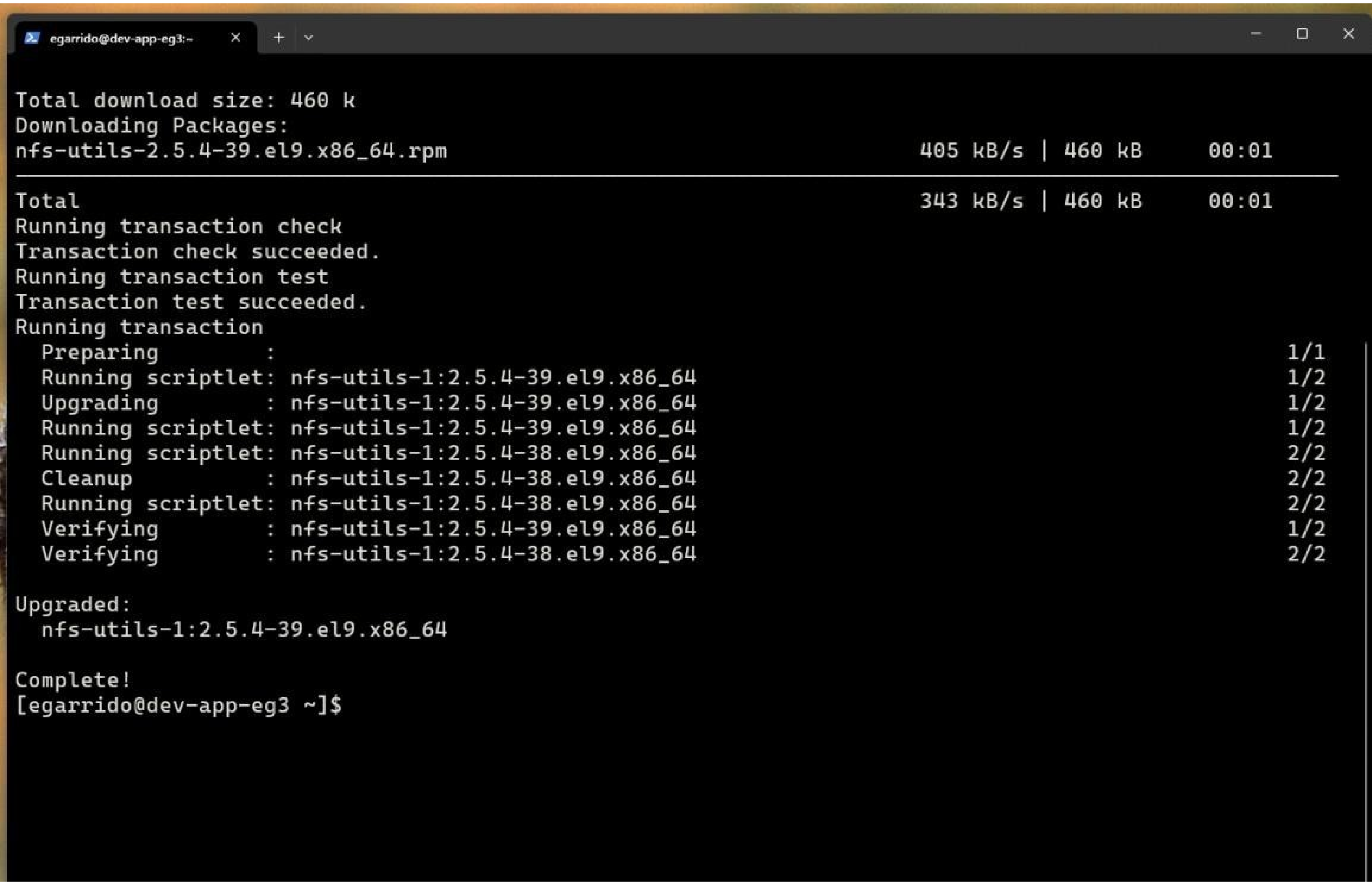
Shared NFS Storage Deployment and Cross-Host Validation

This work centered on building and validating a shared network file system to support cross-host access to application artifacts and installation media. Required NFS components were installed and updated to ensure compatibility and stability. A dedicated export directory was created with appropriate permissions to allow shared access, and export definitions were configured to provide controlled read-write access to the designated network range.

NFS services were enabled and started to make the export available immediately and persistently. Firewall rules were updated to allow NFS-related traffic, ensuring uninterrupted communication between servers. Export definitions were applied and verified to confirm that the shared directory was successfully published and accessible to authorized clients.

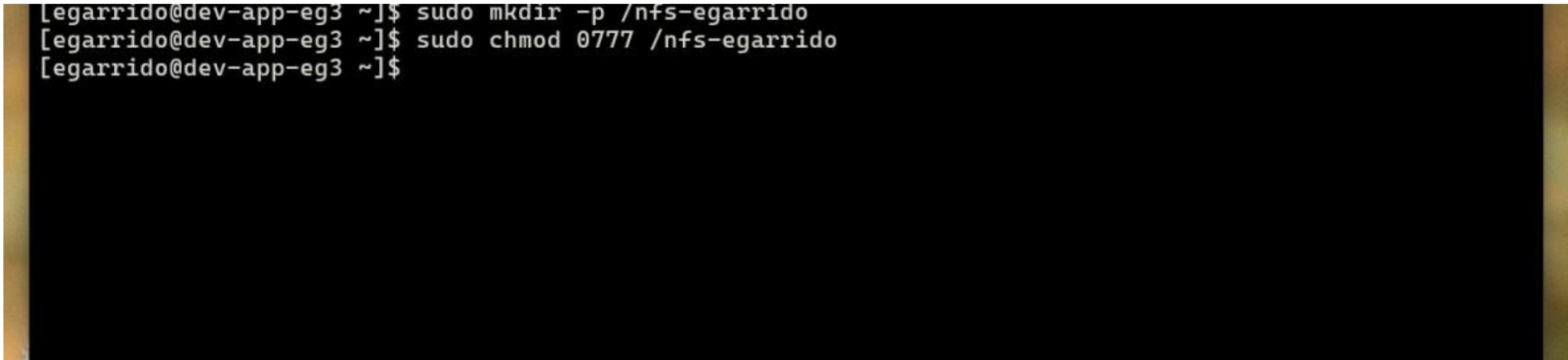
Client-side configuration included creating mount points, mounting the shared export using NFSv4, and validating successful attachment through filesystem and disk usage checks. Data was written to the shared location from one host and verified from another, confirming real-time visibility and consistency across systems. This implementation established a reliable shared storage layer that enables seamless file sharing, supports application dependencies, and ensures consistent access across development and staging environments.

