# WAREWULF v4

* Warewulf is a **open-source** distributed computing management software used for the deployment and administration of cluster.Warewulf is an operating system provisioning platform for Linux that is designed to produce secure, salable, turnkey cluster development that maintain flexibility and simplicity. As of warewulf the virtual node image is a standard container Image.

**Features:**

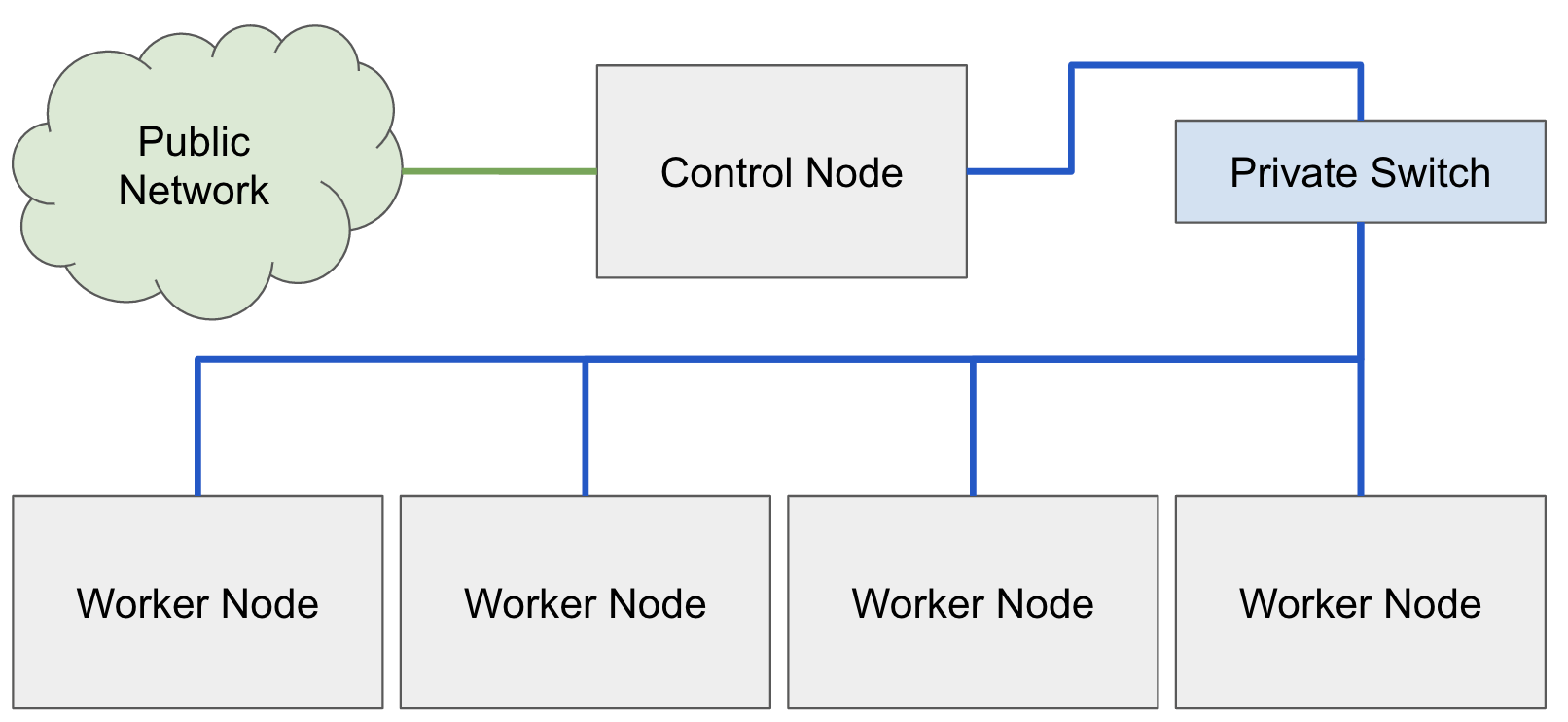
1. Lightweight: Stateless operating system image.
2. Simple:
3. Flexible: Cluster scale.
4. Agnostic: Work under any environment.
5. Secure: Warewulf support SElinux.
6. Open-source:

(Easy to deploy, Easy to manage, Easy to use, Scalability and Flexibility, Container support

, Expandable, Monitoring and Diagnostics, Optimized)

**Architecture:**

The achitecture is defined as a group of similar compute nodes all connected together by using standard commodity equipment on a private network segment. The control node has two network interface, the one with public network connected to the master node and the second one is in privete network interface to perform provisioning of compute node.



**Provisioning:**

Defination is the process of putting an operating system onto a system. In cluster environment this mean one could group all the nodes together to be installed in bulk.

That’s why we need to use provisioning tools like warewulf.

**Note:** Warewulf will need to configure to use the private cluster management network. Warewulf will use this network for booting the nodes over PXE.

There are three network protocols use to accomplish this DHCP/BOOT, TFTP and HTTP on port 9873.

**Warewulf.conf: $ vim /etc/warewulf/warewulf.conf**

WW\_INTERNAL: 43 (version 4.3.0)

ipaddr: 192.168.200.1 (masternode ip)

netmask: 255.255.255.0

network: 192.168.200.0 (subnet)

warewulf:

port: 9873

secure: false (If true then compute node can have privilage of 987 port)

update interval: 60 (time in second update fatch from master)

autobuild overlays: true

host overlay: true

syslog: false (whether server log go to syslog or directly store in /var/log/warewulf.log)

dhcp:

enabled: true

range start: 192.168.200.50 (start ip range for compute node)

range end: 192.168.200.99 (end ip range for compute node)

systemd name: dhcpd

tftp:

enabled: true

systemd name: tftp

nfs:

enabled: true

export paths:

- path: /home

export options: rw,sync

mount options: defaults

mount: true

- path: /opt

export options: ro,sync,no\_root\_squash

mount options: defaults

mount: false

systemd name: nfs-server

**Note:** The /etc/warewulf/ipxe directory which contains templete of warewulf.conf file.

**Nodes.conf:**  $ /etc/warewulf/nodes.conf

It is a flat YAML configuration file, it’s a primary database file for all compute nodes.where we can configure compute nodes details include interface, mac, ip, hostname etc.

**Container Manage:**

Warewulf version 4 has used the model of the VNFS as a template image for the compute nodes.This is similar to a Golden master image that exists within a directory call CHROOT.

Now the warewulf 4 directly integrates within the container ecosystem to facilitate the process of VNFS image management.

In warewulf there is **wwint overlay** to configure services during boot services include network manager, SElinux etc.

**When the cluster boots, the following oder of operations will occur:**

1. **BIOS:** \* The system BIOS will bootstrap the initiliazation of the hardware.

\* The the network card will regestred its option ROM into the BIOS.

\* The boot device are attempt in order.

\* When it goes into Network Boot device (PXE RUN)

2.  **PXE:** \* PXE requirest a DHCP address on the network.

\* Warewulf controller DHCP server will respond with network conf and Filename to try and boot.

\* PXE will attempt to download the filename referred to DHCP responce via TFTP.

\* The downloaded file will execute So, they reach out to the warewulf server for its configuration.

3.  **BOOTSTRAP**: \* The warewulf server will generate the ipxe conf .

\* The kernel, Container image and system overlay all downloaded over HTTP from warewulf server.

\* iPXE execute kernel and system overlay to provide root file system.

\* The warewulf bootstrap execs the container /sbin/init

1. **CONTAINER:** \* The container now boots exactlly as any operating system would expect.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~