APRIL 25, OSTRAVA

Warewulf making cluster installations fast and reliable

Christian Goll <cgoll@suse.com>

SUSE

Introduction

Warewulf is tool for managing beowulf clusters

Beowulf

- old british poem

Beowulf cluster

- became popular in the 90.
- use of the shelf hardware
  - 486 & linux
  - **not** Cray & unix
- warewulf is a typo of werewolf

Introduction

HPC landscape

Top five Supercomputers

1	Frontier	EPYC 64C	AMD MI250X	Slingshot-11
2	Aurora	Xeon 9470	Intel GPU Max	Slingshot-11
3	Eagle	Xeon 8480	NVIDIA H100	NVIDIA Infiniband
4	Fugaku	A64FX 48C 2.2GHz	-	Tofu interconnect D
5	LUMI	EPYC 64C 2GHz	AMD MI250X	Slingshot-11

- only Fugaku uses non standard CPU
- others are beowulf clusters with GPUs attached

Introduction

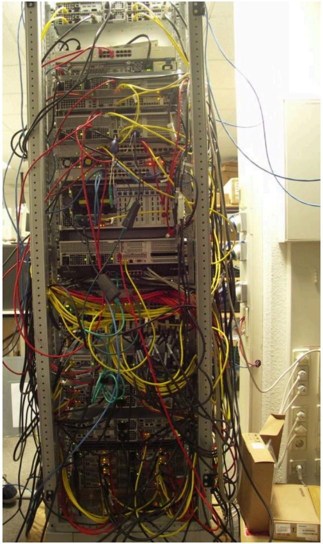
Beowulf cluster

base components

- management node
- compute nodes
- management network

optional components

- more compute nodes
- fast network interconnects
- central storage
- bmc/ipmi

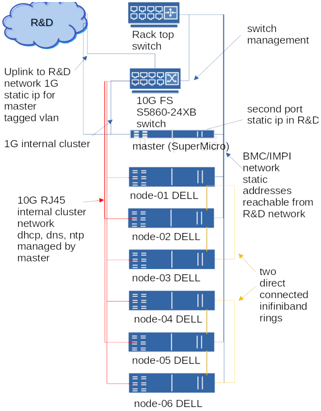


Introduction

Beowulf Cluster

differences to data centers

- compute nodes are cattle
- hierarchical organization
- compute are not updated after boot process
- application come from central storage
- applications are self compiled
- one application can run over several nodes



R&D
Rack top switch
switch management
Uplink to R&D network 1G static ip for master tagged vlan
10G FS S5860-24XB switch
second port static ip in R&D
1G internal cluster
master (SuperMicro)
BMC/IMPI network static addresses reachable from R&D network
10G RJ45 internal cluster network dhcp, dns, ntp managed by master
node-01 DELL
node-02 DELL
node-03 DELL
node-04 DELL
node-05 DELL
node-06 DELL
two direct connected inifiniband rings

Warewulf description

software stack

warewulf components

warewulfd delivers

- kernel & modules
- node image
- node configurations

`wwctl` cmd line tool

- manages node database
- manages node image

external components

dhcp server

- ISC dhcpd server
- dnsmasq

tftp

- kernel tftp
- dnsmasq

optional

- nfs
- manage `/etc/hosts`

Warewulf configuration

database /etc/warewulf/nodes.conf

- plain yaml file
- easy backup
- can be version controlled
- external tools support
  - vim, ansible

profiles

- stores identical values for collection of nodes
- values can be overiden on node basis

```yaml
WW_INTERNAL: 45
nodeprofiles:
  default:
    comment: This profile is automaticall
    container name: leap
    network devices:
      default:
        device: eth0
nodes:
  n01:
    profiles:
    - default
    network devices:
      default:
        hwaddr: 52:54:00:4e:cb:1d
        ipaddr: 172.16.130.101
  n02:
```

Warewulf configuration

command line database manipulation

add node

```
wwctl node add
n01 -I
10.10.10.1
```

modify node

```
wwctl node set
n01 --comment
"Have fun"
```

list node

```
wwctl node list
n01 -a"
```

```
NODE FIELD                    PROFILE    VALUE
n01  Id                       --         n01
n01  Comment                  SUPERSEDED Have fun
n01  ContainerName            default    leap
n01  Ipxe                     --         (default)
n01  RuntimeOverlay           --         (generic)
n01  SystemOverlay            --         (wwinit)
n01  Root                     --         (initramfs)
n01  Discoverable             --         false
n01  Init                     --         (/sbin/init)
n01  Kernel.Args              --         (quiet crashkernel
n01  Profiles                 --         default
n01  PrimaryNetDev            --         (default)
n01  NetDevs[default].Type    --         (ethernet)
n01  NetDevs[default].OnBoot  --         (true)
n01  NetDevs[default].Device  default    eth0
n01  NetDevs[default].Hwaddr  --         52:54:00:4e:8cb:1d
n01  NetDevs[default].Ipaddr  --         172.16.130.101
```

Warewulf configuration

templates& overlays

Configuration templates

- based on go templates
- {{*.foo*}} replaced with variable `foo`
- exported go function can be called

Configuration overlays

- rendered templates packed into overlay
- overlay put ontop of node image

Listing 1: issue.ww

```
Warewulf Node:      {{.Id}}
Container:          {{.Container}}
{{ if .Kernel.Version }}Kernel:
Kernelargs:         {{.Kernel.Args}}

Network:
{{- range $devname, $netdev := .NetDevs}}
    {{$devname}}: {{$netdev.Device}}
    {{$devname}}: {{$netdev.IpCIDR}}
{{if $netdev.Ipaddr6 }}    {{$devname}}:
{{if $netdev.Hwaddr }}    {{$devname}}:
{{end}}
```

Warewulf configuration

overlays

warewulf defines two types of overlays

system overlay

- available on boot
- warewulf boot strap files
- static network configurations:
  - wicked
  - NetworkManager
  - EL scripts
- nfs mounts
- file system mounts

runtime overlay

- updated on regular base
- can be secured

user defined overlays

- users are encouraged to create own configuration templates
- can reside in system & runtime overlays

Warewulf configuration

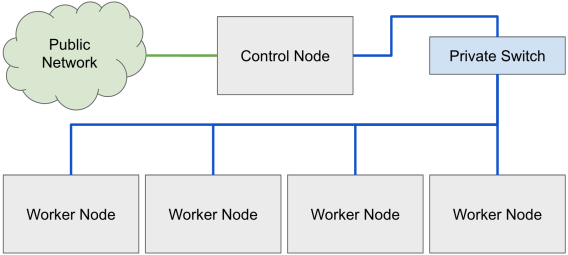
security

assumptions

- private/cluster network is secure
- lateral movement isnt't accounted
  - NFS mounts are common, when not **mandatory**

measurements

- node image & system overlay protected with BIOS UUID
- system overlays **must** be downloaded from privileged port



Warewulf configuration

node images

definition

- complete OS images
- called containers in warewulf
- must be imported from:
  - chroot directory
  - docker registry
  - local `dockerd`
- several different node images can be imported
- node images are vendor independent

`registry.suse.com`

- SUSE SLE 15SP5

`registry.opensuse.org`

- openSUSE Tumleweed
- openSUSE Leap 15SP[3-5]

`ghcr.io`

- openSUSE Leap
- Rocky EL (8&9)
- Debian Bockworm