МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

ВЫСШИЙ КОЛЛЕДЖ ПГТУ «ПОЛИТЕХНИК»

Специальность 09.02.06 (шифр)

Сетевое и системное администрирование (наименование)

Фамилия Юшманов

Имя Юрий \_\_\_

Отчество Михайлович \_\_\_

**Выпускная квалификационная работа**

Дипломный проект на тему:

*Разработка программного обеспечения для мониторинга веб-серверов*

Заместитель директора по УВР Скоробогатова Анна Александровна, преподаватель высшей квалификационной категории

(ф.и.о. уч. степень, звание, категория, должность)

Руководитель Глозштейн Даниил Александрович, старший преподаватель кафедры информационной безопасности ФИиВТ ФГБОУ ВО «ПГТУ»

(ф.и.о. уч. степень, звание, категория, должность)

№ приказа о допуске к защите ВКР\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ВКР начата 31.03.2022 г.

ВКР окончена 06.06.2022 г.

Оценка Государственной экзаменационной комиссии по защите \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Секретарь ГЭК\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Герасимова И.Г. /

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

ВЫСШИЙ КОЛЛЕДЖ ПГТУ «ПОЛИТЕХНИК»

|  |  |
| --- | --- |
| «СОГЛАСОВАНО»  Представитель работодателя  \_\_\_\_\_\_\_\_\_\_/И.В. Абукаев/  (подпись)  «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_20 22 г. | «УТВЕРЖДАЮ»  Зам. директора по УВР  \_\_\_\_\_\_\_\_\_\_\_\_/А.А. Скоробогатова/  (подпись)  «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_20 22 г. |

**З А Д А Н И Е**

**НА ВЫПОЛНЕНИЕ ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ**

Обучающийся Юшманов Юрий Михайлович

(фамилия, имя, отчество)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Специальность 09.02.06 Сетевое и системное администрирование

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Форма обучения очная Группа СиСА-41

Тема выпускной квалификационной работы Разработка программного обеспечения для мониторинга веб-серверов\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(утверждена приказом ректора от 28.03.2022 г. № 382-ОН)

Срок сдачи законченной ВКР 06.06.2022г.

**Содержание задания**

1. Исходные данные (заполняется с руководителем)

Параметры:

1. Диапазон напряжений питания В-4,75 … 5,25;

2. Потребляемый ток, не более mА – 100;

3. Функция обновления прошивки;

4. Диапазон рабочих температур, град.Цельсия +10 …+45;

5. Относительная влажность без конденсации, не более 35%;

6. Тип USB-разъема miniUSB B.

2. Содержание расчетно-пояснительной записки (заполняется с руководителем)

1. Обоснование схемы электрической принципиальной.

2. Конструкторская часть: изготовление, конструкция и разработка программатора.

3. Технологическая часть: технологическая подготовка производства, разработка технологического процесса изготовления платы.

4. Экономическая часть.

5. Безопасность и экологичность.

6. Экспериментальная часть: методика работы с прибором, описание программы MPLAB IDE.

3. Содержание графической части (заполняется с руководителем)

Слайд 1 – Схема электрическая принципиальная

Слайд 2 – Печатная плата

…………. и т.д.

(перечень графического материала, число листов формата)

КОНСУЛЬТАНТЫ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(должность, Ф.И.О., подпись, дата)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(должность, Ф.И.О., подпись, дата)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(должность, Ф.И.О., подпись, дата)

Руководитель ВКР

Старший преподаватель кафедры информационной безопасности ФИиВТ ФГБОУ ВО «ПГТУ», Глозштейн Даниил Александрович 31.03.2022 г. должность, ФИО подпись, дата

Задание принял к исполнению Юшманов Ю. М. 31.03.2022г.

