1	Zic Network	Manchester	192.168.40.0/24

× Frequently Asked Questions

× What is happening to Vultr VPC 2.0?

Vultr has decided to retire VPC 2.0 and consolidate its features into the original Vultr VPC. This means VPC 2.0 will be phased out, and all users will transition to the enhanced Vultr VPC platform.

+ Why is Vultr making this change?

+ How will this transition affect my existing VPC 2.0 configurations?

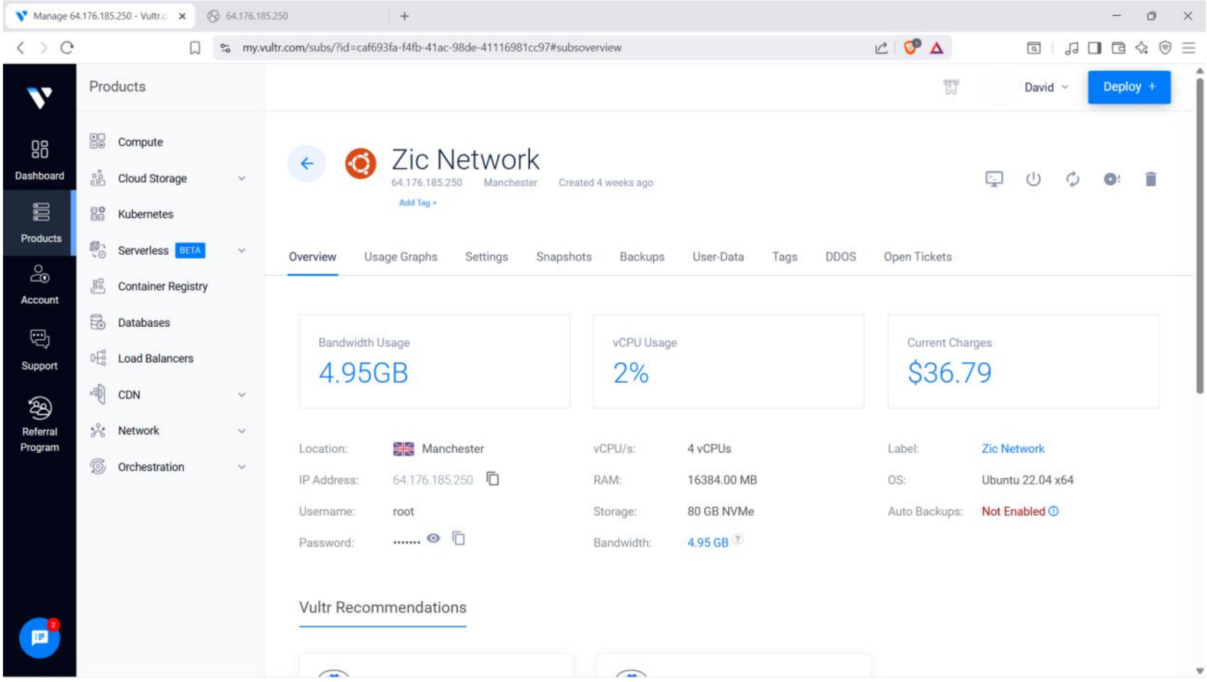
+ What new features can I expect from the enhanced Vultr VPC in 2025?

+ Will there be support during the migration process?

+ When will VPC 2.0 be officially phased out?

+ How can I prepare for this transition?

After creating the VM, you will have an interface with username and password like this.

