ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ “СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИИ И ИНФОРМАТИКИ”

Кафедра ВС

Лабораторная работа № 1
«РАСПРЕДЕЛЁННЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ»

Выполнил:
ст. гр. МГ-165 Марков В.А.
Проверил:
Фульман В.О.

Новосибирск 2016

Задание на лабораторную работу

1. Подключитесь к ресурсу ssh://jet.cpct.sibsutis.ru:22. После первого подключения измените пароль для своей учетной записи.

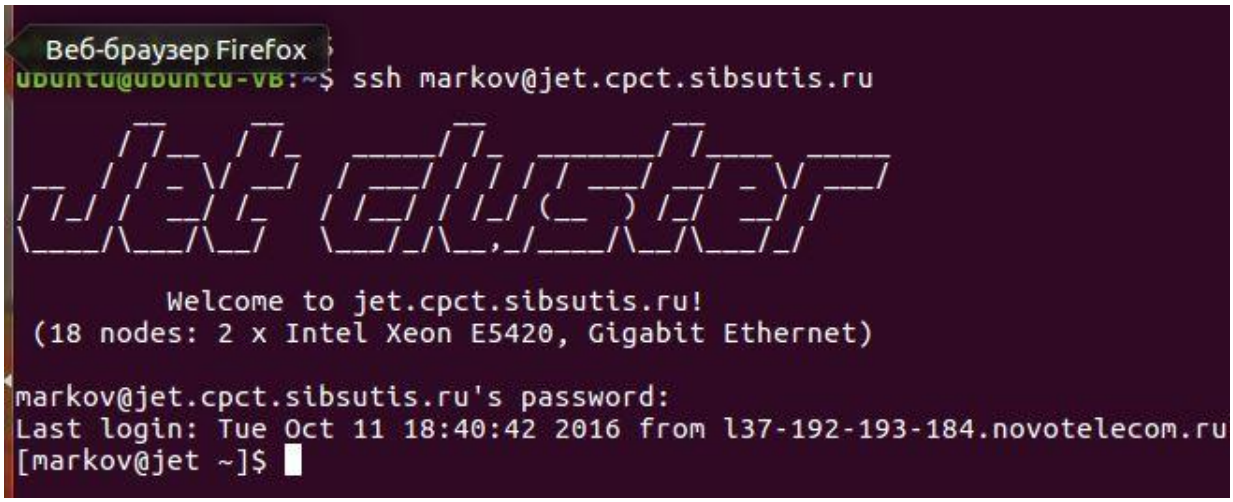

Рисунок 1 - Подключение

2. Подготовьте программное обеспечение, реализующее алгоритм умножения двух прямоугольных матриц целых чисел. Размеры матриц задаются параметрами командной строки. Исходные матрицы генерируются псевдослучайным образом (стандартный генератор). Исходные матрицы и результат их перемножения выводятся в стандартный поток вывода. Язык программирования и средства разработки, с использованием которых будет реализовано программное обеспечение, выбираются из числа доступных в распределённой вычислитель- ной системе Jet.

Листинг 1 - Умножение матриц

```
#include <iostream>
#include <vector>
#include <random>

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

using namespace std;

void generatingMatrix(vector<vector<int>> &a, int row, int
col)
{
    std::default_random_engine generator;
    std::uniform_int_distribution<int> distribution(0, 100);

    cout << "Generating matrix:" << endl;

    for (auto i = 0; i < row; ++i) {
        vector<int> v(col);
        a.push_back(v);
```

```
    }

    for (auto i = 0; i < row; ++i) {
        for (auto j = 0; j < col; ++j) {
            a[i][j] = distribution(generator);
        }
    }

    for (auto &i : a) {
        for (auto &j : i) {
            cout << j << " ";
        }
        cout << endl;
    }
}

void          multiplyMatrix(vector<vector<int>>          &a,
vector<vector<int>> &b, int m, int n, int q)
{
    vector<vector<int>> c;

    for (auto i = 0; i < m; ++i) {
        vector<int> v(q);
        c.push_back(v);
    }

    cout << "The new matrix is:" << endl;

    for (auto i = 0; i < m; ++i) {
        for (auto j = 0; j < q; ++j) {
            c[i][j] = 0;

            for (auto k = 0; k < n; ++k) {
                c[i][j] = c[i][j] + (a[i][k]*b[k][j]);
            }
            cout << c[i][j] << " ";
        }
        cout << endl;
    }
}

int main(int argc, char const *argv[])
{
    int m, n; // matrix 1 row, col
    int p, q; // matrix 2 row, col

    vector<vector<int>> matrix1;
    vector<vector<int>> matrix2;

    if (argc < 5) {
        cerr << "usage: m(row1) n(col1) p(row2) q(col2)" <<
endl;
        return -1;
    }

    m = atoi(argv[1]);
    n = atoi(argv[2]);
    p = atoi(argv[2]);
    q = atoi(argv[4]);

    generatingMatrix(matrix1, m, n);
    generatingMatrix(matrix2, p, q);

    multiplyMatrix(matrix1, matrix2, m, n, q);
```

```
    return 0;
}
```

