**4 ВЫБОР КОМПОНЕНТОВ СЕРВЕРА ДЛЯ РАБОТЫ С РАЗРАБОТАННОЙ ПРОГРАММОЙ**

Для применения алгоритма в разрабатываемой системе, он был запрограммирован, с использованием языка PHP. Алгоритм выполняется на удоленном сервере, взаимодействие с которым происходит по средствам API. В зависимости от задачи на удаленном сервере должно быть установлено то или иное ПО. В ходе работы были изучены различные пакеты программного обеспечения, среди которых был выбран позволяющий реализовать соответствующий требованиям сервер. Для того что бы было возможно решить поставленную задачу, и разработанная система могла развиваться а также расширять свой функционал, не требуя при этом много ресурсов, таких как время, финансы, с которых оплачивется труд разработчиков и т.д, удаленный сервер должен соостветствовать следующим требованиям. Возможность быстрого перехода с более слабых на более мощные сервера, легкость переноса всей серверной части, легкость адаптации в случае необходимости, рациональные и оправдывающие себя запроссы в финансовой поддержке и т.д.

**4.1 Компоненты сервера**

Выполнив исследования, различного серверного программного обеспечения, мною было принято решение в пользу FAMP, так как именно он идеально соответствовал требованиям. Далее будет подробно описанно что такое FAMP, и сравнение его каждой отдельно взятой компоненты с аналогами, где так же будут приведены более подробные пояснения, что именно побудило сделать выбор в пользу FAMP.

FAMP — комплекс серверного программного обеспечения. FAMP - это абривеатура составленная по первым буквам названий входящих в этот комплекс компонентов:

· FreeBSD — UNIX-подобная операционная система;

· Apache — веб-сервер;

· MySQL — система управления базой данных;

· PHP — язык программирования, ориентированный на веб разработку для серверной части.

Многим больше известен альтернативный набор LAMP, это тот же самый набор в котором вместо FreeBSD, используется операционная система Linux. Справедливо отметить что именно LAMP задал тон именования подобных наборов серверного ПО.

Все програмные продукты воходяшие в состав подобных комплексов разрабатывались отдельно как независимые проекты, некоторые из них разрабатывались разными компаниями. Такой комплекс стал популярным в ходе развития такой области как веб-разработка. Популярным он стал из-за своей гибкости а так же высокой производительности, так же на ряду с этим все компоненты распростроняются под открытой лицензией что приводит к снижению финансовых затрат для проектов использующих его. На сегоднешний день набор FAMP предостовляется многими хостинг-площадками.

**4.1.1 Apache**

HTTP-сервер Apache, Кроссплатформенное ПО, которое поддерживается практически всеми существующими операционными системами.

К главными преимуществами Apache относятся его надёжность и гибкость при конфигурировании. Все эти характеристики очень важны для разрабатывемой системы, кросс платформинность, обеспечивает гибкость в

проектировании сервера. Гибкость данного ПО заключается в том что если по каким либо причинам необходимо будет поменять операционную систему на сервере, например при необходимости использовать ПО под Linux сисемы а не под FreeBSD, которая была выбрана на момент написания работы, то у нас не возникнет проблем с переходом на другую ОС. В таблице 2.1 приведен список операционнх систем под которые адаптирован Apache.

**Таблица 2.1 ОС под которые адаптирован Apache.**

|  |  |  |
| --- | --- | --- |
| **№** | **Название** | **Поддержка** |
| 1 | OC на базе Linux | полная |
| 2 | ОС линии BSD | полная |
| 3 | Mac OS | полная |
| 4 | Microsoft Windows | полная |
| 5 | Novell NetWare | полная |
| 6 | BeOS | полная |

Так же часто бывает, когда проект растет, он от части меняет свою специфику, или же расширяет ее, что приводит к появлению ряда задач которые проще и быстрее и надежней решать используя другой язык программирования, благодоря поддержки не малого колличества языков программирования, мы можем разработать отдельные части или же полностью переписать систему на другом языке программирования, не затронув других настроек сервера. В таблице 2.2 приведен список подрживаемых языков.

**Таблица 2.2 Поддерживаемые языки Apache.**

|  |  |  |
| --- | --- | --- |
| № | Название | Поддержка |
| 1 | [PHP](https://ru.wikipedia.org/wiki/PHP) | Полная |
| 2 | Python | Полная |
| 3 | Ruby | Полная |
| 4 | Perl | Полная |
| 5 | ASP | Полная |
| 6 | TCL | Полная |

Apache предостовляет возможность подключения сторонних модулей, выполнять аутентификацию пользователей используя СУБД, гибкость работы с сообщениями об ошибках а так же множество других дополнительных опций. Включает поддержку IPv6.