Ф.И.О. обучающегося подпись, дата

Задание зарегистрировано: 31.03.2022г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

ВЫСШИЙ КОЛЛЕДЖ ПГТУ «ПОЛИТЕХНИК»

|  |  |  |
| --- | --- | --- |
|  |  | **УТВЕРЖДАЮ**  Зам. директора по УВР  \_/А.А. Скоробогатова/  «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2022 г. |

ГРАФИК

написания и оформления выпускной квалификационной работы

Ф.И.О. обучающегося Юшманов Юрий Михайлович

Тема выпускной квалификационной работы Разработка программного обеспечения для мониторинга веб-серверов

|  |  |  |  |
| --- | --- | --- | --- |
| №  п/п | Мероприятия | Сроки выполнения | Отметка руководителя о выполнении |
| 1. | Подбор литературы, ее изучение и обработка. Составление библиографии по основным источникам | до «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022 |  |
| 2. | Составление плана ВКР и согласование его с руководителем | до «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022 |  |
| 3. | Разработка и представление на проверку первой главы | до «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022 |  |
| 4. | Накопление, систематизация, анализ практических материалов | до «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022 |  |
| 5. | Разработка и представление на проверку второй главы | до «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022 |  |
| 6. | Согласование с руководителем выводов и предложений | до «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022 |  |
| 7. | Переработка (доработка) ВКР в соответствии с замечаниями и представление ее в учебную часть | до « 02 » июня 2022 |  |
| 8. | Разработка тезисов доклада для защиты | до « 03 » июня 2022 |  |
| 9. | Ознакомление с отзывом и рецензией | до « 11 » июня 2022 |  |
| 10. | Завершение подготовки ВКР к защите с учетом отзыва и рецензии | до « 12 » июня 2022 |  |

Руководитель ВКР \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«31» марта 2022 г.

СОДЕРЖАНИЕ

[​ ВВЕДЕНИЕ 7](#__RefHeading___Toc1598_178206272)

[​ 1. ПОСТАНОВКА ЗАДАЧИ И ОБЗОР ИСПОЛЬЗУЕМЫХ СРЕДСТВ 8](#__RefHeading___Toc1619_178206272)

[​ 1.1 Техническое задание 8](#__RefHeading___Toc1621_178206272)

[​ 1.2 Обзор использованных средств 10](#__RefHeading___Toc1623_178206272)

[1.2.1 Язык программирования 10](#__RefHeading___Toc1625_178206272)

[1.2.2 Базы данных 10](#__RefHeading___Toc1627_178206272)

[1.2.4 Среда разработки 11](#__RefHeading___Toc1633_178206272)

[1.2.5 Используемые средства в разработке 12](#__RefHeading___Toc1631_178206272)

[​ 2. ТЕОРЕТИКО-МЕТОДОЛОГИЧЕСКИЕ АСПЕКТЫ 14](#__RefHeading___Toc2123_178206272)

[​ 2.1 Единицы измерения информации 14](#__RefHeading___Toc4900_178206272)

[​ 2.2 Файловая система /proc 15](#__RefHeading___Toc2125_178206272)

[​ 3. ОПИСАНИЕ НАПИСАННОГО ПО 19](#__RefHeading___Toc4277_178206272)

[​ 3.1 Основы взаимодействия 19](#__RefHeading___Toc4383_178206272)

[​ 3.2 Используемые сервисы 20](#__RefHeading___Toc4385_178206272)

[​ 3.3 Проектирование базы данных 21](#__RefHeading___Toc2612_178206272)

[​ 3.4 Backend 22](#__RefHeading___Toc2614_178206272)

[​ 3.5 Frontend 23](#__RefHeading___Toc2616_178206272)

[​ 3.6 gRPC и background задачи 24](#__RefHeading___Toc2618_178206272)

[3.6.1 gRPC master 24](#__RefHeading___Toc2661_178206272)

[3.6.2 gRPC slave 24](#__RefHeading___Toc2663_178206272)

[3.6.3 background задачи 24](#__RefHeading___Toc2665_178206272)

[​ 3.7 Защита приложения 25](#__RefHeading___Toc2646_178206272)

[3.7.1 JWT 25](#__RefHeading___Toc2648_178206272)

[3.7.2 Двухфакторная авторизация 25](#__RefHeading___Toc2650_178206272)

