

МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра дискретной математики и информационных технологий

**РАЗРАБОТКА СИСТЕМЫ МОНИТОРИНГА ОТКАЗА УЗЛОВ НА  
ПРИМЕРЕ КЛАСТЕРА СГУ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 421 группы  
направления 09.03.01 — Информатика и вычислительная техника  
факультета КНиИТ  
Насонова Максима Олеговича

Научный руководитель  
доцент, к. ф.-м. н.

\_\_\_\_\_

И. Д. Сагаева

Заведующий кафедрой  
доцент, к. ф.-м. н.

\_\_\_\_\_

Л. Б. Тяпаев

Саратов 2020

## ВВЕДЕНИЕ

В настоящее время очень большую важность приобретают системы большой вычислительной мощности. К таким системам можно отнести суперкомпьютеры, вычислительные кластеры и другие подобные системы. Такие системы играют важные роли в нашем информационном обществе уже довольно много времени. С помощью современных вычислительных систем всё больше задач решается с их помощью. Это такие задачи как различные моделирования физических процессов, расчёт каких либо биологических систем, задач по криптографии, вопросы, связанные с прогнозированием погоды, исследования земной коры и многие другие задачи.

В основном подобные вычислительные системы ставят в научно-исследовательских институтах. Они требуют больших специализированных помещений для установки оборудования. Сам же пульт управления и контроля над вычислительной системой находится в отдельном помещении, к которому имеет доступ ограниченное количество людей. Таким образом, получается, что вся мощь суперкомпьютера может быть не использована, что несёт в себе убытки, как в научном плане, так и в материальном.

Кластер – это группа компьютеров, которые работают вместе и составляют единый унифицированный вычислительный ресурс. Хотя кластер и состоит из множества машин, операционных систем и приложений, пользователи «видят» его как одну систему. Объединение в кластеры позволяет создавать высокопроизводительные и надежные системы с использованием стандартных структурных компонентов. Возникновение кластеров было обусловлено ростом популярности интернета – стандартные серверные процессоры и системы, рассчитанные на обработку больших объемов данных, начали использоваться для обслуживания внешних приложений (хотя каждая система, как правило, управлялась отдельно, имела собственную операционную систему и прикладное ПО), что обеспечивало более высокую надежность.

Однако для обеспечения бесперебойной точной работы, а так же сохранности данных и оборудования, требуется своевременное выявление ошибок и отказа системы для этого потребуются обновить образовательно-научный вычислительный кластер СГУ, а так же написать модули для быстрой разветки кластера. На основе всего этого будет построена система мониторинга состояния узлов.

Целью дипломной работы является разработка системы мониторинга узлов на базе существующего вычислительного кластера с обеспечением полного контроля над ним.

Была поставлена задача построить систему мониторинга узлов кластера и обновить программное обеспечение, опираясь на документацию и старую версию ПО. Было принято решение разбить задачи на следующие этапы:

1. Выбор оптимальных методов решения
2. Изучение основ работы с Linux;
3. Изучение пакетного менеджера и puppet ;
4. Изучение программного обеспечения Prometheus;
5. Изучение программного обеспечения Grafana;
6. Настройка и написание модулей;
7. Тестирование готовой системы и ввод ее в эксплуатацию;

## КРАТКОЕ СОДЕРЖАНИЕ

В первом разделе дипломной работы рассматриваются общие понятия, такие как сервер, кластер, мониторинг, управление. А далее в трех подразделах рассматриваются описание аппаратной части, области применения, а так же постановка задачи и выявление требуемых ресурсов.

В первом подразделе говорится о постановке задачи, ее сложностях и особенностях реализации, об автоматизации работы и ускорениях процессов.

Во втором описывается аппаратная часть кластера.

В третьем подразделе говорится про области применения, как и где используется, и почему так это так важно при работе с большими вычислительными мощностями.

Во втором разделе работы речь пойдет о выполнении поставленной задачи, и начнем мы с обновления операционной системы и ее базовой настройки, а так же о настройке менеджера конфигурации.

Кроме того, для выполнения данной задачи, необходимо было получить дополнительные знания в следующих областях:

1. Установка операционной системы Linux Fedora и базовая настройка;
2. Основы работы с Linux;
3. Основы создания Kickstart файла;
4. Изучение пакетного менеджера;
5. Изучение программы puppet;

Далее в теоретической части работы описываем используемые технологии и приводим все необходимые предварительные сведения для практической реализации.

В третьей разделе говорится о практической части работы. Раздел содержит 7 подразделов.

В первом подразделе описаны установка и настройка мониторинга, представлено описание используемого программного обеспечения Prometheus и Grafana.

Во втором описывается загрузка всех компонентов.

Третий подраздел посвящен установке систем. В четвертом и пятом подразделах описывается назначение прав и настройка мониторинга. В шестом рассматривается запуск и проверка системы. И в седьмом подразделе говорится о модуле автозапуска.

На протяжении всей практической части приводятся скриншоты настройки и листинг кода.

В этой части работы описывается реализуемый программный продукт, демонстрируются примеры его тестовых запусков и сделать соответствующие выводы.