The NFS utilities package is successfully upgraded on the system, completing all transaction checks and scriptlet runs, ensuring the host has the required and up-to-date components to support network file system mounting and operations.



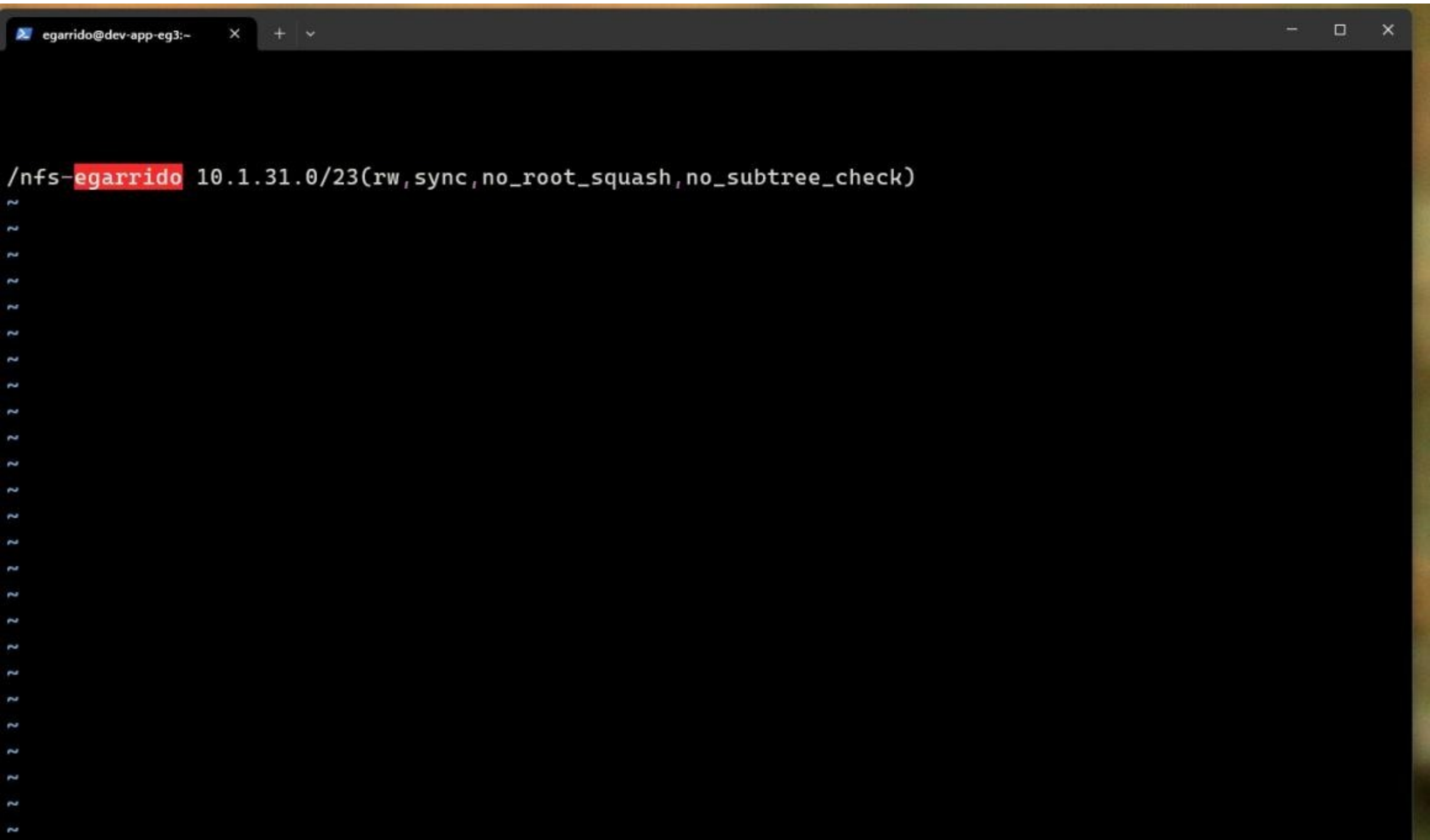
```
Total download size: 460 k
Downloading Packages:
nfs-utils-2.5.4-39.el9.x86_64.rpm                                      405 kB/s | 460 kB     00:01
--------------------------------------------------------------------------------------------------------
Total                                                                  343 kB/s | 460 kB     00:01
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                                                1/1
  Running scriptlet: nfs-utils-1:2.5.4-39.el9.x86_64                                                1/2
  Upgrading        : nfs-utils-1:2.5.4-39.el9.x86_64                                                1/2
  Running scriptlet: nfs-utils-1:2.5.4-39.el9.x86_64                                                1/2
  Running scriptlet: nfs-utils-1:2.5.4-38.el9.x86_64                                                2/2
  Cleanup          : nfs-utils-1:2.5.4-38.el9.x86_64                                                2/2
  Running scriptlet: nfs-utils-1:2.5.4-38.el9.x86_64                                                2/2
  Verifying        : nfs-utils-1:2.5.4-39.el9.x86_64                                                1/2
  Verifying        : nfs-utils-1:2.5.4-38.el9.x86_64                                                2/2

Upgraded:
  nfs-utils-1:2.5.4-39.el9.x86_64

Complete!
[egarrido@dev-app-eg3 ~]$
```

A new directory is created at /nfs-egarrido to serve as a mount or shared location on the system. Permissions are then set to allow full read, write, and execute access for all users, ensuring the directory is accessible for testing or shared use as required.