[3.7.3 API-key 25](#__RefHeading___Toc2652_178206272)

[3.7.4 Управление пользователями 25](#__RefHeading___Toc2654_178206272)

[​ 4. ТЕСТИРОВАНИЕ 26](#__RefHeading___Toc2630_178206272)

[​ 5. РУКОВОДСТВО ПО УСТАНОВКЕ 27](#__RefHeading___Toc2632_178206272)

[​ 6. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ 28](#__RefHeading___Toc2634_178206272)

[​ 6.1 Управление пользователями 28](#__RefHeading___Toc2636_178206272)

[​ 6.2 Управление серверами 29](#__RefHeading___Toc2638_178206272)

[​ 6.3 Удаление приложения 30](#__RefHeading___Toc2640_178206272)

[6.3.1 Удаление client 30](#__RefHeading___Toc2642_178206272)

[6.3.2 Удаление master 30](#__RefHeading___Toc2656_178206272)

[​ ЗАКЛЮЧЕНИЕ 31](#__RefHeading___Toc2644_178206272)

[​ СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 32](#__RefHeading___Toc1600_178206272)

[​ ПРИЛОЖЕНИЯ 33](#__RefHeading___Toc1602_178206272)

# ВВЕДЕНИЕ

Мониторинг серверов – это одна из важнейших задач в работе системного администратора. Мало просто создать компьютерную инфраструктуру. За ней нужно ещё и присматривать, ведь в ней могут возникнуть какие-то неисправности. Причём, в данном случае не очень важно, идёт ли речь об аппаратных серверах или о виртуальных — сегодня в прикладном смысле их можно не различать.

В ходе мониторинга о наблюдаемом объекте собирают сведения, по которым можно своевременно узнать о возникновении проблемы и, соответственно, незамедлительно начать принимать меры по её устранению. В принципе, имеются универсальные извещатели о проблемах — пользователи информационной системы. Но сообщения от них всегда приходят позже, чем о них должен был бы узнать системный администратор.

В самом простом варианте мониторинга проверяют лишь факт работоспособности наблюдаемого объекта. Однако мониторинг может выявлять не только уже возникшие проблемы, но и те, вероятность появления которых возрастает. Например, наблюдение за объёмом свободной дисковой или оперативной памяти может предупредить о приближающемся их дефицита. Здесь следует сделать одно уточнение. Сбор данных бывает «контактным» или «дистанционным». В первом случае наблюдение ведётся программами, работающими на самом наблюдаемом сервере (их часто называют агентами). Во втором — программами, работающими на других компьютерах и наблюдающими по сети.

Во многих случаях для выявления самого факта функционирования оборудования, достаточно просто отправить ему некий запрос по сети, дождаться ответа и сравнить его с ожидаемым.

# 1. ПОСТАНОВКА ЗАДАЧИ И ОБЗОР ИСПОЛЬЗУЕМЫХ СРЕДСТВ

## 1.1 Техническое задание

**Назначение ВКР:** Создание базы данных для хранения данных о серверах, статистике, ошибках, логов, а так же создание веб-интерфейса для доступа к данным.

**Требование к системе администрирования:**

1. Авторизация и аутентификации пользователей.
2. Обработка данных серверов на мастер-сервере.
3. Наличие модуля realtime-метрик

**Требования к Интернет-проекту:**

1. Использование базы данных.
2. Двухфакторная авторизация пользователей.
3. Добавление списка серверов.
4. Преобразование метрик в удобные графики.

**Требования к базе данных:**

Необходимо разработать структуру базы данных, содержащую следующие таблицы:

1. Активные пользователи.
2. Сервера.
3. Статистика метрик серверов.

**Требования к web-интерфейсу:**

1. Интерфейс должен быть с современным дизайном, учитывающим разные устройства вывода информации – широкий, квадратный экран монитора, планшет, смартфон.
2. Корректное отображение web-интерфейса в браузерах Google Chrome, Mozilla Firefox, Opera.
3. Веб интерфейс должен быть singlepage-приложением

## 1.2 Обзор использованных средств

### 1.2.1 Язык программирования

На сегодняшний день существет множество языков программирования для создания web-приложений.

