

Państwowa Wyższa Szkoła Zawodowa w Nowym Sączu Programowanie współbieżne i rozproszone	
Projekt - PWIR08 - Skonfiguruj klaster MPI na wirtualnych maszynach (1 master 3 node).	
Nazwiska i imiona: Jakubczak Michał Zwoliński Szymon	
Data oddania: 14.06.2022	Grupa: P2

1.Instalacja i konfiguracja wirtualnych maszyn na systemie Ubuntu dla klaster_master, klaster_node1, klaster_node2 i klaster_node3.

Dodanie maszyn wirtualnych

Utwórz wirtualną maszynę

Nazwa i system operacyjny

Please choose a descriptive name and destination folder for the new virtual machine and select the type of operating system you intend to install on it. The name you choose will be used throughout VirtualBox to identify this machine.

Name:

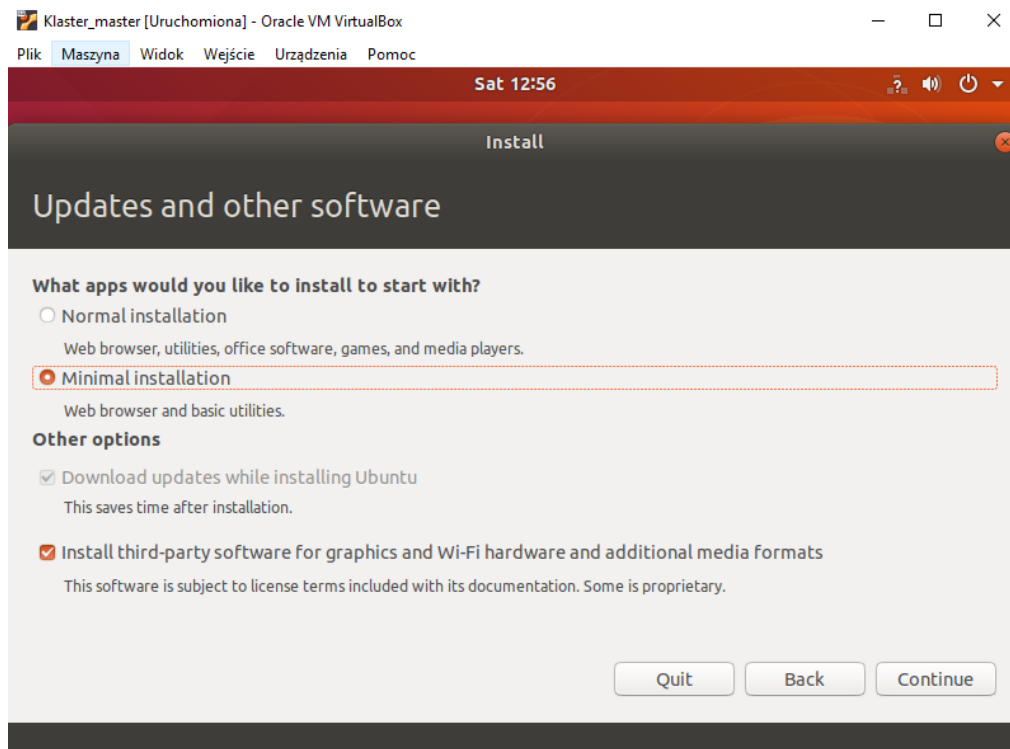
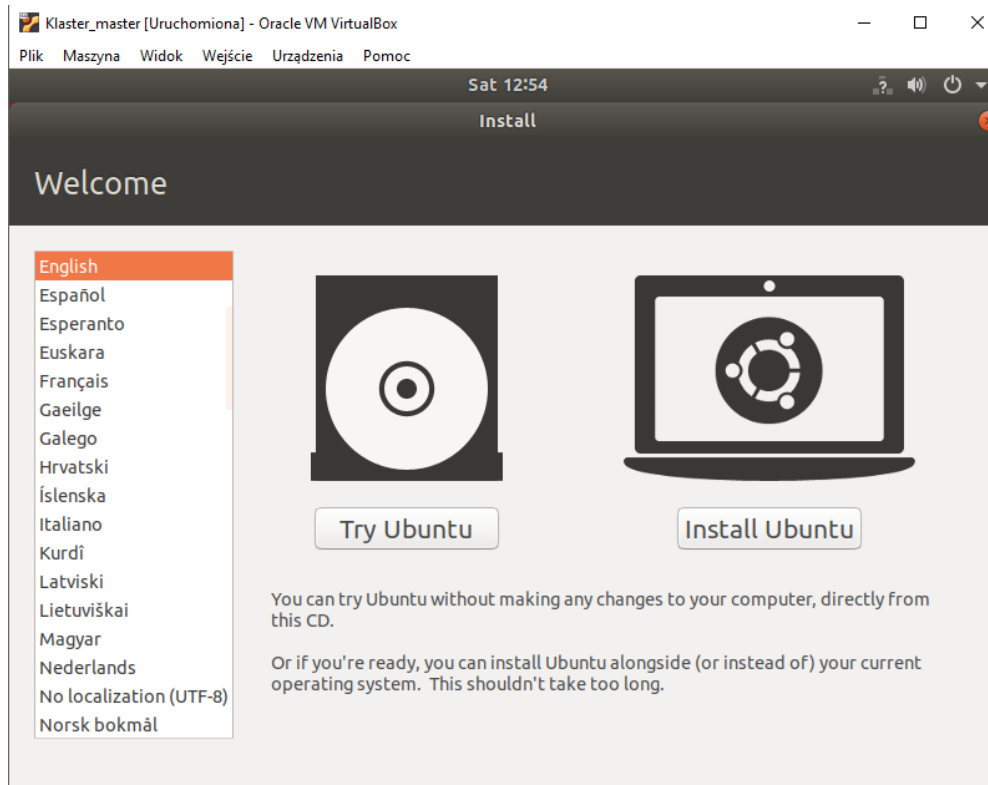
Machine Folder:

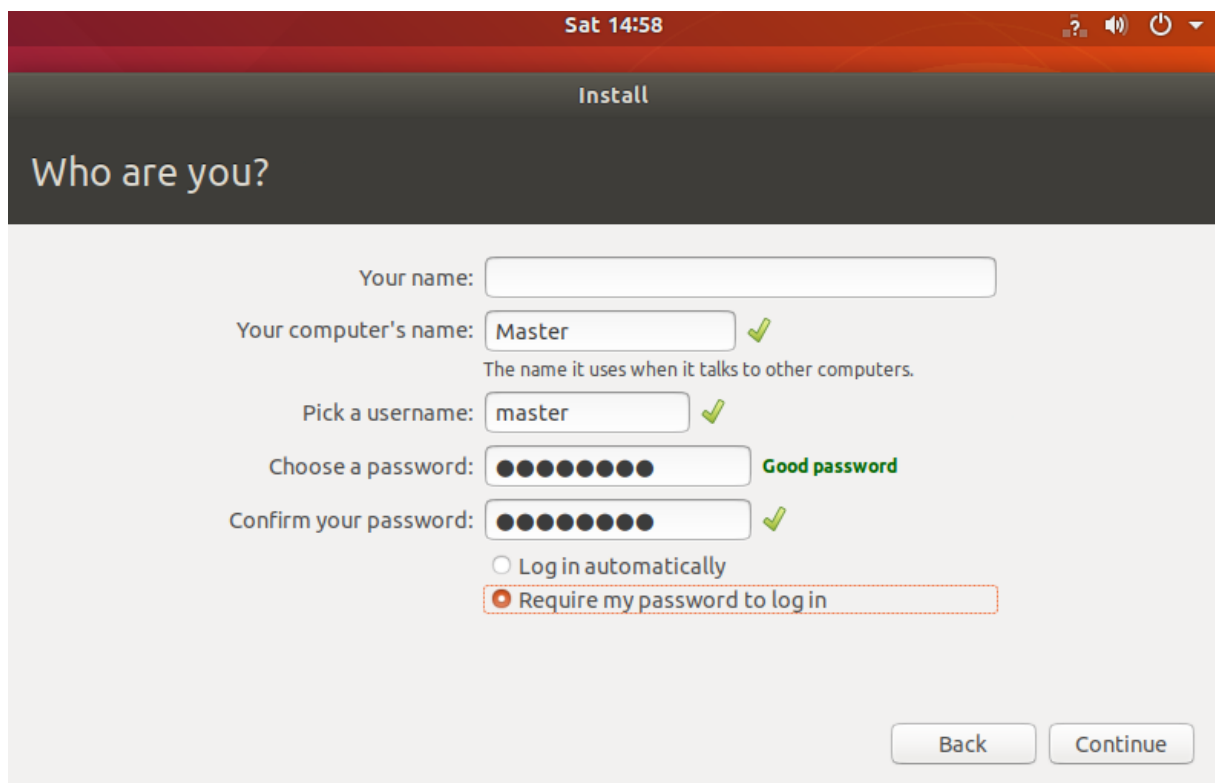
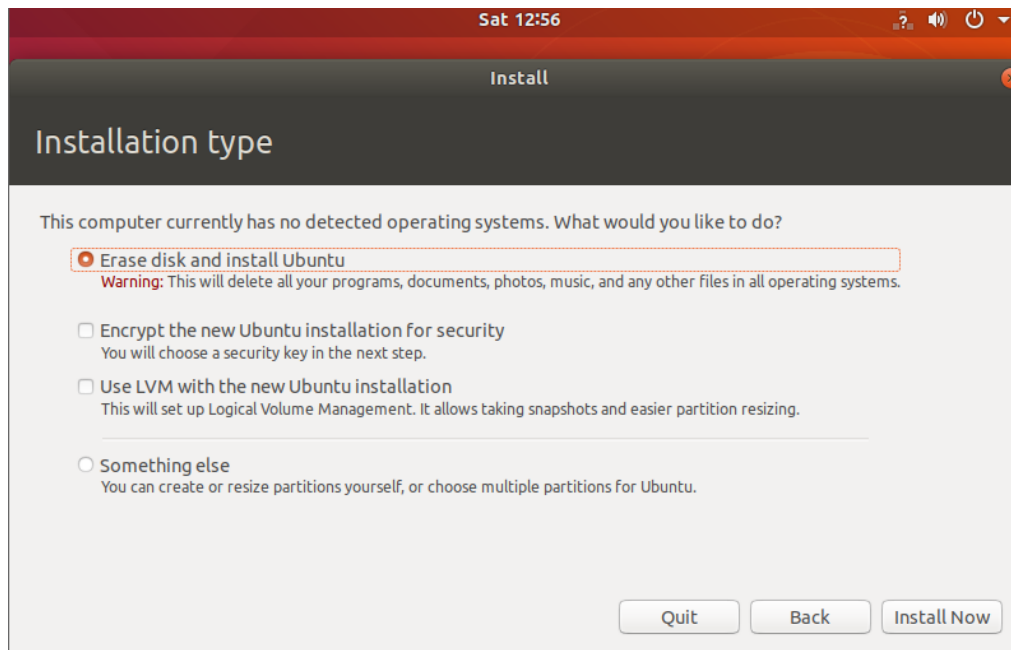
Typ:

Wersja:

Tryb eksperta **Dalej** Anuluj

Instalacja Ubuntu





Na końcu usuwamy obraz systemu i można przystąpić do konfiguracji klastrów (wszystkie czynności wykonujemy analogicznie dla każdej z 4 maszyn).

2. Tworzenie klastrów

Definiowanie nazwy hostów na poszczególnych systemach kolejno na master, node1, node2 i node3:

```
master@Master: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/hosts Modified
127.0.0.1 localhost
192.168.133.100 Master
192.168.133.101 node1
192.168.133.102 node2
192.168.133.103
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

```
node1@node1: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/hosts
127.0.0.1 localhost
192.168.133.100 Master
192.168.133.101 node1
```

```
node2@node2: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/hosts
127.0.0.1 localhost
192.168.133.100 Master
192.168.133.102 node2
```

```
node3@node3: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/hosts
127.0.0.1 localhost
192.168.133.100 Master
192.168.133.103 node3
```

Instalacja NFS

Robimy z mastera serwer NFS:

```
master@Master:~$ sudo apt-get install nfs-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Na pozostałych nodach wykonujemy poniższe polecenie w konsoli:

```
node1@node1:~$ sudo apt-get install nfs-client
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Udostępnienie folderu mastera dla pozostałych węzłów w sieci

Poprzez edytowanie pliku /etc/exports i zrestartowaniu usługi nfs-server:

```
master@Master:~$ sudo mkdir /mirror
master@Master:~$ echo "/mirror *(rw,sync)" | sudo tee -a /etc/exports
/mirror *(rw,sync)
master@Master:~$ sudo service nfs-kernel-server restart
master@Master:~$
```

Następnie montujemy udostępniony folder na pozostałych klastrach:

```
node1@node1: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/fstab

# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to
# specify devices that works even if disks are added and removed. See
#
# <file system> <mount point> <type> <options>
# / was on /dev/sda1 during installation
UUID=68b1b9cf-2eaa-4f17-9daf-426a3e58a949 /
/swapfile none
Master:/mirror /mirror nfs
```

```
node1@node1:~$ sudo mkdir /mirror
node1@node1:~$ sudo mount -a
node1@node1:~$
```

Kolejnym krokiem jest zdefiniowanie użytkownika do uruchamiania programów MPI (poniższe czynności wykonujemy kolejno na każdym węźle):

```
master@Master:~$ sudo adduser mpiu
Adding user `mpiu' ...
Adding new group `mpiu' (1001) ...
Adding new user `mpiu' (1001) with group `mpiu' ...
Creating home directory `/home/mpiu' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for mpiu
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] y
master@Master:~$ sudo chown mpiu /mirror
master@Master:~$
```

Instalacja SSH we wszystkich węzłach:

```
master@Master:~$ sudo apt-get install openssh-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Konfiguracja SSH do komunikacji pomiędzy węzłami:

Zalogowanie na nowego użytkownika do węzła głównego

```
master@Master:~$ su - mpiu
Password:
mpiu@Master:~$
```

Wygenerowanie klucza RSA:

```

mpiu@Master:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/mpiu/.ssh/id_rsa):
Created directory '/home/mpiu/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/mpiu/.ssh/id_rsa.
Your public key has been saved in /home/mpiu/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:2L2qJLpKDQPQiaFk75QQQlCQYyIRHv0YL3ctPXz2ST0 mpiu@Master
The key's randomart image is:
+---[RSA 2048]---+
|@^+. . . + . |
|&.*o.o . o = o E. |
|*o += . . . + o o |
|. o. . o . . o |
| o . . S . |
| + . . . |
|. . . . . |
|. . o . . |
|..o. . . . |
+---[SHA256]-----+
mpiu@Master:~$ cat id_rsa.pub >> authorized_keys
cat: id_rsa.pub: No such file or directory
mpiu@Master:~$ cd .ssh
mpiu@Master:~/.ssh$ cat id_rsa.pub >> authorized_keys
mpiu@Master:~/.ssh$

```

Sprawdzenie połączenia SSH:

```

mpiu@Master:~$ ssh node1 hostname
mpiu@node1's password:
node1
mpiu@Master:~$

```

Instalacja GCC na węźle głównym:

```

master@Master:~$ sudo apt-get install build-essential
Reading package lists... Done
Building dependency tree
Reading state information... Done

```

Instalacja MPICH2 na wszystkich węzłach:

```
node1@node1:~$ sudo apt-get install mpich
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Utworzenie pliku o nazwie machinefile w katalogu domowym mpiu z nazwami węzłów:

```
mpiu@Master: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 machinefile

node3:4 # rozpocznie 4 procesy na node3
node2:2 #rozpocznie 2 procesy na node2
node1 #rozpocznie 1 proces na node1
Master #rorpocznie 1 proces na Masterze
```

Testowanie programu mpi

W zasobie /mirror tworzymy poniższy program:

```
mpiu@Master: /mirror
File Edit View Search Terminal Help
GNU nano 2.9.3 mpi_hello.c

#include <stdio.h>
#include <mpi.h>

int main(int argc, char** argv) {
    int myrank, nprocs;

    MPI_Init(&argc, &argv);
    MPI_Comm_size(MPI_COMM_WORLD, &nprocs);
    MPI_Comm_rank(MPI_COMM_WORLD, &myrank);

    printf("Hello from processor %d of %d\n", myrank, nprocs);

    MPI_Finalize();
    return 0;
}
```


Kompilujemy i uruchamiamy:

```
File Edit View Search Terminal Help
mpiu@Master:/$ mpiexec -n 8 -f machinefile ./mpi_hello
Hello from processor 0 of 8
Hello from processor 1 of 8
Hello from processor 2 of 8
Hello from processor 3 of 8
Hello from processor 4 of 8
Hello from processor 5 of 8
Hello from processor 6 of 8
Hello from processor 7 of 8
```