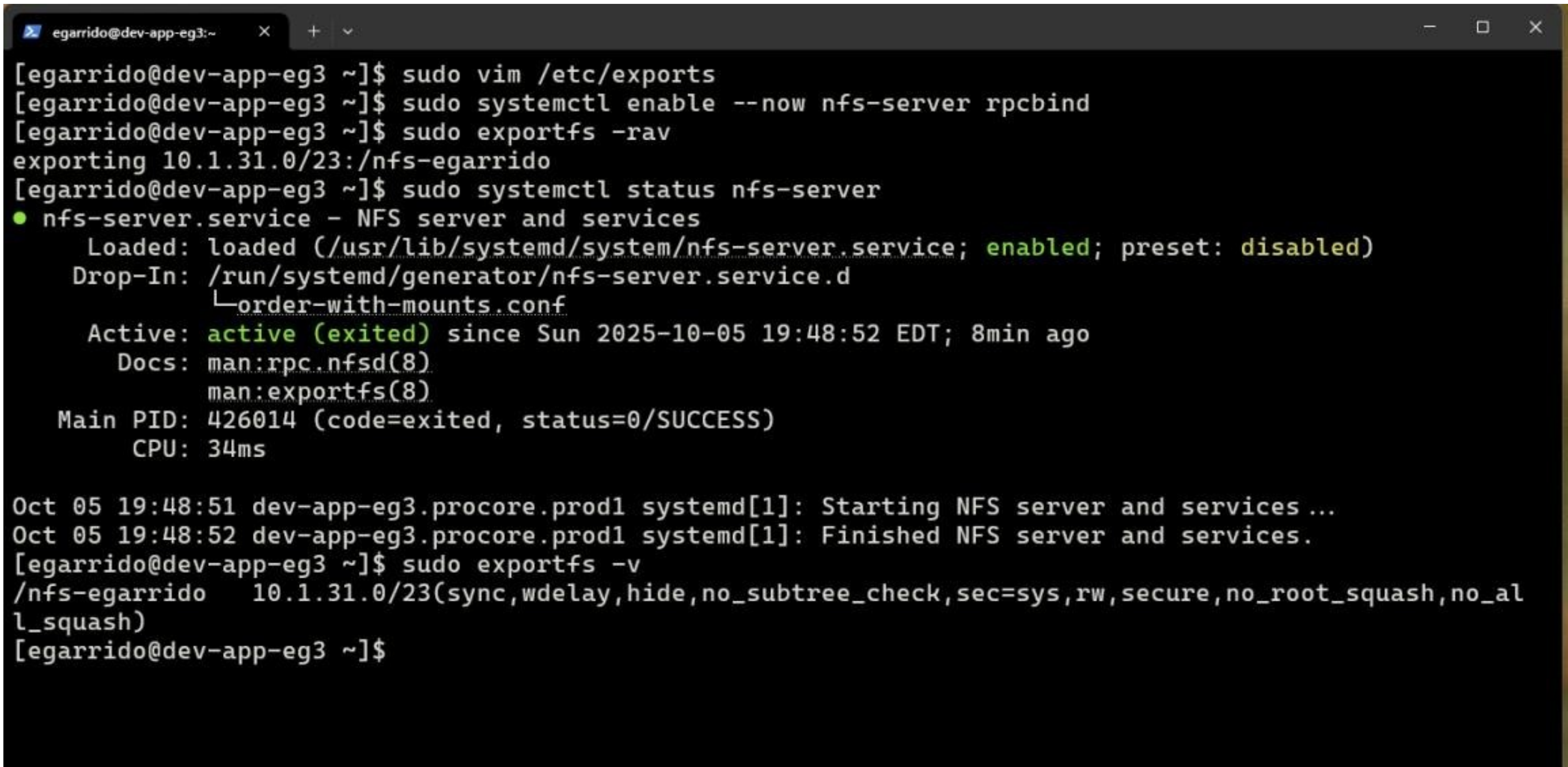
```
[egarrido@dev-app-eg3 ~]$ sudo mkdir -p /nfs-egarrido
[egarrido@dev-app-eg3 ~]$ sudo chmod 0777 /nfs-egarrido
[egarrido@dev-app-eg3 ~]$
```

An export entry is added to the NFS configuration to share the /nfs-egarrido directory with the specified network range. The export is configured for read-write access, synchronous writes, disabled root squashing, and no subtree checking, ensuring clients on the allowed subnet can access the share with the required permissions and consistent behavior.