3. Опишите задачу, требующую для своего решения один узел и одно ядро на нем. При решении задачи должно запускаться программное обеспечение, реализованное в п. 2. Размеры перемножаемых матриц задаются в виде трех чисел: M (количество строк в 1 таблице), N (ко- личество столбцов в 1 таблице и строк во 2 таблице), P (количество столбцов 2 таблицы), где M – линейный номер задач в её полном идентификаторе, N – удвоенное значение M, P – длина строки, содержащей сетевое имя узла, на котором выполняется задача.

Листинг 2 - Конфигурация запуска

```
#PBS -N dcs_lab1
#PBS -l nodes=1:ppn=1
#PBS -j oe

cd $PBS_O_WORKDIR

#string="3434.jet.cluster.local"
string=$PBS_JOBID

col_m1=0

row_m1=$(echo $string | tr -dc '0-9')
let "col_m1 = 2 * row_m1"

foo=${string#$row_m1"."}
foo1=${foo%%.*}

col_m2=${#foo1}

echo "1 M "$row_m1
echo "1 N "$col_m1
echo "2 M "$col_m1
echo "2 P "$col_m2

cat $PBS_NODEFILE

./task $row_m1 $col_m1 $col_m1 $col_m2
```

4. Поставьте задачу по умножению матриц в очередь. Посмотрите состояние очереди. Дождитесь завершения решения задачи и убедитесь в правильности выполнения программного обеспечения.

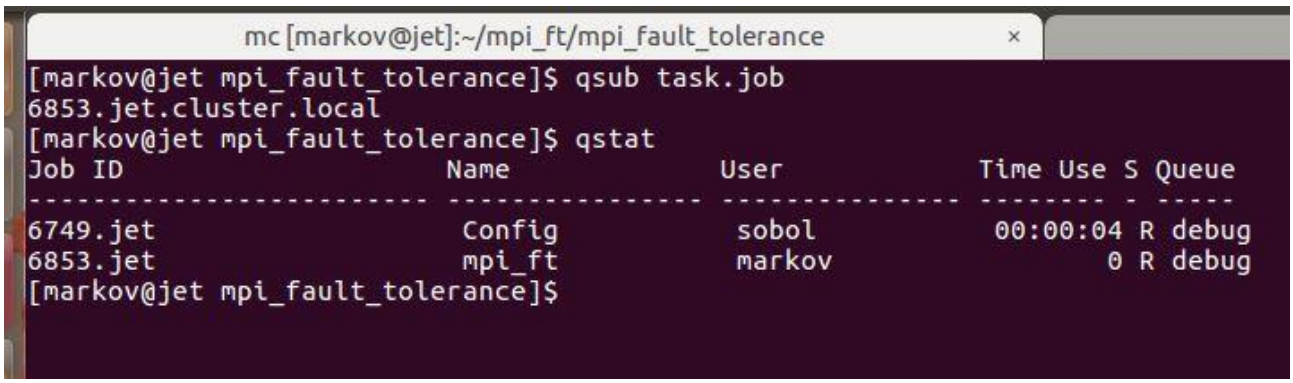

Рисунок 2 – Мониторинг исполнения задачи

5. Сконфигурируйте учетную запись в системе Jet так, чтобы при подключении по SSH использовалась пара открытый/закрытый ключ. Убедитесь, что теперь при подключении к системе, аутентификация происходит автоматически (без запроса паролей).

Листинг 3 – Конфигурация беспарольного доступа

```
ssh-keygen -t rsa
scp .ssh/id_rsa.pub markov@jet.cpct.sibsutis.ru:
ssh markov@jet.cpct.sibsutis.ru
"cat ./id_rsa.pub >> ./.ssh/authorized_keys"
```

6. Используя технологию X11 forwarding запустите графический редактор gedit и создайте в домашнем каталоге на основном узле системы Jet текстовый файл, в котором разместите информацию о себе: номер группы, учетная запись, ФИО.