**PHP** – язык программирования, исполняемый на стороне веб серверах. В настоящий момент проектирования сайтов на PHP выполняется с помощью фреймворков. Самыми известными являются Yii, Laravel, Zend.

**Python** – высокоуровневый язык программирования, предназначенный для создания десктопных и web-приложений. Активно используется в области машинного обущения и искусственного интелекта.

**TypeScript** –язык программирования, представленный Microsoft в 2012 году и позиционируемый как средство разработки веб-приложений, расширяющее возможности JavaScript

### 1.2.2 Базы данных

Наиболее важным и обязательным компонентом в современном проекте любой степени сложности стала система управления базами данных. В web-индустрии наибольшую популярность получили серверные реляционные базы данных.

**Oracle Database**

Объектно-реляционная система управления базами данных компании Oracle.

**Достоинства:**

* Позволяет использовать в высоконагруженных проектах
* Хорошее сжатие данных
* Хорошая система защиты от несанкционированного доступа

**Недостатки:**

* Высокая цена
* Завышенные требования к серверному оборудованию

**MySQL**

свободная реляционная система управления базами данных. Разработку и поддержку MySQL осуществляет корпорация Oracle.

**Достоинства:**

* Легкость в развертке и настройке
* Быстродействие
* Бесплатное распространение

**Недостатки:**

* Наиболее быстрый тип таблиц MyISAM менее надежен

### 1.2.4 Среда разработки

Наиболее важным элементом в процессе разработки является выбор среды разработки (IDE). Зачастую именно от нее зависит качество и скорость написания кода приложения.

**VisualStudio** – наиболее популярная среда разработки проектов, за счет не малого функционала, который может быть расширен. Главным недостатком явзяется то, что VS запускается только на Windows.

**VisualStudio Code** – решает проблему крупноформатности, но является более урезанной.

**JetBrains PyCharm** – кросплатформенный аналог VS. Поддерживает python и TypeScript.

### 1.2.5 Используемые средства в разработке

В результате сравнения, были выбраны следующие средства:

1. Backend
   1. Язык программирования python версии >=3.10
   2. Фреймворк FastAPI версии >=0.75.0
   3. ORM Tortoise-ORM версии >=0.19.0
   4. ASGI-сервер uvicorn версии >=0.17.6
   5. WebSocket-сервер socketio версии >=5.6.0
   6. RPC-сервер grpcio версии >=1.45.0
   7. Пакетный менеджер poetry версии 1.1.12
2. Frontend
   1. Язык программирования TypeScript версии 4.4.4
   2. Фреймворк React версии 17.0.2
   3. UI-библиотека VKUI версии 4.28.1
   4. Менеджер сборки Vite версии 2.6.10
   5. Soket клиент socket.io-client версии 4.4.1
   6. Библиотека для построения диаграмм recharts версии 2.1.9
   7. XHR-клиент axios версии 0.26.1

При условии использования Tortoise-ORM СУБД может быть любая из ниже перечисленных:

1. SQLite3
2. PostgreSQL
3. MySQL
4. Microsoft SQL Server
5. Oracle

# **2.** **ТЕОРЕТИКО-МЕТОДОЛОГИЧЕСКИЕ АСПЕКТЫ**

## 2.1 Единицы измерения информации

**Бит** — единица измерения количества информации. 1 бит информации — символ или сигнал, который может принимать два значения: включено или выключено, да или нет, высокий или низкий, заряженный или незаряженный; в двоичной системе исчисления это 1 (единица) или 0 (ноль). Это минимальное количество информации, которое необходимо для ликвидации минимальной неопределенности.

**Байт** — единица хранения и обработки цифровой информации; совокупность битов, обрабатываемая компьютером одновременно. В современных вычислительных системах байт состоит из 8 бит и, соответственно, может принимать одно из 256 (от 0 до 255) различных значений (состояний, кодов). Однако в истории компьютерной техники существовали решения с иными размерами байта (например, 6, 32 или 36 бит), поэтому иногда в компьютерных стандартах и официальных документах для однозначного обозначения группы из 8 бит используется термин «октет».