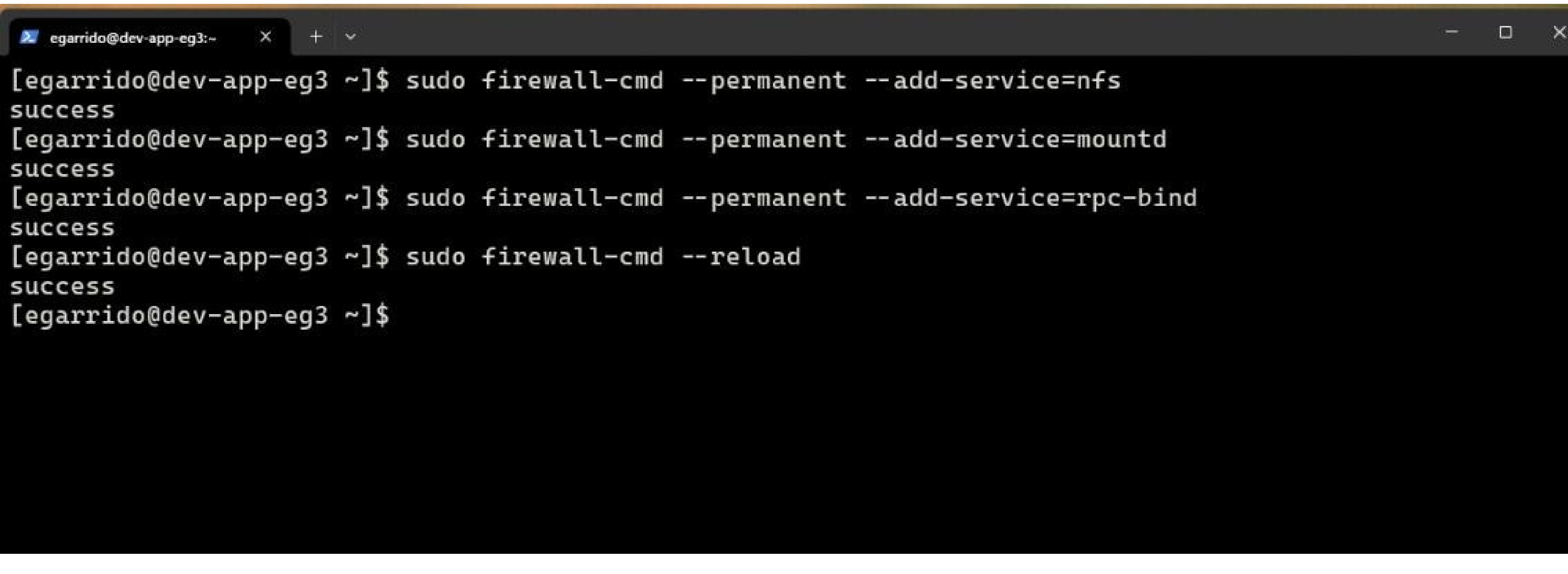
```
/nfs-egarrido 10.1.31.0/23(rw,sync,no_root_squash,no_subtree_check)
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```

The NFS export configuration is finalized by enabling and starting the required NFS services, applying the updated export definitions, and validating their active state. The export is successfully published to the specified network range, and service status confirms that the NFS server is operational. A verification of the active exports confirms the shared directory is available with the intended access options and permissions, ensuring clients can mount and use the share reliably.



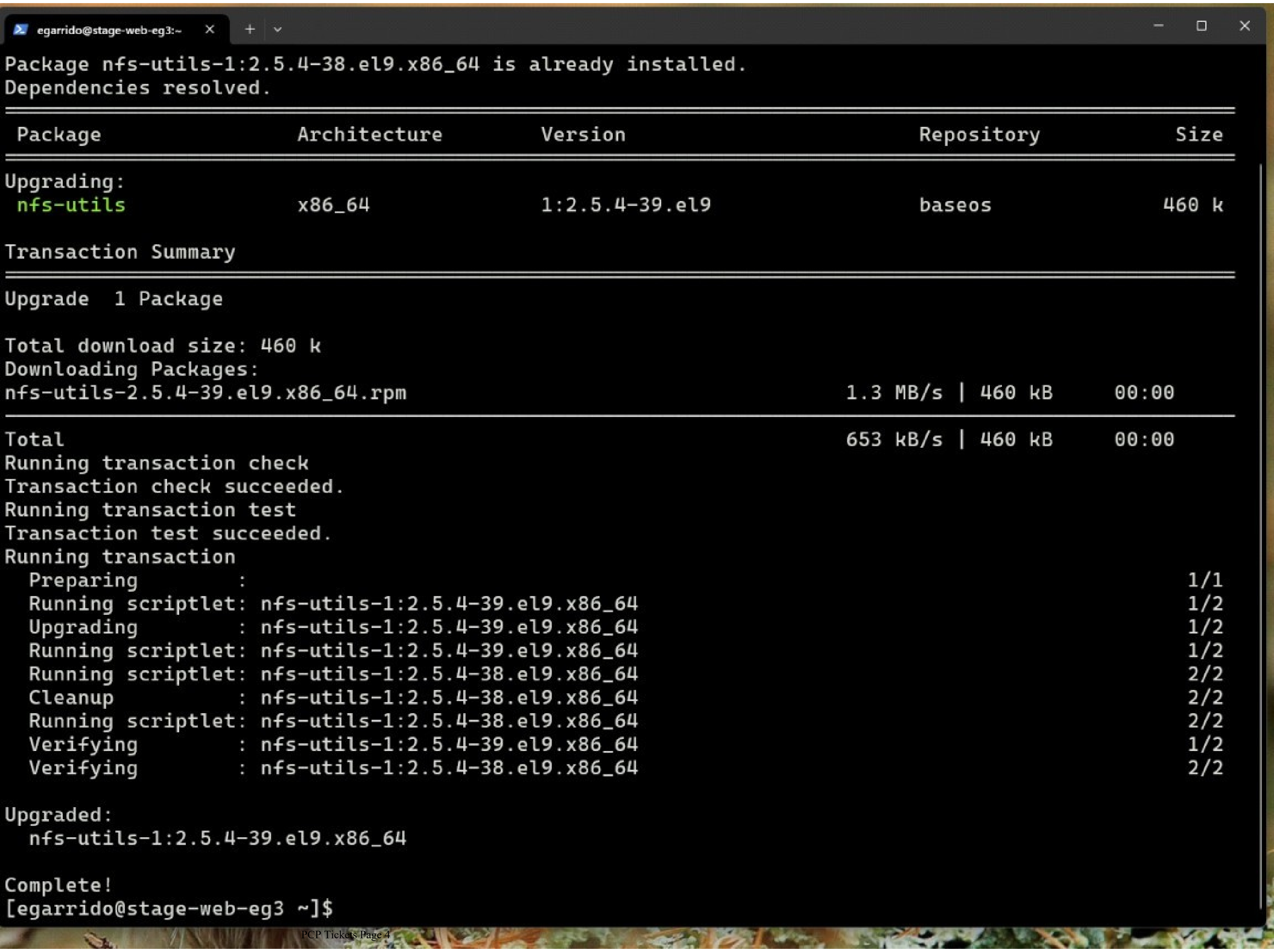
```
[egarrido@dev-app-eg3 ~]$ sudo vim /etc/exports
[egarrido@dev-app-eg3 ~]$ sudo systemctl enable --now nfs-server rpcbind
[egarrido@dev-app-eg3 ~]$ sudo exportfs -rav
exporting 10.1.31.0/23:/nfs-egarrido
[egarrido@dev-app-eg3 ~]$ sudo systemctl status nfs-server
● nfs-server.service - NFS server and services
     Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; enabled; preset: disabled)
    Drop-In: /run/systemd/generator/nfs-server.service.d
             └─order-with-mounts.conf
     Active: active (exited) since Sun 2025-10-05 19:48:52 EDT; 8min ago
       Docs: man:rpc.nfsd(8)
             man:exportfs(8)
   Main PID: 426014 (code=exited, status=0/SUCCESS)
        CPU: 34ms