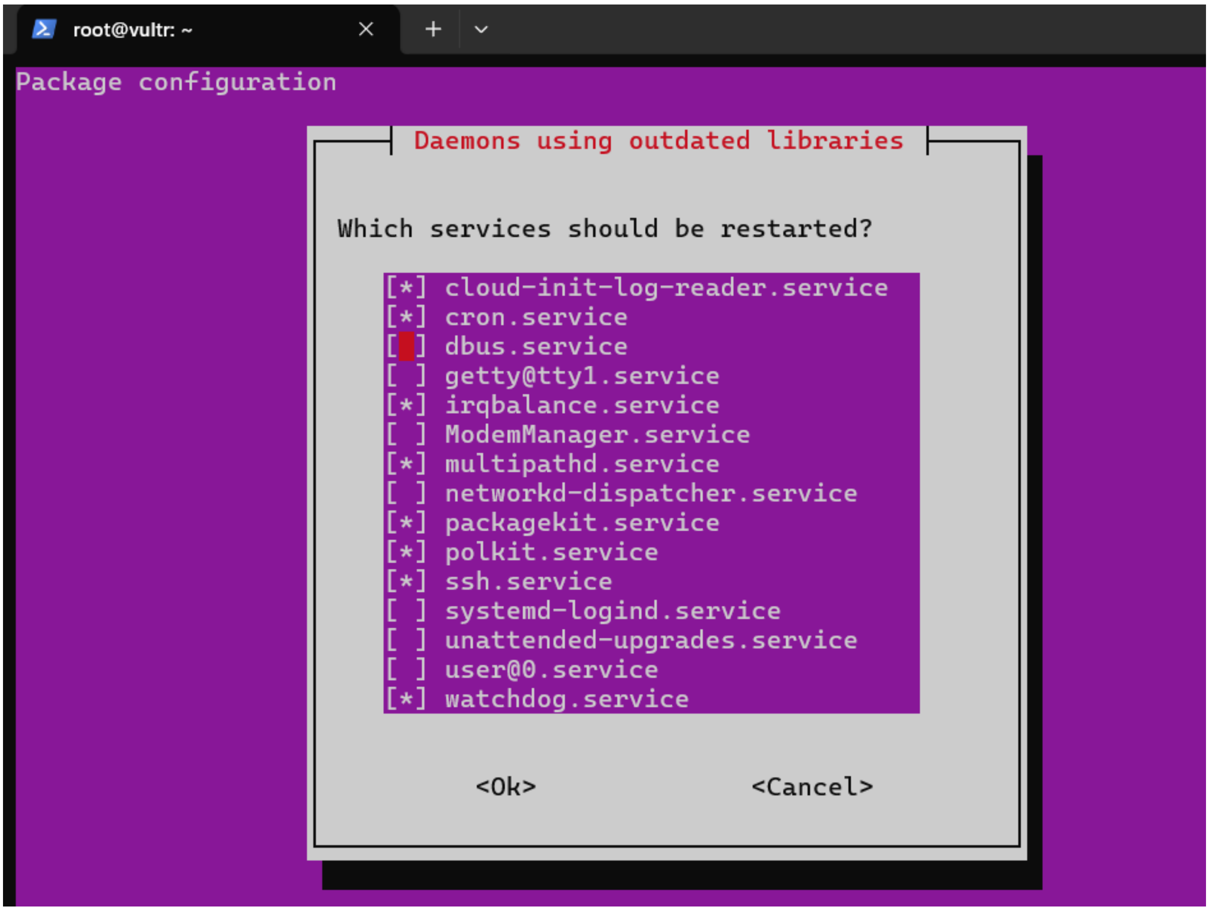


Zic Network
64.176.185.250 Manchester Created 4 weeks ago

Bandwidth Usage 4.95GB
vCPU Usage 2%
Current Charges $36.79

Location: Manchester
IP Address: 64.176.185.250
Username: root
Password: •••••••
vCPU/s: 4 vCPUs
RAM: 16384.00 MB
Storage: 80 GB NVMe
Bandwidth: 4.95 GB
Label: Zic Network
OS: Ubuntu 22.04 x64
Auto Backups: Not Enabled

You will then use the details to login to your windows powershell using the command **ssh root@your generated IP**, then you continue the process to carry out package installation.



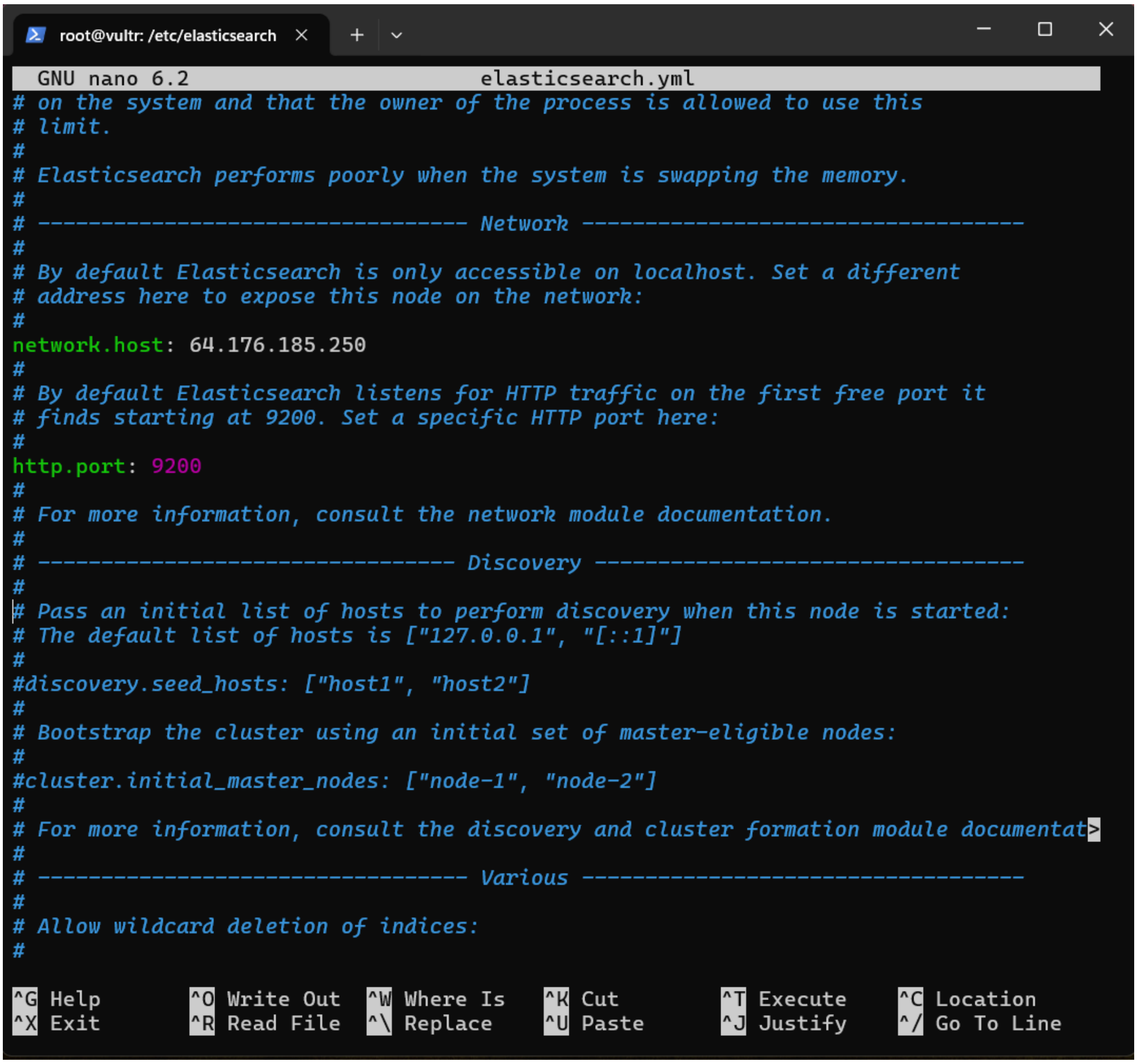
```
Package configuration

        Daemons using outdated libraries

   Which services should be restarted?

       [*] cloud-init-log-reader.service
       [*] cron.service
       [ ] dbus.service
       [ ] getty@tty1.service
       [*] irqbalance.service
       [ ] ModemManager.service
       [*] multipathd.service
       [ ] networkd-dispatcher.service
       [*] packagekit.service
       [*] polkit.service
       [*] ssh.service
       [ ] systemd-logind.service
       [ ] unattended-upgrades.service
       [ ] user@0.service
       [*] watchdog.service

            <Ok>                <Cancel>
```

When done with the package installation, you upgrade and update the repositories using the command **apt-get update && apt-get upgrade –y**.

To download elasticsearch for ubuntu, use this command **wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-9.1.3-amd64.deb**.