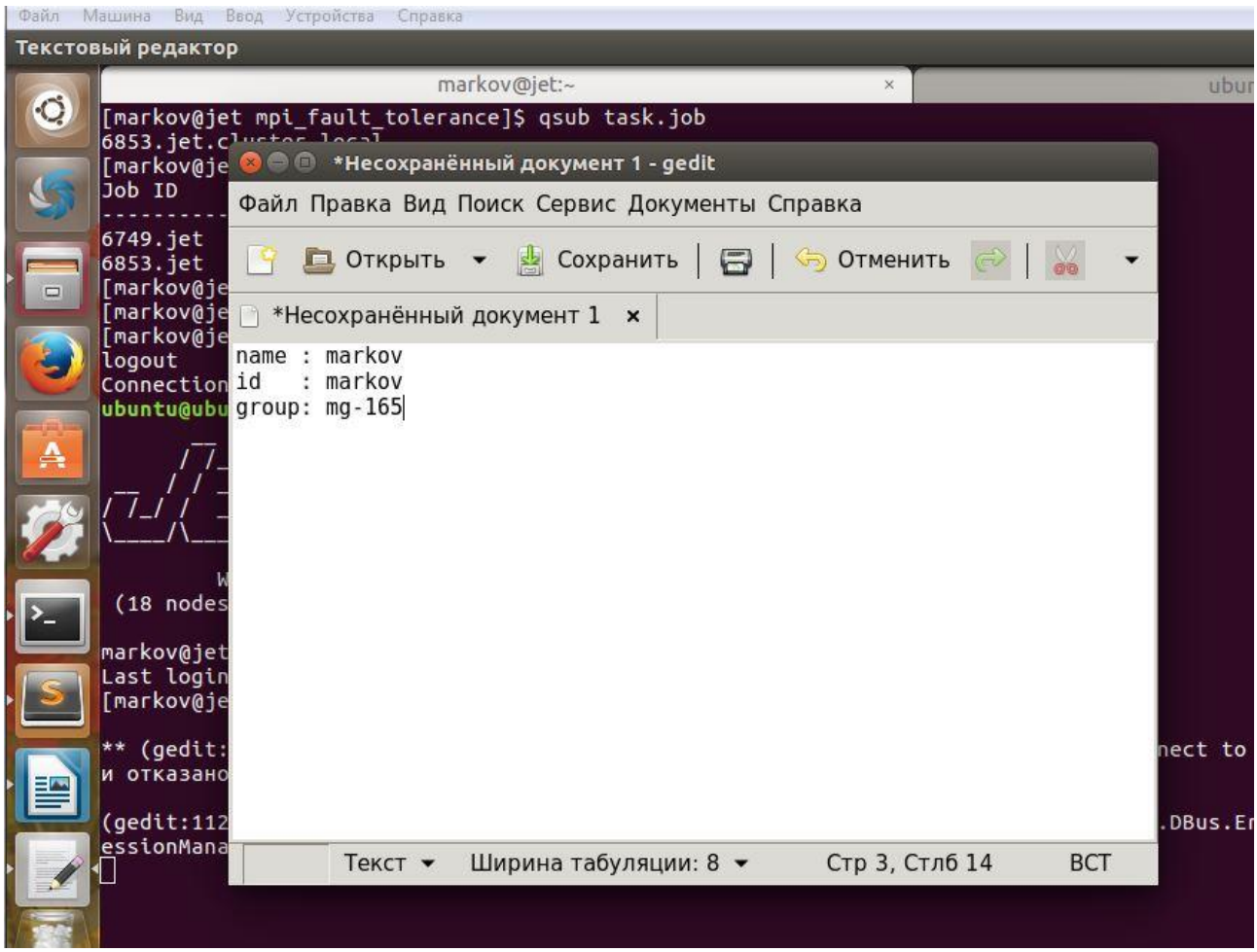

Рисунок 3 – Запуск «графический» приложений по ssh

7. Создайте описание задачи, требующей для своего решения два вычислительных узла и три вычислительных ядра на каждом. Запустите задачу в интерактивном режиме. После получения доступа к командной строке выведите на экран содержимое всех переменных среды окружения, начинающихся с префикса PBS_. Выведите на экран содержимое файла, в котором указаны имена узлов, выделенных для решения задачи. Используя команду mount определите каким образом сконфигурирована файловая система на вычислительном узле.