Oct 05 19:48:51 dev-app-eg3.procore.prod1 systemd[1]: Starting NFS server and services...
Oct 05 19:48:52 dev-app-eg3.procore.prod1 systemd[1]: Finished NFS server and services.
[egarrido@dev-app-eg3 ~]$ sudo exportfs -v
/nfs-egarrido   10.1.31.0/23(sync,wdelay,hide,no_subtree_check,sec=sys,rw,secure,no_root_squash,no_al
l_squash)
[egarrido@dev-app-eg3 ~]$
```

Firewall rules are updated to allow NFS-related traffic through the system firewall. The required services for NFS operation are permanently enabled, and the firewall configuration is reloaded to apply the changes. This ensures that NFS clients can successfully communicate with the server and access exported shares without network-level restrictions.



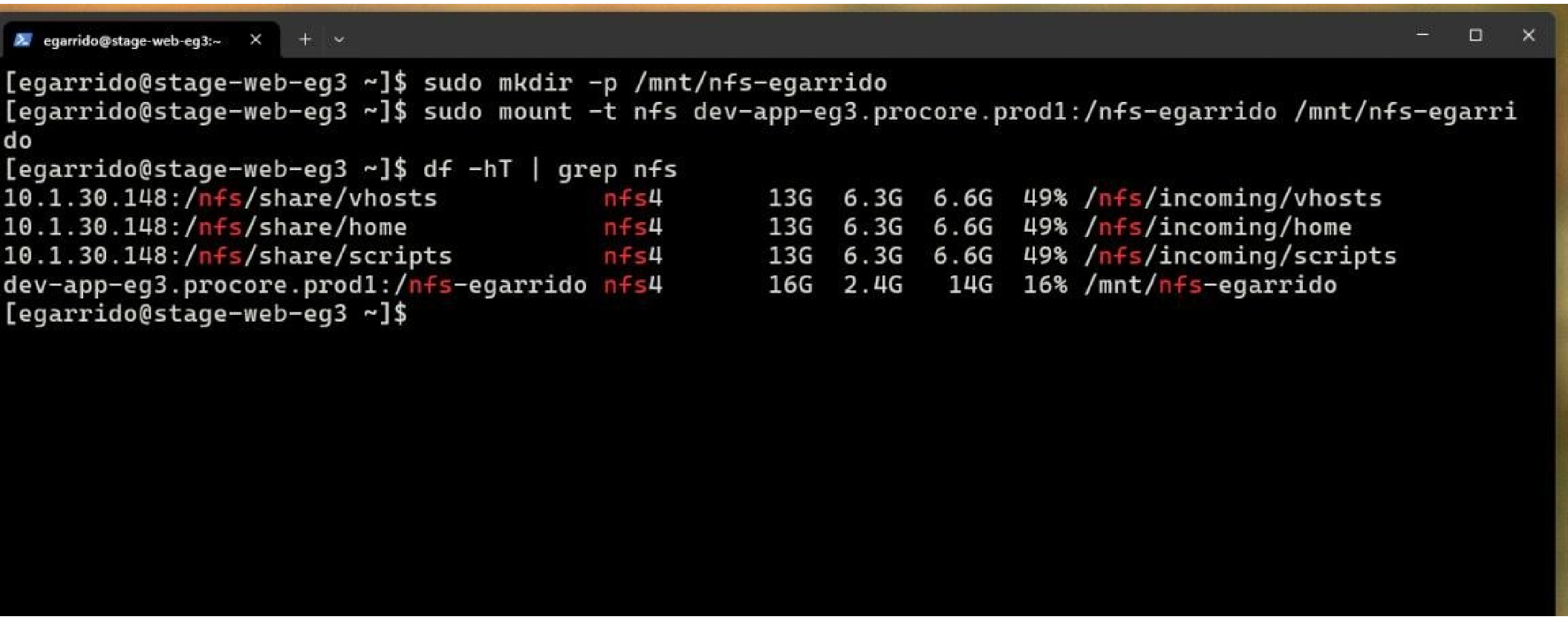
```
[egarrido@dev-app-eg3 ~]$ sudo firewall-cmd --permanent --add-service=nfs
success
[egarrido@dev-app-eg3 ~]$ sudo firewall-cmd --permanent --add-service=mountd
success
[egarrido@dev-app-eg3 ~]$ sudo firewall-cmd --permanent --add-service=rpc-bind
success
[egarrido@dev-app-eg3 ~]$ sudo firewall-cmd --reload
success
[egarrido@dev-app-eg3 ~]$
```

The NFS utilities package is verified and upgraded on the system to the latest available version. Dependency checks, transaction tests, and package verification complete successfully, confirming the host has the required and up-to-date NFS components installed to support mounting and interacting with shared network file systems.



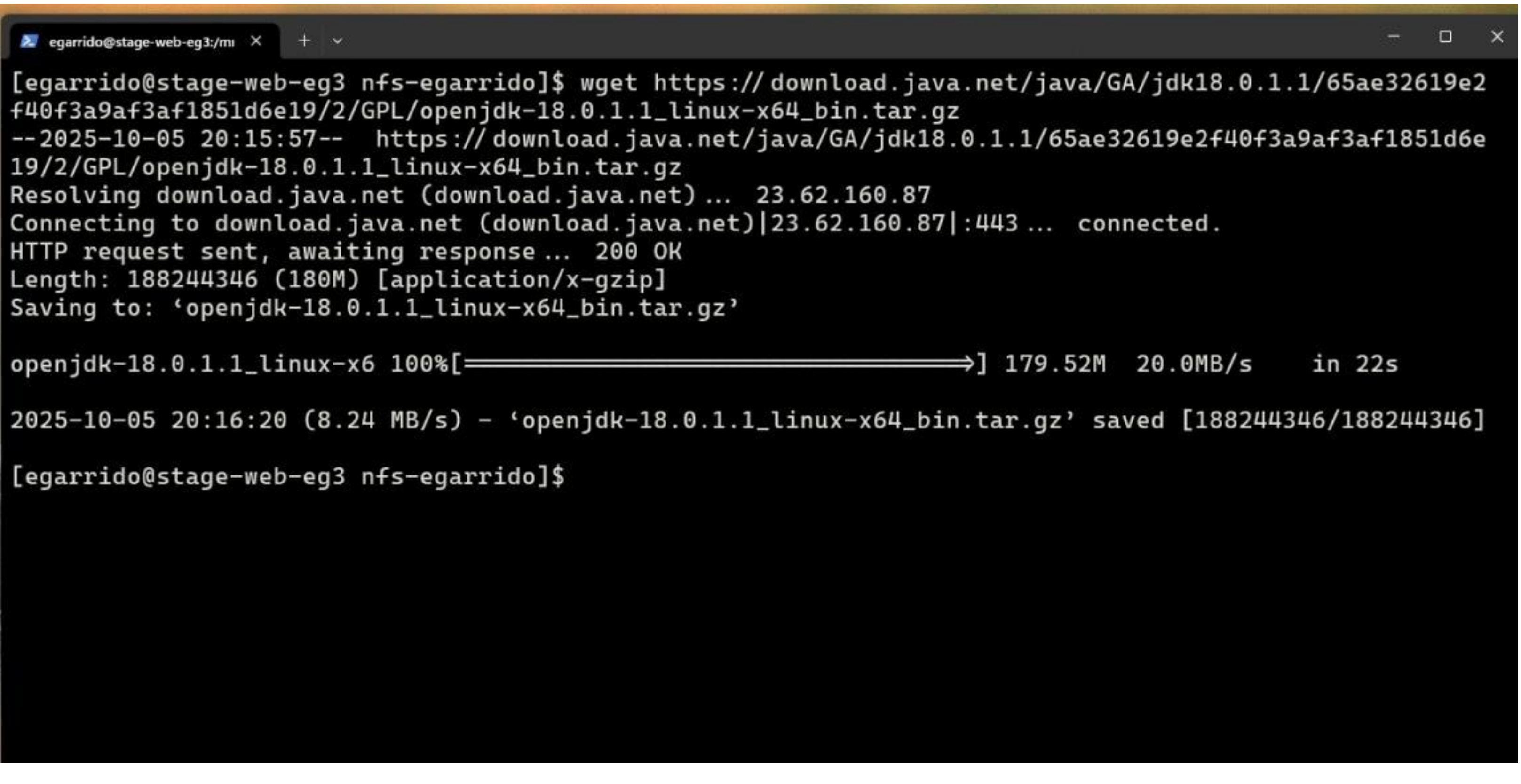
```
Package nfs-utils-1:2.5.4-38.el9.x86_64 is already installed.
Dependencies resolved.
=====================================================================================================
 Package                Architecture         Version                        Repository          Size
=====================================================================================================
Upgrading:
 nfs-utils              x86_64               1:2.5.4-39.el9                 baseos             460 k