To install, you use the command **dpkg –i elasticsearch-9.1.3-amd64.deb.** after installation, you will get your generated password for your elastic search, do not **forget** to save the password somewhere.

Go into the directory using **cd/etc/elasticsearch and list.** You will see **elasticsearch.yml** file. Use the command **nano elasticsearch.yml** to go into the file, edit the local host to the IP address of the Ubuntu Vm you just created on the cloud.



```
  GNU nano 6.2                      elasticsearch.yml
# on the system and that the owner of the process is allowed to use this
# limit.
#
# Elasticsearch performs poorly when the system is swapping the memory.
#
# ---------------------------------- Network -----------------------------------
#
# By default Elasticsearch is only accessible on localhost. Set a different
# address here to expose this node on the network:
#
network.host: 64.176.185.250
#
# By default Elasticsearch listens for HTTP traffic on the first free port it
# finds starting at 9200. Set a specific HTTP port here:
#
http.port: 9200
#
# For more information, consult the network module documentation.
#
# --------------------------------- Discovery ----------------------------------
#
# Pass an initial list of hosts to perform discovery when this node is started:
# The default list of hosts is ["127.0.0.1", "[::1]"]
#
#discovery.seed_hosts: ["host1", "host2"]
#
# Bootstrap the cluster using an initial set of master-eligible nodes:
#
#cluster.initial_master_nodes: ["node-1", "node-2"]
#
# For more information, consult the discovery and cluster formation module documentat>
#
# ---------------------------------- Various -----------------------------------
#
# Allow wildcard deletion of indices:
#

^G Help         ^O Write Out    ^W Where Is     ^K Cut          ^T Execute      ^C Location
^X Exit         ^R Read File    ^\ Replace      ^U Paste        ^J Justify      ^/ Go To Line
```

After the edit, you go to your vultr, go to the Ubuntu Vm created, mine is **Zic Network.**

Head to the settings, then firewall. Create a name for the firewall group and create a firewall rule. **SSH using port 22 and the dropdown of source.**

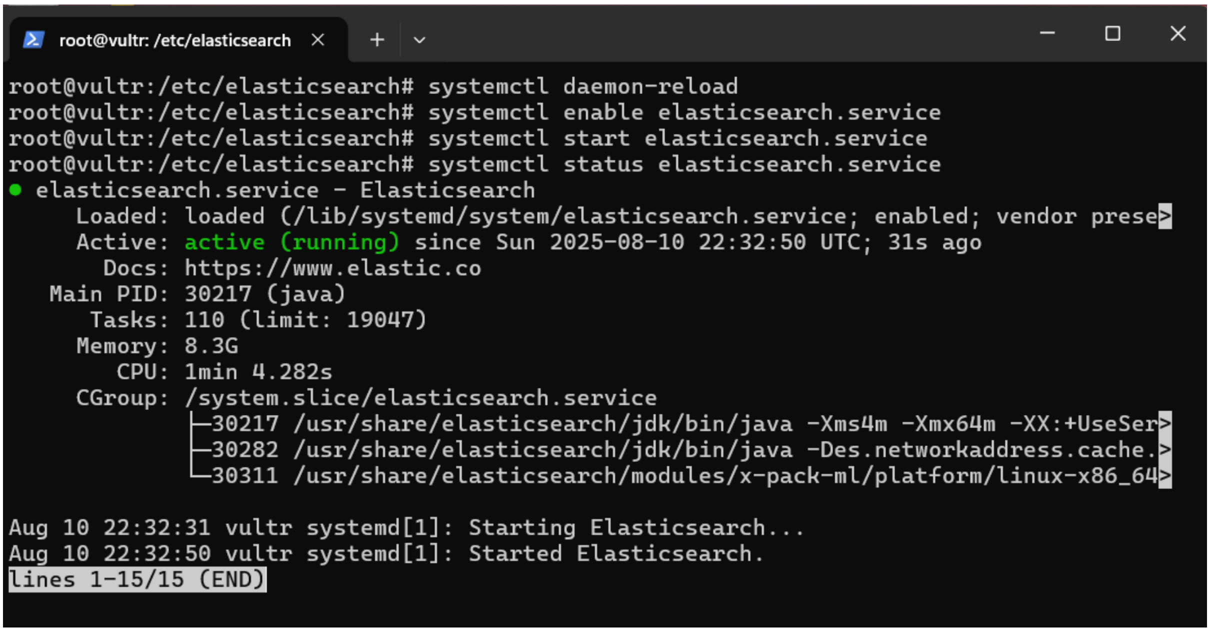
In the elasticsearch directory, use the following commands.

Systemctl daemon-reload

Systemctl enable elasticsearch.service

Systemctl start elasticsearch.service

Systemctl status elasticsearch.service



```
root@vultr:/etc/elasticsearch# systemctl daemon-reload
root@vultr:/etc/elasticsearch# systemctl enable elasticsearch.service
root@vultr:/etc/elasticsearch# systemctl start elasticsearch.service
root@vultr:/etc/elasticsearch# systemctl status elasticsearch.service
● elasticsearch.service - Elasticsearch
     Loaded: loaded (/lib/systemd/system/elasticsearch.service; enabled; vendor prese>
     Active: active (running) since Sun 2025-08-10 22:32:50 UTC; 31s ago
       Docs: https://www.elastic.co
   Main PID: 30217 (java)
      Tasks: 110 (limit: 19047)
     Memory: 8.3G
        CPU: 1min 4.282s
     CGroup: /system.slice/elasticsearch.service
             ├─30217 /usr/share/elasticsearch/jdk/bin/java -Xms4m -Xmx64m -XX:+UseSer>
             ├─30282 /usr/share/elasticsearch/jdk/bin/java -Des.networkaddress.cache.>
             └─30311 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x86_64>

Aug 10 22:32:31 vultr systemd[1]: Starting Elasticsearch...
Aug 10 22:32:50 vultr systemd[1]: Started Elasticsearch.
lines 1-15/15 (END)
```

