

Рассчётно-пояснительная записка по курсовой работе
"Разработка почтового сервера"

Кисленко Максим Германович

19 января 2018 г.

Оглавление

Введение	2
1 Аналитический раздел	3
1.1 Сценарии работы с SMTP-сервером	3
1.2 Достоинства и недостатки реализуемого сервера	3
1.3 Программы аналоги	3
2 Конструкторский раздел	4
2.1 Архитектура сервера	4
2.2 Обработка команд протокола	4
2.3 Работа с Maildir	4
2.4 Логирование в отдельном процессе	4
3 Технологический раздел	5
3.1 Конфигурация в отдельном файле	5
3.2 Автоматические тесты	5
3.3 Сборка программы	5
Выводы	5
Приложение 1. Основные функции программы	7
Приложение 2. Графы вызова функций	8

Введение

Курсовая работа предполагает разработку собственного почтового сервера, который будет использоваться для пересылки электронных писем (email-ов) в IP-сети. Вся работа поделена на две приблизительно одинаковые части - разработка почтового сервера, осуществляющего приём писем, и разработка почтового клиента, который переотправит письма другим почтовым серверам в зависимости от получателей-адресатов. Эти части разрабатываются независимо друг от друга двумя студентами, а потом интегрируются в единую систему. Команде студентов дополнительно присваивается вариант, который определяет основные требования к реализации или функциональности.

При реализации подразумевается, что используются низкоуровневые системные вызовы и, что сервер работает под управлением POSIX-совместимой операционной системы. Последнее условие необходимо, чтобы обеспечить переносимость созданного ПО.

В данном документе будет рассмотрена разработка почтового сервера, с условиями от варианта 12. Условия включают в себя следующие пункты:

- сервер должен обрабатывать входящие соединения в единственном рабочем потоке с применением вызова poll. Такое условие подразумевает использование сокетов в режиме неблокирующего ввода-вывода, так как иначе, единственный поток бы блокировался на одном клиенте и не мог бы считаться полноценным сервером;
- логирование событий и ошибок в работе сервера должно происходить в отдельном процессе. Такое техническое решение увеличивает сложность реализации, но и делает сервер более производительным (отсутствуют операции ввода-вывода на жесткий диск);
- проверка обратной зоны dns не предполагается.

Независимо от условий варианта, при постановке задачи заданы требования к выполнению данной работы. А именно:

- в качестве протокола пересылки электронных сообщений предписано использовать протокол SMTP (simple mail transfer protocol). Вся необходимая информация по нему доступна в RFC 5321;
- для локального хранения писем использовать механизм Maildir;
- конфигурация сервера должна быть вынесена в отдельный конфигурационный файл;
- необходимо разработать автоматическое тестирование созданного ПО (системные, unit-тесты);

- необходимо автоматизировать сборку проекта из исходников.

В поставленной задаче довольно жестко регламентированы используемые технологии. В качестве основного языка программирования - язык Си стандарта 99-го года. Для автоматизации тестирования - cunit или скриптовые языки (python, ruby или другое). Для автоматизации сборки - только GNU Make. Также готовое ПО необходимо протестировать на наличие утечек памяти с помощью утилиты valgrind.

Готовая программа должна обрабатывать входящие соединения, обслуживать их в соответствии с правилами сессии SMTP, сохранять письма локально в Maildir вместе с необходимыми заголовками, и передавать письма на дальнейшую отправку SMTP-клиенту. Далее в этом документе детально рассмотрены:

- Пользовательское взаимодействие с сервером, его достоинства, недостатки и существующие аналоги в аналитическом разделе;
- Проектирование и принятые программные решения при создании ПО в конструкторском разделе;
- Используемые инструменты и технологии, порядок эксплуатации созданного ПО в технологическом разделе;

Глава 1

Аналитический раздел

1.1 Сценарии работы с SMTP-сервером

1.2 Достоинства и недостатки реализуемого сервера

1.3 Программы аналоги

Глава 2

Конструкторский раздел

2.1 Архитектура сервера

2.2 Обработка команд протокола

2.3 Работа с Maildir

2.4 Логирование в отдельном процессе

Глава 3

Технологический раздел

3.1 Конфигурация в отдельном файле

3.2 Автоматические тесты

3.3 Сборка программы

Выводы

Результаты, что сделано и что выполнено.

Приложение 1. Основные функции программы

Приложение 2. Графы вызова функций