Аппаратное обеспечение.

Вычислительный кластер ПРЦНИТ состоит из 13 узлов:

один управляющий (`mpi-m1.sgu.ru`) и 12 вычислительных.

Структура вычислительных узлов гибридна и включает в себя:

1. Девять узлов на процессорах Intel Xeon E5405.
2. Два узла на процессорах Intel Xeon E5-2660.
3. Один узел на процессоре Intel Xeon Phi с архитектурой Intel MIC. В качестве управляющего узла для кластера КНИИТ используется `mpim1.sgu.ru`.

В состав кластера КНИИТ входят:

4. Сервер для расчетов на GPU, включающий в себя две видеокарты Nvidia Tesla k20m.
5. Один сервер на процессоре Intel Xeon Phi с архитектурой Intel MIC.
6. Сервер на процессоре Intel Xeon E5-2640 1.2 Программное обеспечение. Системное ПО Операционная система: Fedora 21. Графический интерфейс отсутствует. Для запуска задач используется менеджер ресурсов TORQUE с планировщиком задач Maui. Компиляторы и интерпретаторы
7. Intel C/C++ Compiler 13: компилятор запускается командами `icc`, `icpc`. Для компиляции MPI-приложений нужно использовать команды `mpiicc`, `mpiicpc`.
8. Intel Fortran Compiler 13: компилятор запускается командой `ifort`. Для компиляции MPI-приложений нужно использовать команду `mpiifort`.
9. GCC 4.9.2: компилятор запускается командами `gcc`, `g++`, `gfortran`, `f95`. Для компиляции MPI-приложений нужно использовать команды `mpicc`, `mpic++`, `mpif77`, `mpif90`.
10. CUDA 7.0: компилятор запускается командой `nvcc`.
11. Python версий 2.7 и 3.4: интерпретатор запускается командами `python` и `python3`, соответственно. Библиотеки MPI
12. MPICH 3.1;

13. Intel MPI 4.1.

14. OpenMPI 1.8

Математические пакеты и библиотеки

Нативные библиотеки: ARPACK, BLACS, CVOODE, GMP, GSL, LAPACK, METIS, OpenBLAS, ParMETIS, MPFR, MUMPS, ParPACK, PETSC, SCALAPACK, Sundials, SuiteSparse, UMFPACK, XBLAS.

Библиотеки Intel: MKL, TBB.

Библиотеки CUDA: cuBLAS, cuDNN, cuFFT, cuRAND, cuSOLVER, CuSPARSE, NPP.

Библиотеки Python: MPI4Py, NumPy, SciPy.

Специализированное ПО для расчетов: Elmer 6.1, Gaussian 09 с TCP Linda 8.

Система мониторинга представляет из себя связку из 2 программных продуктов, модули которых прописываются в системе и в дальнейшем используются.

Установка и настройка.

Мониторинг работы кластера производится с помощью nagios через web интерфейс по адресу <http://mpi-m1.sgu.ru/nagios/>. Кроме того сервис может высылать оповещения через Google Hangouts. Чтобы добавить себя в систему оповещений нужно сделать следующее:

- Добавить свои данные в файл `/etc/puppet/modules/nagios/files/contacts.cfg`
- Зайти в аккаунт Google, указанный в файле `/etc/puppet/modules/nagios/files/sendxmpprc`, выслать на свою почту, указанную в предыдущем пункте приглашение в диалог Hangouts и принять это приглашение. После вам в Hangouts будут приходить оповещения об изменении состояния сервисов в nagios.

Настройки сервиса находятся в папке `/etc/nagios`. Модуль puppet, использующийся для настройки данного сервиса, содержит скрипты для проверки работы сервисов, конфигурацию nagios и список необходимых проверок для каждого узла кластера. Список рабочих узлов и их разбиение на группы хранится в файле `hosts.cfg`.

Для списка используется два файла: `localservices.cfg` и `remoteservices.cfg`. Соответственно, первый описывает сервисы, которые необходимо отслеживать на корневом узле, а второй - сервисы, которые необходимо отслеживать

на рабочих узлах. Во втором файле для каждого сервиса необходимо указывать, для каких групп или узлов его использовать. Кроме того для мониторинга нагрузки на сервера используется `ganglia(mpim1.sgu.ru/ganglia)`.

Точно так же, как и для `nagios`, конфигурация сервиса рассылается при помощи `puppet`.

Использовать будем 2 системы мониторинга:

`Prometheus` — система мониторинга, обладающая возможностями тонкой настройки метрик. Она будет полезна для отслеживания состояния работы сервисов на низком уровне. Мониторинг работы кластера производится с помощью `prometheus` через web интерфейс по адресу `http://cluster.sgu.ru:9090`. Кроме того сервис может высылать оповещения через `Google Hangouts`, `Slack`.

`Grafana` — это платформа с открытым исходным кодом для визуализации, мониторинга и анализа данных. Этот инструмент, в сочетании с `Graylog`, — часть нашей двухсторонней системы мониторинга поведения пользователей и производительности системы. `Grafana` позволяет пользователям создавать дашборды с панелями, каждая из которых отображает определенные показатели в течение установленного периода времени. Каждый дашборд универсален, поэтому его можно настроить для конкретного проекта или с учетом любых потребностей разработки, web интерфейс находится по адресу `http://cluster.sgu.ru:3000`.