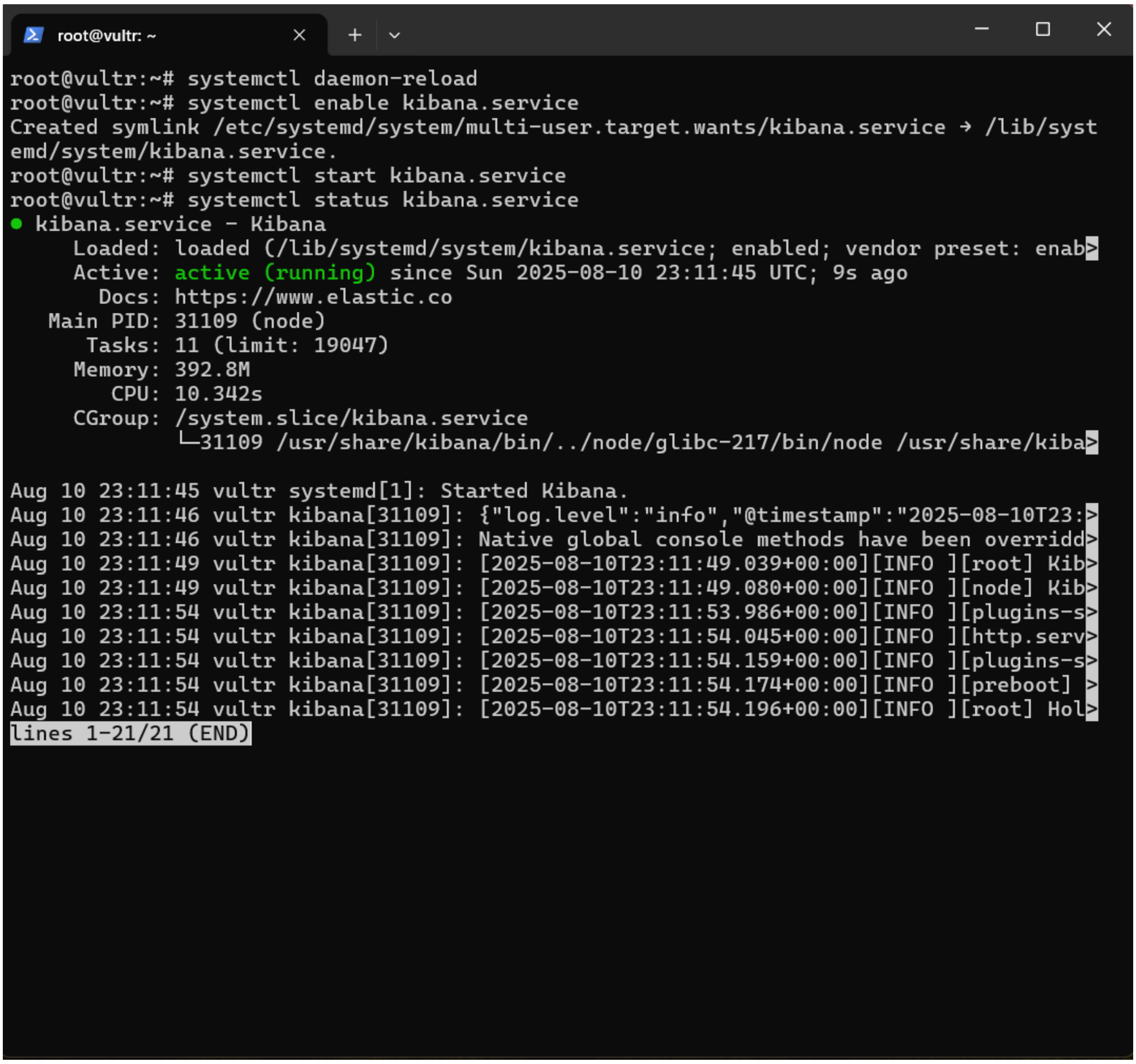
You have successfully installed elasticsearch and from the screenshot above, it shows it is running already.

KIBANA SETUP

To Install **KIBANA,** repeat the same process for elasticsearch. Go to the website and select platform for **deb x86_64** (this is for ubuntu).

Use the command **wget https://artifacts.elastic.co/downloads/kibana/kibana-9.1.3-amd64.deb.**

