Pathfinding into the clouds

Dr. Ole Holm Nielsen
Technical University of Denmark

See the Wiki page:
https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting
**Pathfinding**: Plotting by a computer application the shortest route between two points (Wikipedia)
Scaling an on-premise cluster into the clouds

Use cases:
- Extend cluster on-demand with compute nodes in the cloud.
- Access to CPU types other than what’s in your on-premise cluster.
- Access to GPUs and other accelerators in stead of buying them.

Outline of talk:
1. This work is based on the Azure cloud service.
2. Open an account at the cloud service provider.
3. Create one or more cloud Virtual Machines (VM) and a Virtual Network (Vnet).
4. Create a VPN IPsec tunnel between your on-premise subnet and the cloud Vnet.
5. Write a script to power up/down VMs.
6. Define storage space in the cloud for application software and user data.
Open an account at a cloud service provider

• The present work is based on the **Azure cloud service** with guidance from Azure experts in Denmark.

• Many universities and other organizations already have a central IT **Azure Subscription**. We asked our central IT to create an Azure account for us.

• Details are at [https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting#resources-for-slurm-on-microsoft-azure](https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting#resources-for-slurm-on-microsoft-azure)
Create cloud Virtual Network and Machines

- Create an Azure Virtual Network (Vnet)
- Create a VPN Gateway
  [https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting#vpn-gateway-to-azure](https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting#vpn-gateway-to-azure)
- Example topology for a Vnet named AzureHPC:
Create a Virtual Machine

- Create a VM based upon some pre-existing minimal VM image: https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting#create-resources-and-machines-in-azure-home

- **AlmaLinux** images are freely available in **Azure**.

- **RockyLinux** images are only available for pay in **Azure**.

- You must save the VM's *SSH public key file* and use it later to login via the *VPN tunnel*.

- Remember to shut down VMs when they are not in use!
On-premise IPsec VPN Gateway

• This is the least documented part of the cloud bursting adventure!

• Create a **Site-to-Site IPSec VPN connection** in Azure: [https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting#configure-vpn-gateways](https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting#configure-vpn-gateways)

• Azure provides a list of supported *compatible hardware router devices* (Cisco, Juniper, etc.) with links to vendors’ configuration guides.

• **Problem**: What to do if you don’t have the money to buy an expensive router, and/or the time to become an expert in the router’s OS configuration???
Solution: Build your own IPsec tunnel
Beware of strange problems lurking in tunnels

Sulking reindeer!

Copyright Ole Holm Nielsen
Libreswan comes to the rescue

- **Libreswan** is a free software implementation of the most widely supported and standardized VPN protocol using *IPsec* and the *Internet Key Exchange* (IKE).

- Use any Linux server with 2 NICs as the on-premise VPN gateway to the cloud service. One NIC faces the public Internet, the other NIC is in your cluster subnet.

- *RHEL 8* (and clones such as *RockyLinux* and *AlmaLinux*) comes with *Libreswan* v4.4.

- There are numerous web-pages with instructions for setting up a site-to-site VPN tunnel. We have tried dozens of these methods, but none have worked on an EL8 server 😞

- Azure does not *support* Libreswan for VPN tunnels 😞 But it works nevertheless!
Libreswan setup that actually works

• *Libreswan* setup details (firewall, IPsec, routing, etc.) are in: [https://wiki.fysik.dtu.dk/it/Libreswan_IPsec_VPN](https://wiki.fysik.dtu.dk/it/Libreswan_IPsec_VPN)

• Example IPsec configuration file `/etc/ipsec.d/azure.conf`:

```plaintext
conn azure          # Connection name
left=123.45.67.89   # Local VPN gateway public address
leftsubnet=10.2.0.0/16 # Local subnet
leftsourceip=10.2.0.1 # Local VPN gateway on the local private subnet
right=20.21.22.23   # Azure VPN gateway public address
rightsubnet=10.0.0.0/16 # Azure subnet
authby=secret      # Use shared secret with Azure
auto=start          # Start Ipsec at reboot
dpdaction=restart  # Restart if peer has died
dpddelay=30        # Dead peer delay
dpdttimeout=120    # Dead peer timeout
ike=aes256-shal;modp1024 # IKE encryption/authentication algorithm
ikelifetime=3600s   # IKE renegotiation
pfs=yes             # Perfect Forward Secrecy
esp=aes128-shal     # Child SA negotiation algorithms
salifetime=3600s     # Expiry of a connection
```
Integrating cloud VM nodes with your on-premise cluster

• At this point your on-premise cluster subnet servers can communicate directly with TCP/IP via the *Libreswan* *IPsec* router (or a hardware router).

• Now you can SSH to a VM to configure the OS and install any necessary applications, as you would do with any on-premise node.

• We use Ansible for node configuration: [https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting#ansible-with-azure](https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting#ansible-with-azure)

• Authentication: Configure users in the VMs as you normally do in your cluster (for example, add users to `/etc/passwd`).

• Your custom VM can now be cloned to create many identically configured VMs: [https://docs.microsoft.com/en-us/azure/virtual-machines/capture-image-portal](https://docs.microsoft.com/en-us/azure/virtual-machines/capture-image-portal)
Configure Slurm to power up and down VMs

- Slurm Power Saving guide https://slurm.schedmd.com/power_save.html

- Configure Slurm powering scripts:
  https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting#slurm-configuration-for-cloud-nodes

- Slurm powering scripts for Azure:
  https://github.com/OleHolmNielsen/Slurm_tools/tree/master/cloud
  The scripts can also be used without Slurm.
Azure shared NFSv3 Storage Blobs

- In February 2022 Azure started offering NFSv3 Storage Blobs:

- This support provides Linux file system compatibility at object storage scale and prices and enables Linux clients to mount a container in Blob storage from an Azure Virtual Machine (VM) or a computer on-premises.

- Setting up an Azure Storage Account:
  https://wiki.fysik.dtu.dk/niflheim/Slurm_cloud_bursting#azure-storage-accounts

- Now you can NFS-mount the Azure Blob storage container’s IP-address/DNS-name. This works inside your VMs as well as in your on-premise servers (via the VPN tunnel)! We use the NFS auto-mounter with Azure storage.

- You may create user home directories in the Azure storage.
Test Slurm with cloud nodes

• Read the Slurm Cloud Scheduling Guide: https://slurm.schedmd.com/elastic_computing.html

• Add cloud nodes to Slurm as described in a previous slide.

• Submit batch jobs to run on the cloud nodes. Cloud nodes will be powered up and down on-demand.