В большинстве вычислительных архитектур байт — это минимальный независимо адресуемый набор данных.

Таблица 1 - Измерения в битах

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ГОСТ 8.417-2002 | | | Приставки МЭК | | | |
| Название | Символ | Степень | Название | Символ | | Степень |
| килобит | Kбит | 103 | кибибит | Kibit | Кибит | 210 |
| мегабит | Мбит | 106 | мебибит | Mibit | Мибит | 220 |
| гигабит | Гбит | 109 | гибибит | Gibit | Гибит | 230 |
| терабит | Тбит | 1012 | тебибит | Tibit | Тибит | 240 |
| петабит | Пбит | 1015 | пебибит | Pibit | Пибит | 250 |
| эксабит | Эбит | 1018 | эксбибит | Eibit | Эибит | 260 |
| зеттабит | Збит | 1021 | зебибит | Zibit | Зибит | 270 |
| йоттабит | Йбит | 1024 | йобибит | Yibit | Йибит | 280 |

Таблица 2 - Измерения в байтах

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ГОСТ 8.417-2002 | | | Приставки СИ | | Приставки МЭК | | | |
| Название | Обозначение | Степень | Название | Степень | Название | Обозначение | | Степень |
| байт | Б | 100 | – | 100 | байт | B | Б | 20 |
| килобайт | Кбайт | 103 | кило- | 103 | кибибайт | KiB | КиБ | 210 |
| мегабайт | Мбайт | 106 | мега- | 106 | мебибайт | MiB | МиБ | 220 |
| гигабайт | Гбайт | 109 | гига- | 109 | гибибайт | GiB | ГиБ | 230 |
| терабайт | Тбайт | 1012 | тера- | 1012 | тебибайт | TiB | ТиБ | 240 |
| петабайт | Пбайт | 1015 | пета- | 1015 | пебибайт | PiB | ПиБ | 250 |
| эксабайт | Эбайт | 1018 | экса- | 1018 | эксбибайт | EiB | ЭиБ | 260 |
| зеттабайт | Збайт | 1021 | зетта- | 1021 | зебибайт | ZiB | ЗиБ | 270 |
| йоттабайт | Ибайт | 1024 | йотта- | 1024 | йобибайт | YiB | ЙиБ | 280 |

## **2.2 Файловая система /proc**

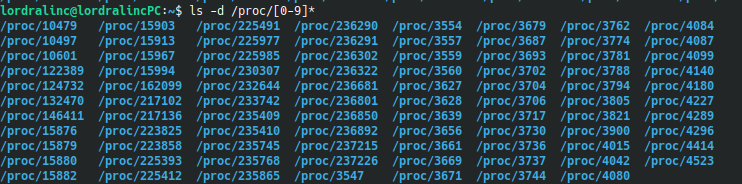
Файловая система /proc является особой для GNU/Linux. Это виртуальная файловая система и она не занимает места на диске. Это очень удобный способ получить информацию о системе, тем более, что большинство файлов в этом каталоге читаемы для человека. Большинство программ реально получают информацию из файлов в /proc, форматируют их своим способом, а затем отображают. Так делают все программы, которые отображают информацию о процессах, и некоторые из них мы уже видели (top, ps и сотоварищи). /proc это также хороший источник информации о аппаратных средствах, и таким же образом некоторые программы являются только интерфейсами к информации, содержащейся в /proc.

Существует также специальный подкаталог /proc/sys. Он позволяет изменять некоторые параметры ядра в реальном режиме времени или отображать их.