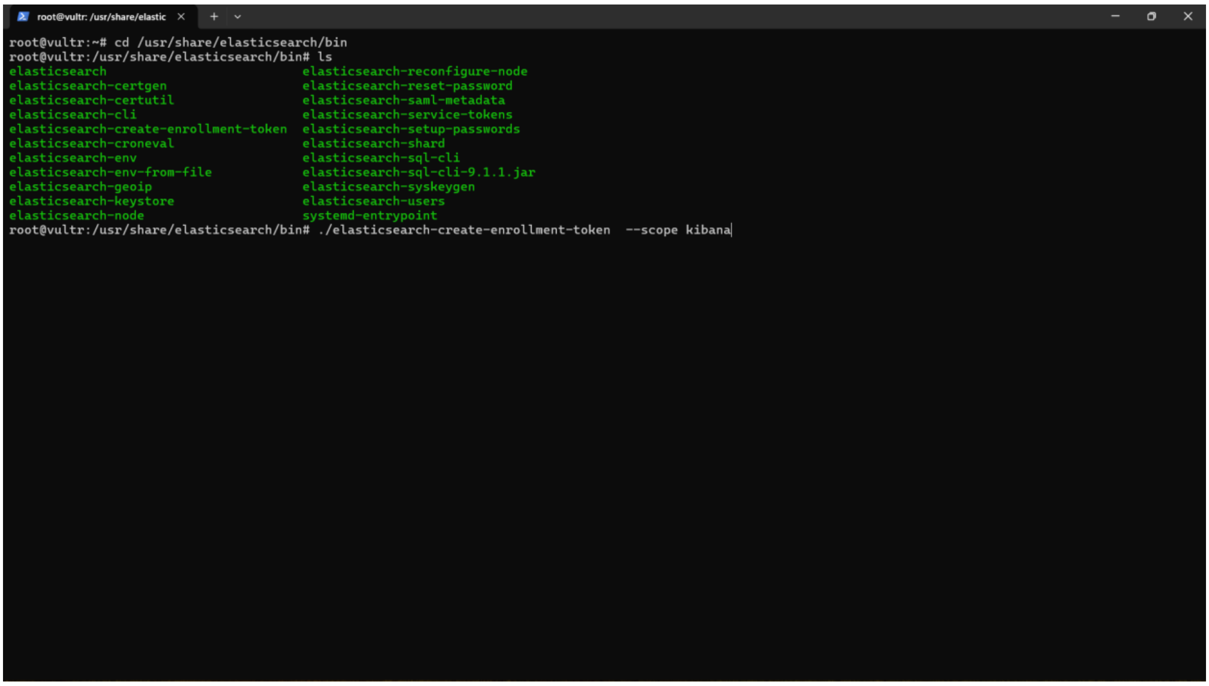
At the end of the process, you will see that kibana is also active and running successfully on elastic.



```
root@vultr:~# systemctl daemon-reload
root@vultr:~# systemctl enable kibana.service
Created symlink /etc/systemd/system/multi-user.target.wants/kibana.service → /lib/syst
emd/system/kibana.service.
root@vultr:~# systemctl start kibana.service
root@vultr:~# systemctl status kibana.service
● kibana.service - Kibana
     Loaded: loaded (/lib/systemd/system/kibana.service; enabled; vendor preset: enab>
     Active: active (running) since Sun 2025-08-10 23:11:45 UTC; 9s ago
       Docs: https://www.elastic.co
   Main PID: 31109 (node)
      Tasks: 11 (limit: 19047)
     Memory: 392.8M
        CPU: 10.342s
     CGroup: /system.slice/kibana.service
             └─31109 /usr/share/kibana/bin/../node/glibc-217/bin/node /usr/share/kiba>

Aug 10 23:11:45 vultr systemd[1]: Started Kibana.
Aug 10 23:11:46 vultr kibana[31109]: {"log.level":"info","@timestamp":"2025-08-10T23:>
Aug 10 23:11:46 vultr kibana[31109]: Native global console methods have been overridd>
Aug 10 23:11:49 vultr kibana[31109]: [2025-08-10T23:11:49.039+00:00][INFO ][root] Kib>
Aug 10 23:11:49 vultr kibana[31109]: [2025-08-10T23:11:49.080+00:00][INFO ][node] Kib>
Aug 10 23:11:54 vultr kibana[31109]: [2025-08-10T23:11:53.986+00:00][INFO ][plugins-s>
Aug 10 23:11:54 vultr kibana[31109]: [2025-08-10T23:11:54.045+00:00][INFO ][http.serv>
Aug 10 23:11:54 vultr kibana[31109]: [2025-08-10T23:11:54.159+00:00][INFO ][plugins-s>
Aug 10 23:11:54 vultr kibana[31109]: [2025-08-10T23:11:54.174+00:00][INFO ][preboot] >
Aug 10 23:11:54 vultr kibana[31109]: [2025-08-10T23:11:54.196+00:00][INFO ][root] Hol>
lines 1-21/21 (END)
```

After the kibana has been setup, you create an enrollment token using the command

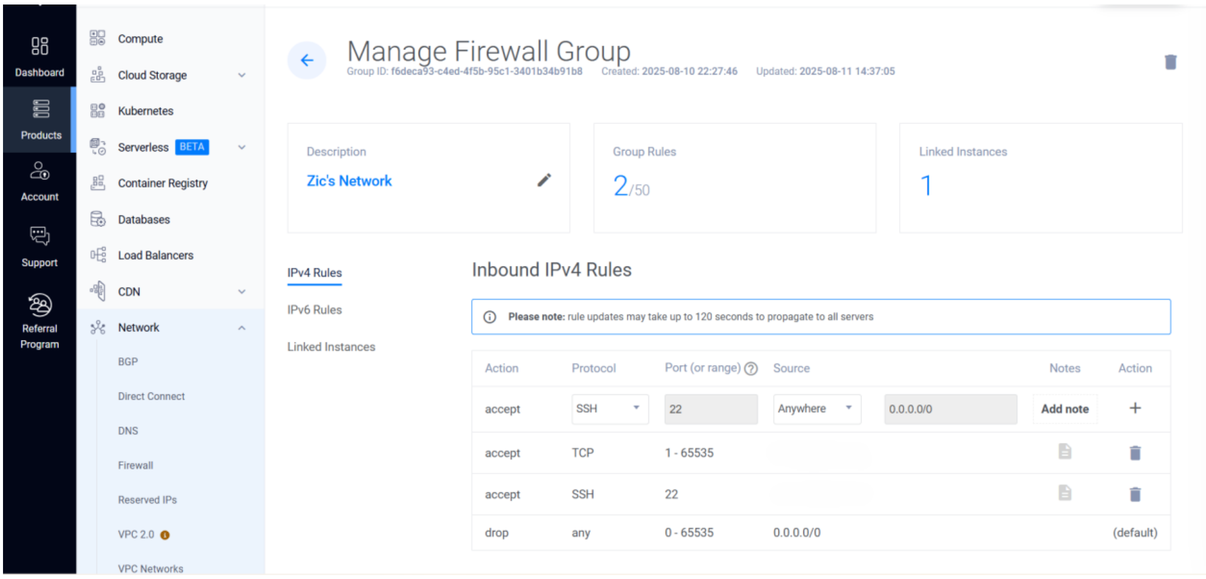
cd /usr/share/elasticsearch/bin/elasticsearch-create-enrollment-token –scope kibana. This will generate a token for you to paste in kibana itself.



```
root@vultr:~# cd /usr/share/elasticsearch/bin
root@vultr:/usr/share/elasticsearch/bin# ls
elasticsearch                         elasticsearch-reconfigure-node
elasticsearch-certgen                 elasticsearch-reset-password
elasticsearch-certutil                elasticsearch-saml-metadata
elasticsearch-cli                     elasticsearch-service-tokens
elasticsearch-create-enrollment-token elasticsearch-setup-passwords
elasticsearch-croneval                elasticsearch-shard
elasticsearch-env                     elasticsearch-sql-cli
elasticsearch-env-from-file           elasticsearch-sql-cli-9.1.1.jar
elasticsearch-geoip                   elasticsearch-syskeygen
elasticsearch-keystore                elasticsearch-users
elasticsearch-node                    systemd-entrypoint
root@vultr:/usr/share/elasticsearch/bin# ./elasticsearch-create-enrollment-token  --scope kibana
```