Основное что реализовано в ядре Apache это: работа с файлами конфигураций, HTTP(протокол), интерфейс для интеграции модулей которые были разработаны сторонними разработчиками. Разработкой ядра и на сегоднешний день его развитием занимается компания Apache Software Foundation без превличения сторонних разработчиков, которые допускаются только к разработке модулей. Для написания ядра использовался язык Си.

Система конфигурации Apache основывается на фаилах в текстовом формате которые еще называются конфигурационными фаилами. Состоит из трех условных уровней конфигурации:

Конфигурация сервера (httpd.conf).

Конфигурация виртуального хоста (httpd.conf c версии 2.2, extra/httpd-vhosts.conf).

Конфигурация уровня директории (.htaccess).

Для управления конфигурационными фаилами использует собственный язык, который основывается на блоках директив. По средствам фаилов конфигураций можно изменить практически любые параметры ядра. Практически у всех модулей собственные параметры.

Так же есть модули которые в своей работе конфигурационные фаилы операционной системы такие как etc/pass или etc/host.

Система модулей. Apache HTTP Server инегрировать сторонние модули. На сегоднешний день разработанно более 500 модулей, предостовляющих различную функциональность. Некоторые из этих модулей были разработанны непосредственно самой компанией Apache Software Foundation. Но большая часть была разработанна сторонними open source-разработчиками.

Процесс интеграции модулей довольно гибкий, их можно интегрировать как в момент компиляции так и загрузить динмически по средствам конфигурационного фаила используя диррективы.

Создание новых модулей позволяет реализовать такие решения как:

· Поддержка языков программирования.

· Расширение функциональности.

· Доработка включенных в ядро функций.

· Работа с безопасностью.

Подводя итог по выше сказанному, возможность подключать модули, наши возможности становятся практически не ограниченными, вплоть до адаптации его до собственного языка программирования.

В качестве примера, часть веб-приложений, например панели управления ISPmanager и VDSmanager реализованы в виде модуля Apache.

Виртуальные хосты, в ядре Apache реализована технология виртуальных хостов. Такого рода технология дает возможность поддерживать на одном IP-адресе несколько разных доменных имён, выводя содержимое индивидуально для каждого. Так же он позволяет задавать индивидуально настройки ядра, модулей а так же доступов для каждого отдельного виртуального хоста.

Некоторые MPM, например Apache-ITK позволяют запускать процесс httpd для каждого виртуального хоста с отдельными идентификаторами uid и guid.

Также, есть модули, предостовляющие возможноть упровлять ресурсами сервера, такими как трафик, оперативная память, настройки процессора, для каждого из виртуальных хостов в отдельности.

Безопасность, В ядре Apache реализованы функции для обеспечения безопасности и управление доступом к данным. Среди основных можно отметить:

Настройка доступа для дирректорий и фаилов.

Авторизация пользователей для получения доступов используя HTTP-аутентификации (mod\_auth\_basic) и digest-аутентификации (mod\_auth\_digest).

Настрой доступов к непосредственно самому серверу, ориентируясь на IP-адреса.

Есть модули, которые выполняют авторизацию используя системы управления базой данных или подключаемые модули аутентификации.

Также, в Apache присутствует suexec, это механизм выполняющий запуск скриптов а так же CGI-приложений с данными об идентификационными и правами пользователя.

Для шифрования данных, которые передаются между клиентом и сервером используется алгоритм SSL, который был написан с использованием библиотеки OpenSSL. Для удостоверения что веб-сервер подлинный применяют сертификаты X.509.

Присутствуют внешние средства для работы с безопасностью, к таким относится mod\_security.

Интернационализация, в версии 2.0 была добавлена функциональность позволяющая определить локаль клиента на стороне сервера. С этого момента посылаемые браузеру системные сообщения, такие как сообщения об ошибках, могут быть отправленны на разных языках.

Еще один важный момент который стоит отметить, то что появилась возможность управлять на стороне сервера выводом содержимого страниц в зависимости от локали пользователя. Apache в состоянии работать с большим количеством кодировок, в том числе Юникод, это дает возможность обробатывать страницы, сохраненных в любых кодировках и написанных на любых языках.

Обработка событий, администратору предоставленна возможность самому задать собственные страницы и обработчики для возникающих HTTP ошибок или событий, например 404 (страница не найдена) или 403 (запрещен доступ). В том числе возможность запуска скриптов и отображения сообщений на разных языках.

Server Side Includes, начиная с версии 1.3 и выше добавлен механизм Server Side Includes, который дает возможность динамически работать с HTML-документами на серверной стороне.

За управление SSI отвечает модуль mod\_include, который входит в ядро Apache.