Transaction Summary
=====================================================================================================
Upgrade  1 Package

Total download size: 460 k
Downloading Packages:
nfs-utils-2.5.4-39.el9.x86_64.rpm                                      1.3 MB/s | 460 kB     00:00
-----------------------------------------------------------------------------------------------------
Total                                                                  653 kB/s | 460 kB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                                             1/1
  Running scriptlet: nfs-utils-1:2.5.4-39.el9.x86_64                                             1/2
  Upgrading        : nfs-utils-1:2.5.4-39.el9.x86_64                                             1/2
  Running scriptlet: nfs-utils-1:2.5.4-39.el9.x86_64                                             1/2
  Running scriptlet: nfs-utils-1:2.5.4-38.el9.x86_64                                             2/2
  Cleanup          : nfs-utils-1:2.5.4-38.el9.x86_64                                             2/2
  Running scriptlet: nfs-utils-1:2.5.4-38.el9.x86_64                                             2/2
  Verifying        : nfs-utils-1:2.5.4-39.el9.x86_64                                             1/2
  Verifying        : nfs-utils-1:2.5.4-38.el9.x86_64                                             2/2

Upgraded:
  nfs-utils-1:2.5.4-39.el9.x86_64

Complete!
[egarrido@stage-web-eg3 ~]$
```

A mount point is created on the stage-web-eg3 host and the NFS export from the remote server is mounted successfully using NFSv4. Disk and filesystem output confirms the new share is mounted at the intended location alongside existing NFS mounts, validating network connectivity, permissions, and proper access to the shared directory.