To login into elasticsearch, use **yourgeneratedip:5601.** You might get some errors, in the powershell of the Ubuntu server, allow port 5601 using the command **ufw allow 5601.**

Also create a firewall rule using **TCP**, all ports **(1-65535)** and to your IP address in the dropdown of **Source.**



Manage Firewall Group

Group ID: f6deca93-c4ed-4f5b-95c1-3401b34b91b8 Created: 2025-08-10 22:27:46 Updated: 2025-08-11 14:37:05

Description: Zic's Network

Group Rules: 2/50

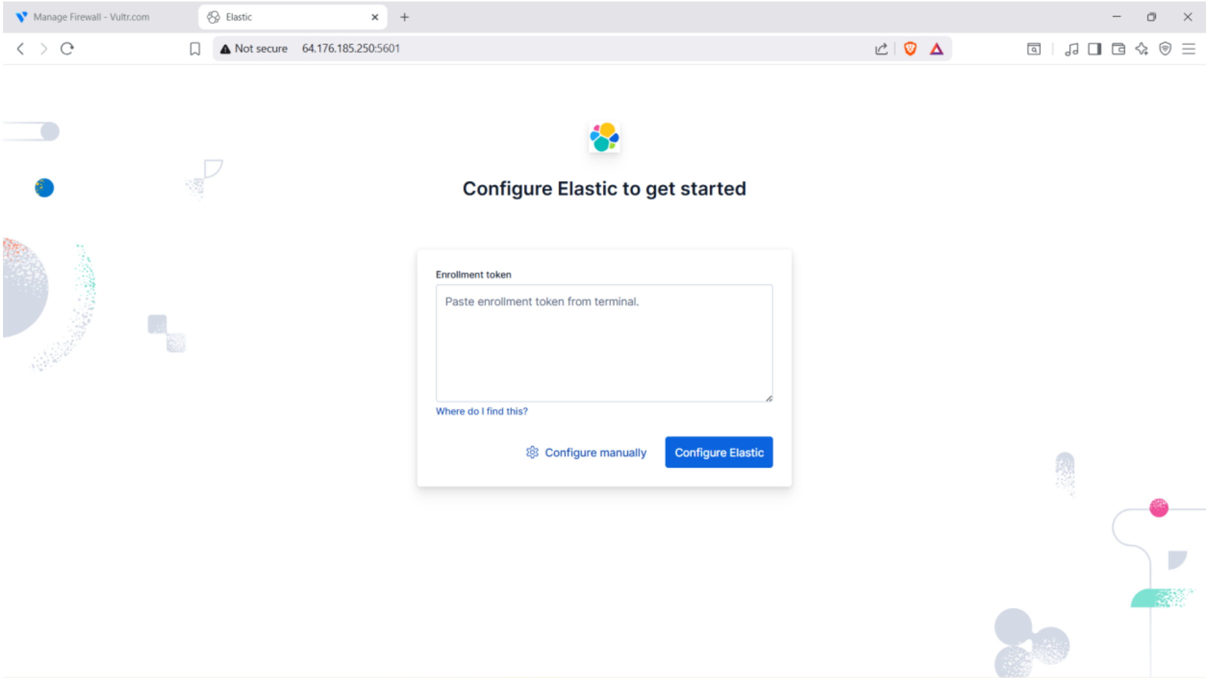
Linked Instances: 1

Inbound IPv4 Rules

Please note: rule updates may take up to 120 seconds to propagate to all servers

Action	Protocol	Port (or range)	Source		Notes	Action
accept	SSH	22	Anywhere	0.0.0.0/0	Add note	+
accept	TCP	1 - 65535				
accept	SSH	22				
drop	any	0 - 65535	0.0.0.0/0			(default)

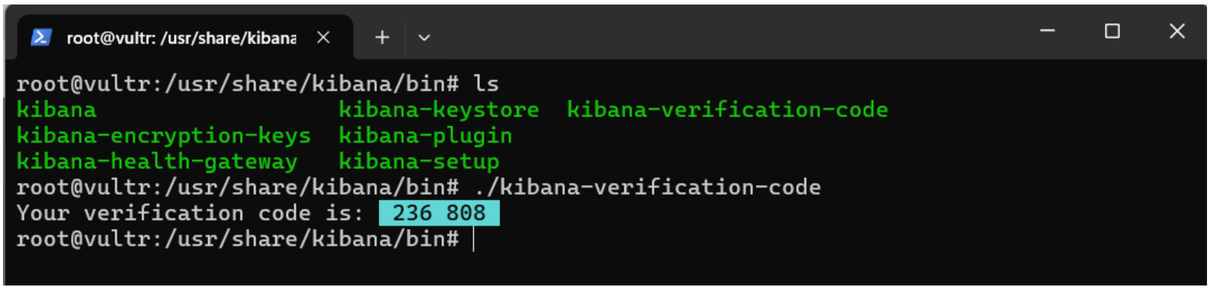
When presented with the elasticsearch page, you need to configure the elasticsearch with the enrollment token generated earlier, it will request a verification code and the dir is **/usr/share/kibana/bin/** and you go to the binary **kibana-verification-code** and see the code here.