Установка `Prometheus Linux`:

После того, как мы скачали архив `prometheus`, необходимо его распаковать и скопировать его содержимое по разным каталогам. Для начала создаем каталоги, в которые скопируем файлы для `prometheus`:

```
mkdir /etc/prometheus
mkdir /var/lib/prometheus
```

Распакуем наш архив:

```
tar zxvf prometheus-*.linux-amd64.tar.gz
```

и перейдем в каталог с распакованными файлами:

```
cd prometheus-*.linux-amd64
```

Распределяем файлы по каталогам:

```
cp prometheus promtool /usr/local/bin/  
cp -r console_libraries consoles prometheus.yml /etc/prometheus
```

Установка Grafana Linux:

```
\$sudo apt-get install grafana
```

Запускаем prometheus командой:

```
/usr/local/bin/prometheus --config.file /etc/prometheus/prometheus.yml --  
storage.tsdb.path /var/lib/prometheus/ --  
web.console.templates=/etc/prometheus/consoles -  
web.console.libraries=/etc/prometheus/console_libraries
```

мы увидим лог запуска — в конце «Server is ready to receive web requests»:

```
level=info ts=2019-08-07T07:39:06.849Z caller=main.go:621 msg="Server is
```

Открываем веб-браузер и переходим по адресу <http://cluster.sgu.ru:9090> — загрузится консоль Prometheus:

Убедимся, что сервис Grafana действительно запущен и не произошло никаких ошибок:

```
\$sudo systemctl status grafana-server
```

Если всё прошло хорошо, то в выводе вы увидите сообщение `active(running)`, выделенное зеленым цветом.

Установка завершена.



## ЗАКЛЮЧЕНИЕ

В ходе выполнения данной работы были получены практические и теоретические знания и навыки работы с программным обеспечением, написания модулей и настройке серверов и мониторинга.

По итогам работы был обновлен образовательно-научный вычислительный кластер СГУ.

Созданы и написаны вручную модули, для быстрой развертки вычислительного кластера и мониторинга состояния узлов системы. Работа модулей была проверена в рабочем режиме кластера. Разработанная система мониторинга прошла тестирование в рабочем режиме кластера СГУ, что позволяет сделать вывод что система надежна и отказоустойчева. Все, даже незначительные изменения параметров отображаются в ходе работы и администратор в курсе всех изменений в любом месте и в любое время где бы он не находился. В данный момент система используется в вычислительном кластере СГУ.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Структура сети маршрутизаторов Интернет (2006 г.). URL: <http://www.cise.ufl.edu/research/sparse/mat/Pajek/internet.mat> (дата обращения: 04.02.2020).
- 2 Официальный сайт URL: <https://prometheus.io/> (дата обращения: 01.02.2020).
- 3 Официальный сайт URL: <https://grafana.com/> (дата обращения: 05.02.2020).
- 4 Puppet 3 Cookbook August 25, 2013 274 pages
- 5 Linux Fedora Core 2. Практическое руководство Издательство Корона-Принт, Век +, НТИ 2005 688
- 6 А. Сергеев "Основы локальных компьютерных сетей"(2016)
- 7 Андрей Кушнир: Сборка сервера. Руководство администратора Эксмо-Пресс, (2007 г)
- 8 Построение и использование кластерных вычислительных систем К.В. Корняков, А.Н. Свистунов, А.В. Сенин, А.В. Шишков Нижний Новгород (2007)
- 9 Эделман, Лоу, Осуолт: Автоматизация программируемых сетей 616 (Офсет) ДМК-Пресс, 2019 г.
- 10 Официальный сайт URL: <https://getfedora.org/ru/> (дата обращения: 02.04.2020)
- 11 Шафрин Ю., Основы компьютерной технологии.- М.: АБФ, 1997
- 12 Журнал для пользователей персональных компьютеров Мир ПК.
- 13 Официальный сайт URL: <https://compress.ru/> (дата обращения: 08.02.2020)
- 14 Еженедельник для предпринимателей и специалистов в области информационных технологий  
ComputerWeek Moscow.
- 15 Linux: руководство по операционной системе : В 2 т. / Р. Петерсен; Пер. с англ. С. М. Тимачева;

Под ред. М. В. Коломыцева. - 2. изд., перераб. и доп. - Киев и др. : BHV, 1998.

- 16 Операционная система UNIX". А. Робачевский, Стесик Ольга Львовна, Немнюгин Сергей Андреевич  
BHV, 2010 г. 656
- 17 Журнал «Upgrade» 04.08.05. Официальный сайт URL:  
<https://upweek.ru/upgrade-844>  
(дата обращения:06.04.2020)
- 18 Журнал «Компьютер пресс» Официальный сайт URL:  
<https://compress.ru/> (дата обращения:  
10.03.2020)
- 19 Официальный сайт URL: <https://ubuntu.ru/> (дата обращения: 08.02.2020)