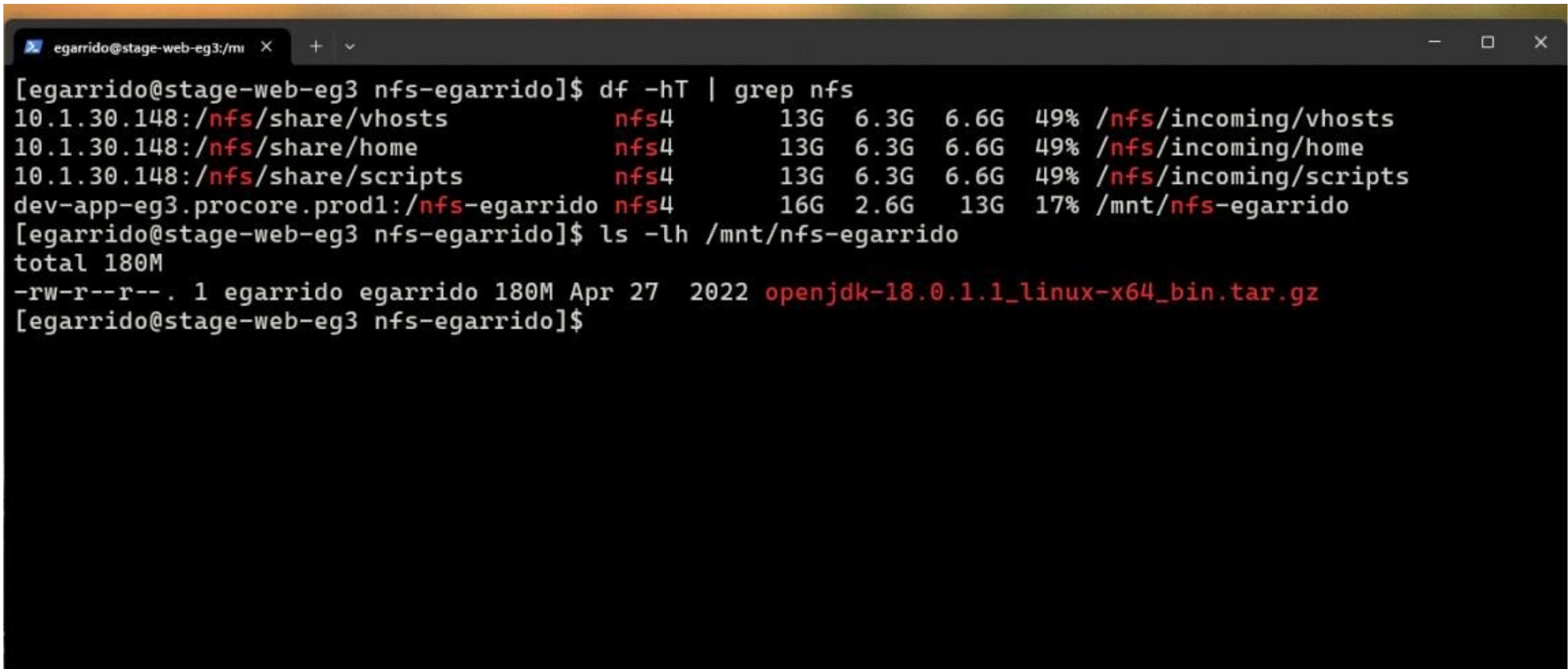
```
[egarrido@stage-web-eg3 ~]$ sudo mkdir -p /mnt/nfs-egarrido
[egarrido@stage-web-eg3 ~]$ sudo mount -t nfs dev-app-eg3.procore.prod1:/nfs-egarrido /mnt/nfs-egarri
do
[egarrido@stage-web-eg3 ~]$ df -hT | grep nfs
10.1.30.148:/nfs/share/vhosts           nfs4       13G  6.3G  6.6G  49% /nfs/incoming/vhosts
10.1.30.148:/nfs/share/home             nfs4       13G  6.3G  6.6G  49% /nfs/incoming/home
10.1.30.148:/nfs/share/scripts          nfs4       13G  6.3G  6.6G  49% /nfs/incoming/scripts
dev-app-eg3.procore.prod1:/nfs-egarrido nfs4       16G  2.4G   14G  16% /mnt/nfs-egarrido
[egarrido@stage-web-eg3 ~]$
```

The OpenJDK 18.0.1 binary archive is downloaded successfully using wget from the official source onto the mounted NFS directory. The transfer completes without errors, confirms a valid HTTP response, and saves the compressed archive locally, verifying network connectivity, write access to the shared location, and successful retrieval of the required Java runtime package.



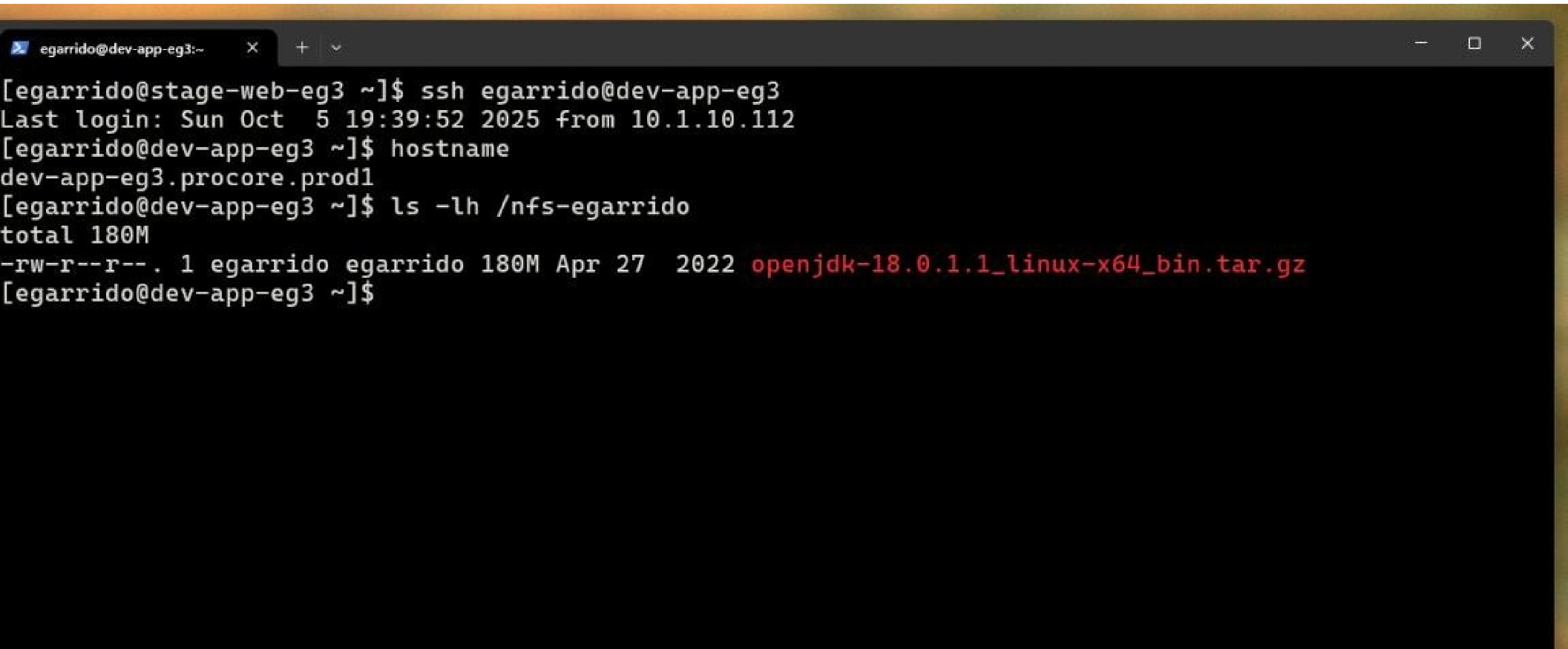
```
[egarrido@stage-web-eg3 nfs-egarrido]$ wget https:// download.java.net/java/GA/jdk18.0.1.1/65ae32619e2
f40f3a9af3af1851d6e19/2/GPL/openjdk-18.0.1.1_linux-x64_bin.tar.gz
--2025-10-05 20:15:57--  https:// download.java.net/java/GA/jdk18.0.1.1/65ae32619e2f40f3a9af3af1851d6e
19/2/GPL/openjdk-18.0.1.1_linux-x64_bin.tar.gz
Resolving download.java.net (download.java.net)...  23.62.160.87
Connecting to download.java.net (download.java.net)|23.62.160.87|:443...  connected.
HTTP request sent, awaiting response...  200 OK
Length: 188244346 (180M) [application/x-gzip]
Saving to: ‘openjdk-18.0.1.1_linux-x64_bin.tar.gz’