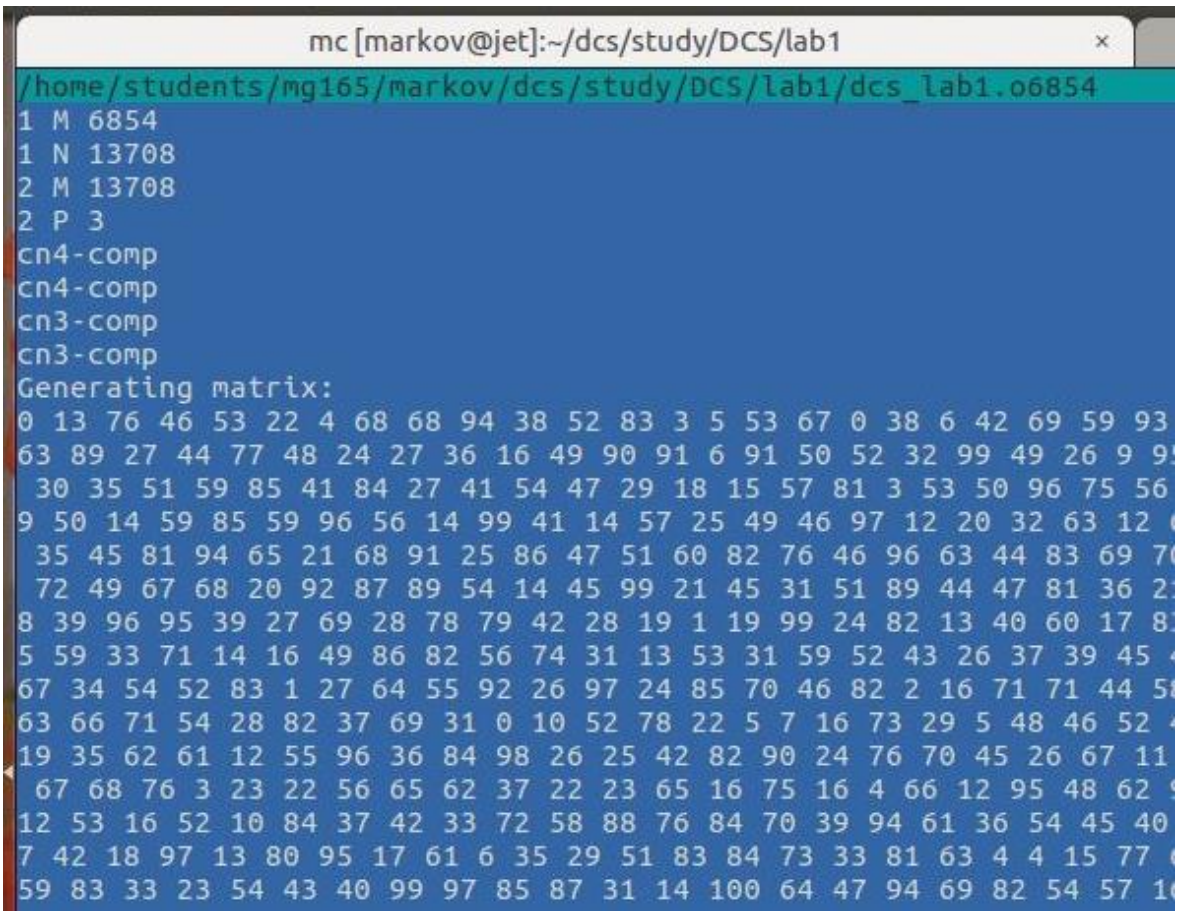

Рисунок 4 – Результат работы

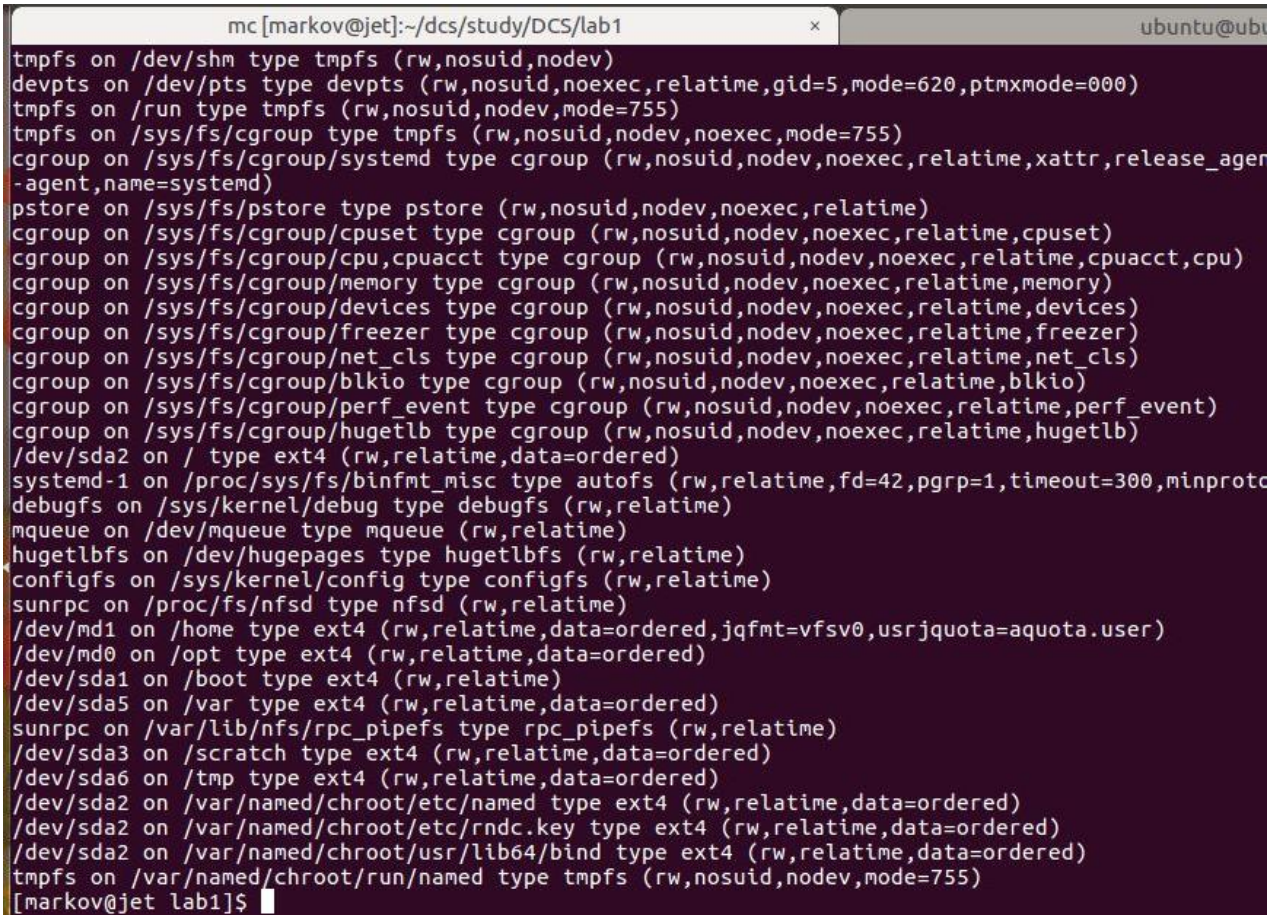

Рисунок 5 – Конфигурация файловой системы

8. Разработайте распределённое программное обеспечение реализующее следующий функционал:

- Модуль отображения состояния распределённой вычислительной системы. Использует информацию, получаемую от первого модуля или загружаемую из локального фай- ла (текущее состояние распределённой системы должно сохранятся в локальный файл). Интерфейс модуля отображения состояния распределённой системы значения не имеет (может быть текстовый, графический и т. п.)
- Модуль вывода информации о состоянии вычислительных узлов системы Jet. Запуск модуля возможен только на внешнем узле. Состоянием узлов считаем значения полей: state, power_state и jobs.

