

# Building and management of distributed computing cluster Vojtěch Bůbela

Faculty of Information Technology, Brno University of Technology, CZ

### Overview

Main goal of this thesis is focused on building distributed computation cluster. One control node and multiple computing nodes are connected together and every node is running Slurm task scheduler<sup>1</sup>software. Secondary goal of this thesis is to manage resources on the cluster. That includes correctly assigning resources to user tasks and preventing them from obtaining more resources than they were given. There is also a need for accounting events on the cluster and managing user privileges on computing nodes.

### Architecture used in first experiment

Slurm task scheduler software was used to build multiple clusters using different computers as nodes. The architecture, shown on figure 1, was kept at the basic level during first experiments. Focus of this experiment was to get familiar with concepts of distributed computing and with Slurm task scheduler.



### Second architecture used with raspberry pi 3B computers

BRNO FACULTY

UNIVERSITY OF INFORMATION

OF TECHNOLOGY TECHNOLOGY

As a part of second experiment the more complex architecture was applied to three raspberry pi 3B computers. Figure 3 shows raspberry pi computers clustered together via network switch and Ethernet cabels. User input was provided by connecting personal computer to the same network switch. Installation of Slurm software and uploading/updating configuration files was managed by Ansible automation.



Figure 1: basic architecture of distributed computing cluster

First experiment was done by creating multiple virtual machines of Ubuntu server 20.04. These virtual machines were connected with virtual network.

# More complex architecture used in second experiment

After successfully building distributed computation cluster from virtual machines using Slurm, more complex architecture shown on figure 2 was designed. In the new architecture additional services and features are available such as accounting database using MySQL or Slurm plugin that uses cgroups<sup>2</sup> for monitoring resources.



Figure 3: Raspberry pi 3B clustered with network switch

Every rpi runs ubuntu server 20.04 edition made for rpi. It should be noted that this linux distribution proved difficult to work with when configuring different services.

# Results

Cluster of three raspberry pi computers was configured to have three task queues, accounting database and advanced resource management. Slurm allows for easy plugin deployment. For resource management two plugins were used: consres and cgroups. Consres provides a way to split resources of compute node between multiple tasks from possibly multiple users. Cgroups plugin monitors tasks running on the compute nodes. Figure 4. shows example of user submitting task to the cluster.

<b>vbubela@node1:</b> ~\$ sinfo		
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST		
debug up infinite 1 idle node3		
batch* up infinite 1 idle node2		
all_nodes up infinite 2 idle node[2-3]		
vbubela@node1:~\$ sbatch basic_script		
Submitted batch job 298		
vbubela@node1:~\$ squeue		
JOBID PARTITION NAME USER ST	TIME	NODES NODELIST(REASON)
298 debug <testjob r<="" td="" vbubela=""><td>0:01</td><td>1 node3</td></testjob>	0:01	1 node3
<b>vbubela@node1:</b> ~\$ sinfo		
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST		
debug up infinite 1 mix node3		
batch* up infinite 1 idle node2		
all_nodes up infinite 1 mix node3		
all_nodes up _infinite 1 idle node2		
vbubela@node1:~\$		

#### Figure 4: user submitting task to cluster

Command sinfo displays task queues and state of nodes in them. Command sbatch submits following parameter as a task to be computed. The task can be then seen as executing on node3.

Figure 2: More complex architecture used in second experiment

Slurm task scheduler uses MUNGE service<sup>3</sup>for authentication of communication between nodes. User can directly connect via ssh to compute node but only if they have active task on the node.

# Conclusion

With guidance of my supervisor i built and configured functioning distributed computation cluster. I also configured several services normally found in real-life computation clusters such as database logging or resource management and tested the cluster by running several tasks. Output of this thesis are configuration files for the cluster and manual describing how to apply these configurations step by step. This can be used as a base for building real-life distributed computation cluster, maybe even on VUT FIT.

<sup>1</sup>Slurm task scheduler is opensource and has very descriptive documentation <sup>2</sup>MUNGE project <sup>3</sup>Cgroups plugin used by slurm