openjdk-18.0.1.1_linux-x6 100%[==================================>] 179.52M  20.0MB/s    in 22s

2025-10-05 20:16:20 (8.24 MB/s) - ‘openjdk-18.0.1.1_linux-x64_bin.tar.gz’ saved [188244346/188244346]

[egarrido@stage-web-eg3 nfs-egarrido]$
```

The mounted network file systems are verified on the host, confirming that all expected NFSv4 shares are active and accessible, including the newly mounted export. Disk usage output validates successful mounts and available capacity. A directory listing of the mounted share confirms the downloaded Java archive is present, readable, and stored correctly on shared storage, verifying both write access and data persistence on the NFS mount.



```
[egarrido@stage-web-eg3 nfs-egarrido]$ df -hT | grep nfs
10.1.30.148:/nfs/share/vhosts           nfs4       13G  6.3G  6.6G  49% /nfs/incoming/vhosts
10.1.30.148:/nfs/share/home             nfs4       13G  6.3G  6.6G  49% /nfs/incoming/home
10.1.30.148:/nfs/share/scripts          nfs4       13G  6.3G  6.6G  49% /nfs/incoming/scripts
dev-app-eg3.procore.prod1:/nfs-egarrido nfs4       16G  2.6G   13G  17% /mnt/nfs-egarrido
[egarrido@stage-web-eg3 nfs-egarrido]$ ls -lh /mnt/nfs-egarrido
total 180M
-rw-r--r--. 1 egarrido egarrido 180M Apr 27  2022 openjdk-18.0.1.1_linux-x64_bin.tar.gz
[egarrido@stage-web-eg3 nfs-egarrido]$
```

A secure connection is established from the staging host to the development server, and the hostname is verified to confirm the target system. The shared NFS directory is then accessed on the development server, where a directory listing confirms that the Java archive downloaded earlier is present and visible. This validates that the NFS export is functioning correctly across hosts and that data written from one system is immediately accessible from another.



```
[egarrido@stage-web-eg3 ~]$ ssh egarrido@dev-app-eg3
Last login: Sun Oct  5 19:39:52 2025 from 10.1.10.112
[egarrido@dev-app-eg3 ~]$ hostname
dev-app-eg3.procore.prod1
[egarrido@dev-app-eg3 ~]$ ls -lh /nfs-egarrido
total 180M
-rw-r--r--. 1 egarrido egarrido 180M Apr 27  2022 openjdk-18.0.1.1_linux-x64_bin.tar.gz
[egarrido@dev-app-eg3 ~]$
```

Summary

Shared NFS storage was configured and validated across development and staging hosts to provide reliable cross-system file access. NFS utilities were installed and updated, export directories were created with appropriate permissions, and export definitions were applied to allow controlled read-write access. Required services were enabled, firewall rules were opened, and mounts were verified on client systems. Files written from one host were successfully accessed from another, confirming proper export configuration, network connectivity, and data consistency across environments.