Configure Elastic to get started

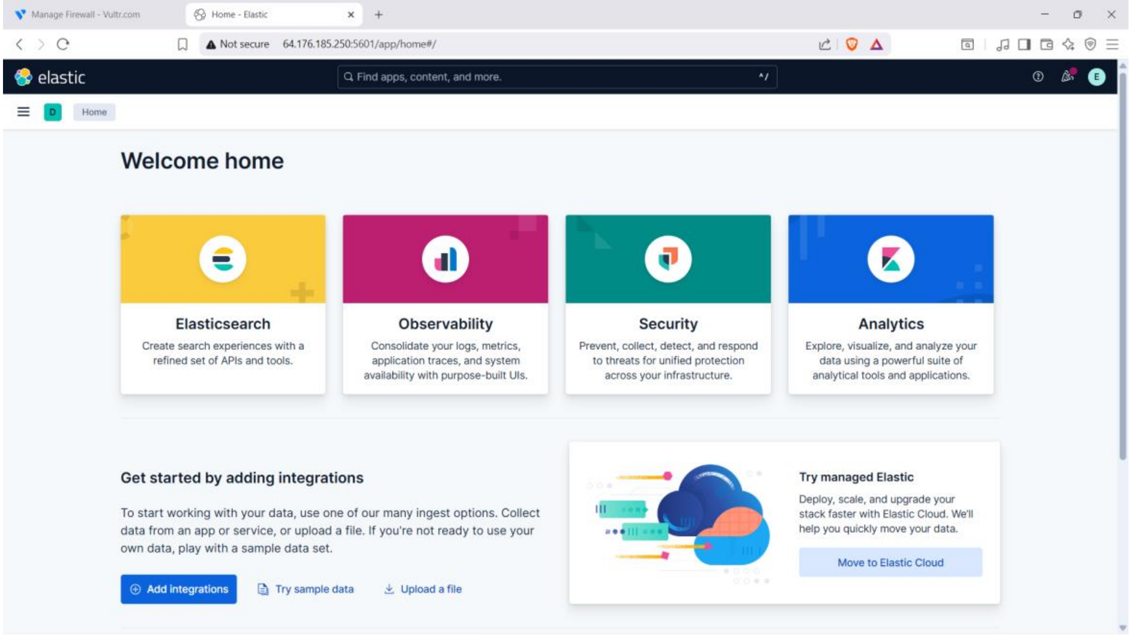
Enrollment token

Paste enrollment token from terminal.



```
root@vultr:/usr/share/kibana/bin# ls
kibana                  kibana-keystore  kibana-verification-code
kibana-encryption-keys  kibana-plugin
kibana-health-gateway   kibana-setup
root@vultr:/usr/share/kibana/bin# ./kibana-verification-code
Your verification code is:  236 808
root@vultr:/usr/share/kibana/bin#
```

You use the default username **elastic** and the generated password you saved earlier.


Manage Firewall - Vultr.com
Home - Elastic
Not secure 64.176.185.250:5601/app/home#/
elastic
Find apps, content, and more.
D
Home

Welcome home

Elasticsearch

Create search experiences with a refined set of APIs and tools.

Observability

Consolidate your logs, metrics, application traces, and system availability with purpose-built UIs.

Security

Prevent, collect, detect, and respond to threats for unified protection across your infrastructure.

Analytics

Explore, visualize, and analyze your data using a powerful suite of analytical tools and applications.

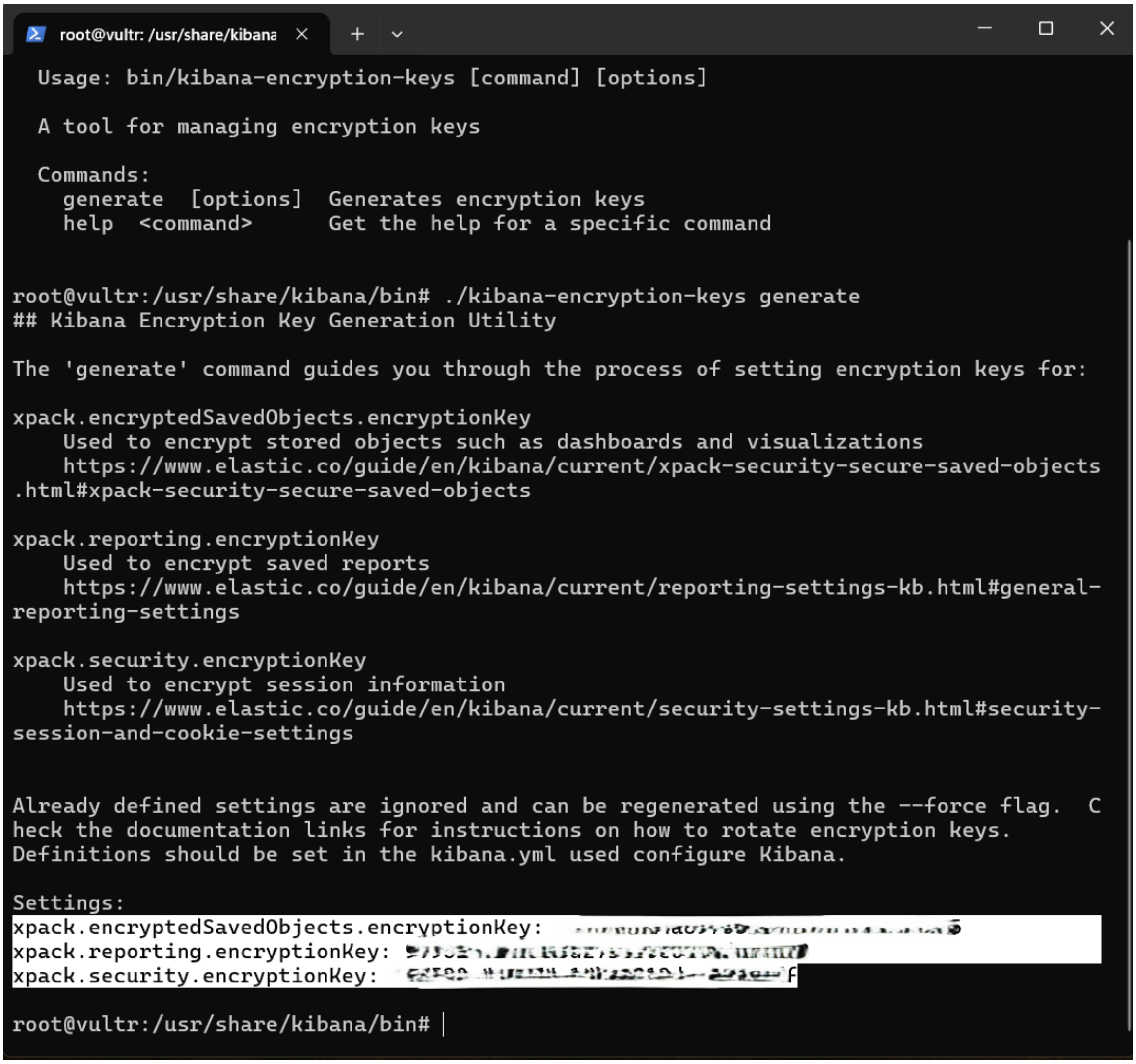
Get started by adding integrations

To start working with your data, use one of our many ingest options. Collect data from an app or service, or upload a file. If you're not ready to use your own data, play with a sample data set.