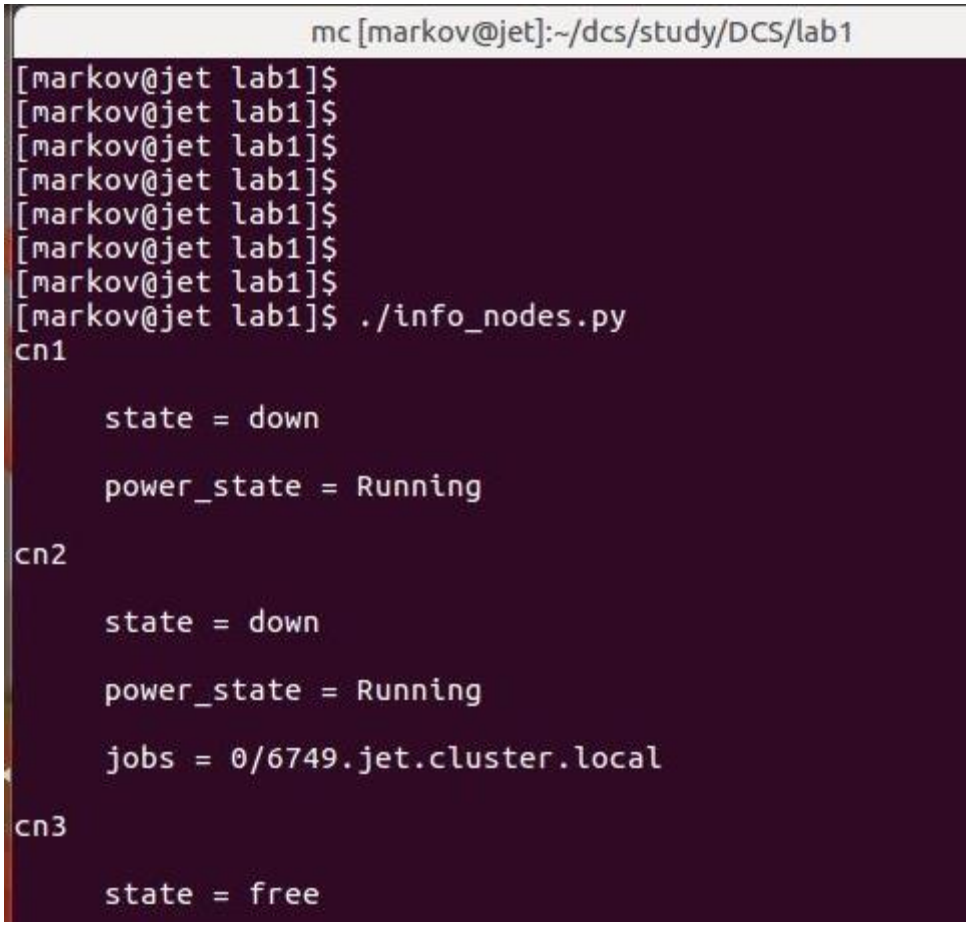

Рисунок 6 – Вывод конфигурации кластера

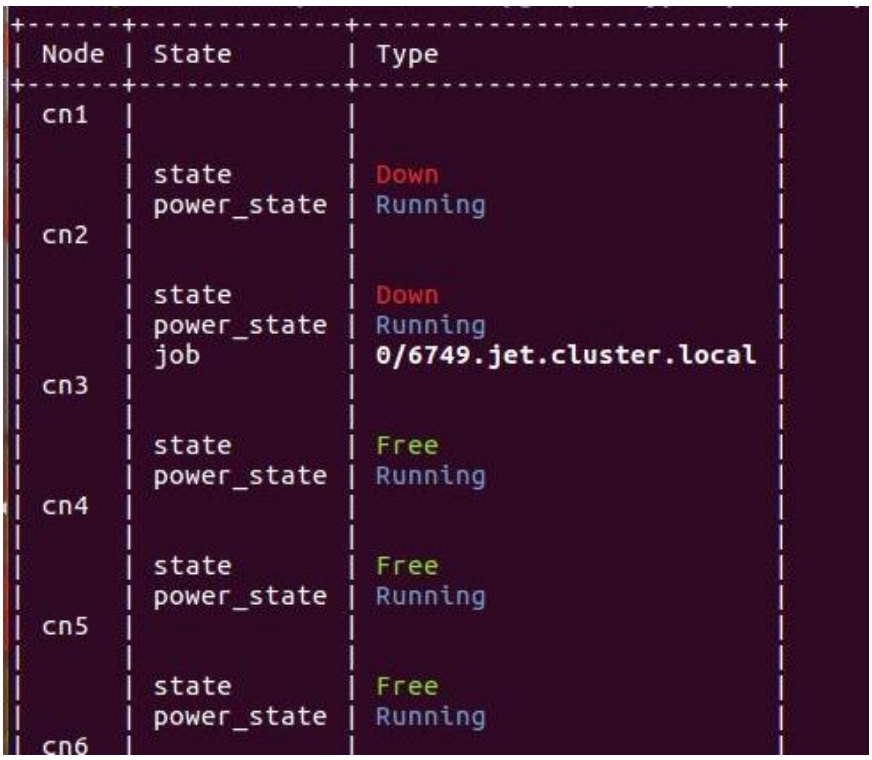

Рисунок 7 – Удаленный доступ к выводу конфигурации кластера