hendrik.nolte@gwdg.de

Hendrik Nolte

Node Provisioning Using Warewulf

Part I - Basics

Table of contents

Discussion of First Tutorial Sheet

Time For Your Feedback!

Introduction: What are Overlays?

- During your hands-on session you have configured 2 nodes to boot into the same stateless image.
- Although this homogeneous state is a general advantage, sometimes there have to be node specific configurations, like
  - just think of `hostname`
  - or your networks like, Infiniband, OPA, IPMI
  - or sometimes for your parallel filesystem
- For this, Warewulf offers 2 different overlays:
  - A `System Overlay` which is called after the bootstrapping of the container is done
  - A `Runtime Overlay` which is called periodically during operation
- Both of these script types can be templated
- If you have a file, where you used templates you need to put a `.ww` suffix behind the filename
  - Only then Warewulf will parse the variables and create individual static files for each node

Introduction: What are Overlays?

- Overlays are individual `cpio` archives for every node
- After the iPXE booted the kernel, the kernel calls `init`, a script provided by the system overlay
- Only afterwards the container boots

Applying Overlays

- Overlays are stored in `/var/lib/warewulf/overlays`
- The Systemoverlay is called `wwinit`
- The Runtime Overlay is called `runtime`
- You can set the Overlays as you set any other attribute

```
NODE                FIELD               PROFILE      VALUE
================================================================================
n1                  Id                  --           n1
n1                  comment             default      This profile is automatically included for each
node
n1                  cluster             --           --
n1                  container           --           --
n1                  ipxe                --           (default)
n1                  runtime             --           (generic)
n1                  wwinit              --           (wwinit)
n1                  root                --           (initramfs)
n1                  discoverable        --           --
n1                  init                --           (/sbin/init)
n1                  asset               --           --
n1                  kerneloverride      --           --
n1                  kernelargs          --           (quiet crashkernel=no vga=791 net.naming-scheme=
v238)
n1                  ipmiaddr            --           --
n1                  ipminetmask         --           --
n1                  ipmiport            --           --
n1                  ipmigateway         --           --
n1                  ipmiuser            --           --
n1                  ipmipass            --           --
n1                  ipmiinterface       --           --
n1                  ipmiwrite           --           --
n1                  profile             --           default
n1                  default:type        --           (ethernet)
n1                  default:onboot      --           --
n1                  default:netdev      --           (eth0)
n1                  default:hwaddr      --           --
n1                  default:ipaddr      --           --
n1                  default:ipaddr6     --           --
n1                  default:netmask     --           (255.255.255.0)
n1                  default:gateway     --           --
n1                  default:mtu         --           --
n1                  default:primary     --           true
```

Figure: Screenshot showing the node attributes which one hascan *to* set

Templating

- Warewulf uses a simple `text/template` engine to convert dynamic, node specific content into static content
- Those files need to have a `.ww` suffix
- Those templates are used for overlays
  - Since templates as well as overlays are used for node specific configurations

Templating - Example

- Maybe you have wondered, why you could do a `ssh n0` during your last exercise?
- The answer is that an corresponding entry was done in `/etc/hosts`
- This file is used to resolve hostnames to ip addresses
- Lets have a look at `/var/lib/warewulf/overlays/generic/etc/hosts.ww`

```
# Warewulf Server
{{$.Ipaddr}} warewulf {{$.BuildHost}}

{{- range $node := $.AllNodes}}                    {{/* for each node */}}
# Entry for {{$node.Id.Get}}
{{- range $devname, $netdev := $node.NetDevs}} {{/* for each network device on the node */}}
{{- if $netdev.Ipaddr.Defined}}                  {{/* if we have an ip address on this network device */}}
{{- /* emit the node name as hostname if this is the primary */}}
{{$netdev.Ipaddr.Get}} {{$node.Id.Get}}-{{$devname}}
{{- if $netdev.Device.Defined}} {{$node.Id.Get}}-{{$netdev.Device.Get}}{{end}}
{{- if $netdev.Primary.GetB}} {{$node.Id.Get}}{{end}}
{{- end}} {{/* end if ip */}}
{{- end}} {{/* end for each network device */}}
{{- end}} {{/* end for each node */}}
```

Live Demo

Any Questions?
Live Demo
At the end, I would like to show you what other cool stuff you can do with overlays ;)