Try managed Elastic

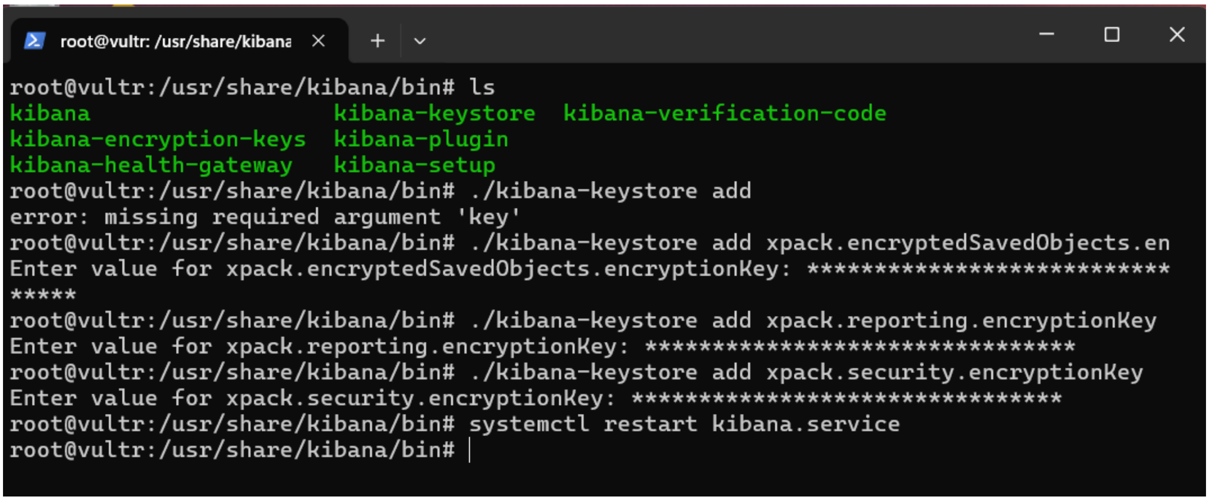
Deploy, scale, and upgrade your stack faster with Elastic Cloud. We'll help you quickly move your data.

After logging in, you need to integrate the API key, you go to your kibana directory and include the binary **./kibana-encryption-keys generate**. This will generate xpack keys.



```
  Usage: bin/kibana-encryption-keys [command] [options]

  A tool for managing encryption keys

  Commands:
    generate  [options]  Generates encryption keys
    help  <command>      Get the help for a specific command


root@vultr:/usr/share/kibana/bin# ./kibana-encryption-keys generate
## Kibana Encryption Key Generation Utility

The 'generate' command guides you through the process of setting encryption keys for:

xpack.encryptedSavedObjects.encryptionKey
    Used to encrypt stored objects such as dashboards and visualizations
    https://www.elastic.co/guide/en/kibana/current/xpack-security-secure-saved-objects
.html#xpack-security-secure-saved-objects

xpack.reporting.encryptionKey
    Used to encrypt saved reports
    https://www.elastic.co/guide/en/kibana/current/reporting-settings-kb.html#general-
reporting-settings

xpack.security.encryptionKey
    Used to encrypt session information
    https://www.elastic.co/guide/en/kibana/current/security-settings-kb.html#security-
session-and-cookie-settings


Already defined settings are ignored and can be regenerated using the --force flag.  C
heck the documentation links for instructions on how to rotate encryption keys.
Definitions should be set in the kibana.yml used configure Kibana.

Settings:
xpack.encryptedSavedObjects.encryptionKey: [redacted]
xpack.reporting.encryptionKey:
xpack.security.encryptionKey:

root@vultr:/usr/share/kibana/bin# 
```

Save the keys in a notepad and use the binary **./kibana-keystore.** Add the field names of the keys individually before adding the encryption keys. Do this process for the 3 xpack encryption keys.



```
root@vultr:/usr/share/kibana/bin# ls
kibana                    kibana-keystore  kibana-verification-code
kibana-encryption-keys    kibana-plugin
kibana-health-gateway     kibana-setup
root@vultr:/usr/share/kibana/bin# ./kibana-keystore add
error: missing required argument 'key'
root@vultr:/usr/share/kibana/bin# ./kibana-keystore add xpack.encryptedSavedObjects.en
Enter value for xpack.encryptedSavedObjects.encryptionKey: ****************************
*****
root@vultr:/usr/share/kibana/bin# ./kibana-keystore add xpack.reporting.encryptionKey
Enter value for xpack.reporting.encryptionKey: **********************************
root@vultr:/usr/share/kibana/bin# ./kibana-keystore add xpack.security.encryptionKey
Enter value for xpack.security.encryptionKey: **********************************
root@vultr:/usr/share/kibana/bin# systemctl restart kibana.service
root@vultr:/usr/share/kibana/bin# 
```

Ensure to restart kibana when done using the command **systemctl restart kibana.service**.

Challenges faced.

- Configuring VPC and firewall rules correctly to avoid access issues.
- Remembering to save and manage generated passwords and tokens.
- Editing configuration files (elasticsearch.yml and Kibana settings) properly to match the server IP.
- Handling installation errors due to missing dependencies or incorrect package versions.
- Network access errors such as blocked ports (e.g., port 5601 for Kibana).

What Elasticsearch and Kibana are Used For

- **Elasticsearch**:
  - Stores, indexes, and searches large volumes of data quickly.
  - Provides the backend for log analytics, monitoring, and security detection.
  - Supports advanced queries for threat hunting and data correlation.
- **Kibana**:
  - Acts as the visualization layer for Elasticsearch data.
  - Provides dashboards, charts, and search capabilities.
  - Enables monitoring, alerting, and security event investigation.