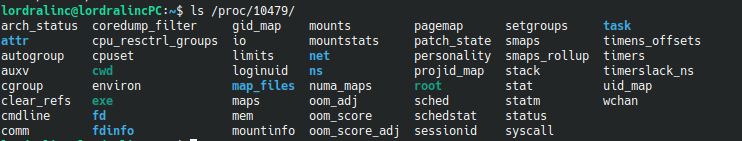
**Информация о процессах**

Если пролистать содержимое каталога /proc, мы увидим много каталогов, названиями которых являются номера. Эти каталоги содержат информацию о все запущенных в данный момент процессах в системе:

Посмотрим какая информация доступна для процесса 10479:

Рисунок 1 - Список запущенных процессов в /proc

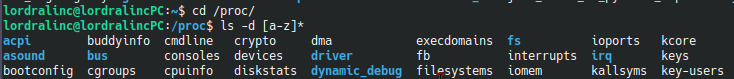
Каждый каталог содержит одинаковые входы. Вот краткое описание некоторых из них:

Рисунок 2 - Структура каталога для процесса

1. **cmdline**: этот (псевдо-) файл содержит целую командную строку, использованную для вызова процесса. Он не отформатирован: нет пробелов между программой и ее аргументами, и нет разделителя в конце строки. Чтобы посмотреть его, можно использовать: perl -ple 's,\00, ,g' cmdline.
2. **cwd**: это символическая ссылка на текущий рабочий каталог (следует из имени) процесса.
3. **environ:** Этот файл содержит все переменные окружения, определенные для процесса, в форме VARIABLE=value. Подобно cmdline, вывод не форматирован вообще: нет разделителей между различными переменными, и нет разделителя в конце. Единственное решение как его посмотреть: perl -pl -e 's,\00,\n,g' environ.
4. **exe**: это символическая ссылка на соответствующий запущенному процессу выполняемый файл.
5. **fd**: этот подкаталог содержит список файловых дескрипторов, открытых процессом в данный момент.
6. **root**: Это символическая ссылка на корневой каталог, используемый процессом. Обычно это будет /.
7. **status**: этот файл содержит различную информацию о процессе: название выполняемой программы, его текущее состояние, его PID и PPID, его реальные и эффективные UID и GID, использование им памяти, и другое.

**Информация об аппаратном обеспечении**

Кроме каталогов, связанных с различными процессами, /proc также содержит несметное количество информации об аппаратном обеспечении машины. Список файлов каталога /proc показывает следующее:

Рисунок 3 - Информация об аппаратном обеспечении

Полная распечатка файлов была бы слишком длинной, тем не менее описание некоторых из них:

1. **cpuinfo**: этот файл содержит, соответственно своему названию, информацию о процессоре(рах) в машине.
2. **modules**: этот файл содержит список модулей, которые используются ядром в данный момент, а также процент использования для каждого модуля. Фактически, это та же самая информация, которую предоставляет команда lsmod.
3. **meminfo**: этот файл предоставляет информацию о загрузке памяти на момент запроса. Более аккуратно форматированный вид этой же информации можно получить командой free.
4. **bus**: этот подкаталог предоставит вам информацию о всех периферийных устройствах, найденных на различных шинах компьютера. Информация внутри него редко вообще читаема, и главным образом с ней имеют дело и умеют форматировать внешние утилиты, такие как: lspcidrake, lspnp, и др.

# 3. **ОПИСАНИЕ НАПИСАННОГО ПО**

## **3.1 Основы взаимодействия**

## 3.2 Используемые сервисы

## 3.3 Проектирование базы данных

## 3.4 Backend

## 3.5 Frontend

## 3.6 gRPC и background задачи

### 3.6.1 gRPC master

### 3.6.2 gRPC slave

### 3.6.3 background задачи

## 3.7 Защита приложения

### 3.7.1 JWT

### 3.7.2 Двухфакторная авторизация

### 3.7.3 API-key

### 3.7.4 Управление пользователями

# 4. ТЕСТИРОВАНИЕ

# 5. РУКОВОДСТВО ПО УСТАНОВКЕ

# 6. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

## 6.1 Управление пользователями

## 6.2 Управление серверами

## 6.3 Удаление приложения

### 6.3.1 Удаление client

### 6.3.2 Удаление master

# ЗАКЛЮЧЕНИЕ

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

# ПРИЛОЖЕНИЯ